

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**



ПОБЕДИТЕЛЬ КОНКУРСА ИННОВАЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММ ВУЗОВ

НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК

Выпуск 48

**МЕХАТРОНИКА, ТЕХНОЛОГИИ,
СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО
ПРОЕКТИРОВАНИЯ**



**САНКТ-ПЕТЕРБУРГ
2008**

В научно-техническом вестнике СПбГУ ИТМО, Выпуск 48 «МЕХАТРОНИКА, ТЕХНОЛОГИИ, СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ» представлены работы, выполненные в рамках:

- инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» вузов России на 2007–2008 гг.;
- аналитической ведомственной целевой программы «Развитие научного потенциала высшей школы (2006–2008 гг.)» (Федеральное агентство по образованию);
- Федеральной целевой программы развития образования на 2006–2010 гг. (Федеральное агентство по образованию);
- Федеральной целевой программы развития научно-технологического комплекса России на 2007–2012 гг. (Федеральное агентство по науке и инновациям);
- Российского фонда фундаментальных исследований, а также инициативные разработки.



В 2007 году СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007–2008 годы. Реализация инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» позволит выйти на качественно новый уровень подготовки выпускников и удовлетворить возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях экономики.

ISSN 1819-222X

© Санкт-Петербургский государственный университет информационных технологий, механики и оптики, 2008

**ИССЛЕДОВАНИЕ ТЕПЛООБМЕНА И ГИДРОДИНАМИКИ
НА ПОВЕРХНОСТЯХ СО СФЕРИЧЕСКИМИ УГЛУБЛЕНИЯМИ
ПРИ РАЗЛИЧНЫХ ЧИСЛАХ РЕЙНОЛЬДСА****О.И. Панова****(Московский государственный технический университет им. Н.Э. Баумана)****Научный руководитель – к.т.н., н.с. К.С. Егоров****(Институт механики МГУ им. Ломоносова)**

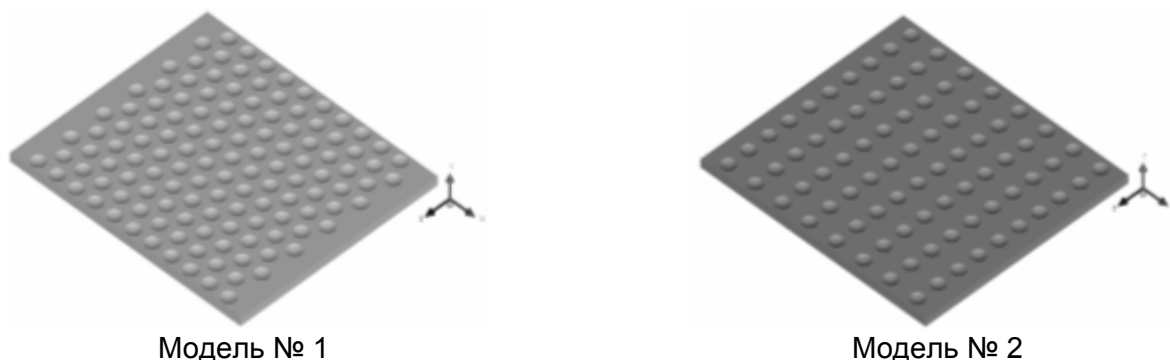
Работа посвящена численному исследованию обтекания потоком теплоносителя луночного рельефа на пластине при различных числах Рейнольдса. Для сравнения выбраны два типа расположения лунок в пакете – шахматное и коридорное. Моделирование проводится с использованием программного комплекса STAR-CD. В заключение приводятся теплогидравлические характеристики таких рельефов в зависимости от числа Рейнольдса.

Введение

Интенсификация теплообмена при обтекании стенок, покрытых полусферическими углублениями (лунками), уже несколько десятилетий является актуальной проблемой теплофизики. Исследование течения в каналах с такими интенсификаторами проводилось множеством научных коллективов и авторов, как в России, так и за рубежом. Стоит отметить, что до недавнего времени работы, посвященные данной тематике, в подавляющем большинстве случаев были экспериментальными [1].

В настоящее время, в связи с бурным развитием компьютерной техники, появилась возможность проведения численного (виртуального) эксперимента даже для таких сложных геометрических моделей. Одна из последних работ в данной области [2] выполнена в ONERA (Office National d'Etudes et de Recherchés Aérospatiales, France) и представляет собой параметрический анализ теплогидравлической эффективности луночных рельефов.

Данная работа посвящена разработке методики расчета гидродинамики и теплообмена таких поверхностей с использованием коммерческого пакета STAR-CD. Для исследования были выбраны две модели пластин (рис. 1) с нанесенными на них в шахматном и коридорном порядке неглубокими лунками (по классификации, приведенной в работе [1]).

**Рис. 1. Модели пластин**

В работе было исследовано влияние скорости внешнего потока на теплогидравлическую эффективность рельефов.

Описание моделей

Для исследования были выбраны две модели пластин с нанесенными на них полусферическими углублениями. Основные геометрические параметры, характеризующие пакет лунок, показаны на рис. 2.

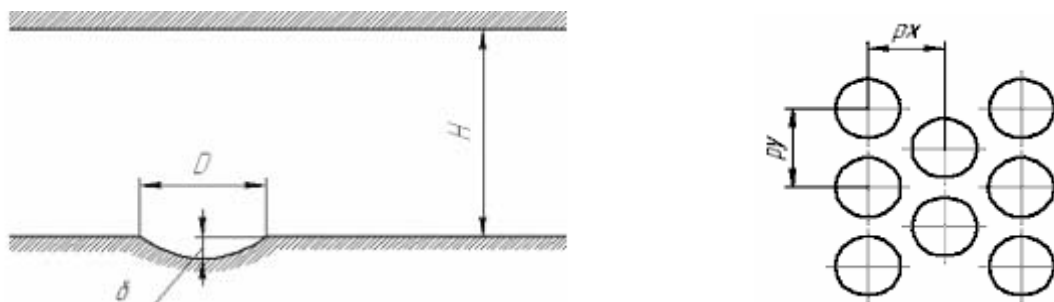


Рис. 2. Геометрические параметры, характеризующие рельеф

Глубина и диаметр лунок одинаковы для обеих моделей и равны $D = 7,5$ мм, $\delta = 1,38$ мм, число лунок и их расположение варьируется (для шахматного расположения лунок в пакете – $p_x = 9,5$ мм, $p_y = 11$ мм; для коридорного – $p_x = 12$ мм, $p_y = 20$ мм).

Высота канала $H = 50$ мм ($H/D = 6,67$), т.е. исследуется обтекание рельефа без влияния верхней стенки, расстояние от входа в канал до центра первой лунки $l_1 = 100$ мм, длина канала $l_{ch} = 500$ мм, длина нагреваемого участка пластины с углублениями $l_h = 133$ мм, толщина пластины $h = 5$ мм.

Математическая модель

Принятые допущения:

- задача стационарная;
- рабочее вещество (воздух) считается ньютоновской, вязкой и несжимаемой средой;
- теплофизические свойства потока и твердого тела принимаются постоянными;
- пластина рассматривается как полубесконечная (рис. 3).

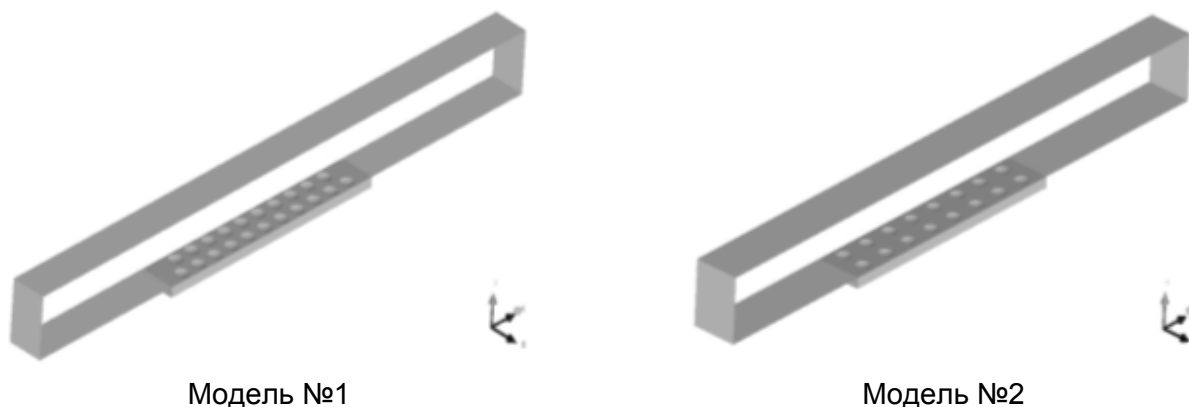


Рис. 3. Расчетная область

Полная система уравнений для описания конвективного теплообмена принимает вид [3, 4]:

- уравнение неразрывности: $div(\rho_f \mathbf{w}) = 0$,
- уравнения движения: $div(\rho_f \mathbf{w} \mathbf{w} - \mu_{ef} grad(\mathbf{w})) = -grad(p)$,
- уравнение состояния: $\rho_f = const$,
- уравнение энергии для теплоносителя: $div(\rho_f c_{pf} \mathbf{w} T_f - \lambda_{ef} grad(T_f)) = 0$, где

$$\lambda_{ef} = \lambda_f + \frac{\mu_t}{Pr_t}$$
- уравнение теплопроводности для пластины: $div(\lambda_c grad(T_c)) = 0$,

где ρ – плотность, кг/м³; μ – динамический коэффициент вязкости, Па·с; μ_t – коэффициент турбулентной вязкости, Па·с; λ – коэффициент теплопроводности, Вт/(м·К); Pr_t – турбулентное число Прандтля; c_p – теплоемкость при постоянном давлении, Дж/(кг·К); p – давление, Па; T – температура, К; \mathbf{w} – вектор скорости, м/с. Индекс f соответствует жидкости, c – твердому телу.

Для моделирования процессов турбулентного переноса использовалась двухпараметрическая модель $k-\omega$ LR (Low Reynolds). Уравнения, описывающие данную модель, приведены в [5].

Система уравнений определена для области, ограниченной поверхностями, указанными на рис. 4.

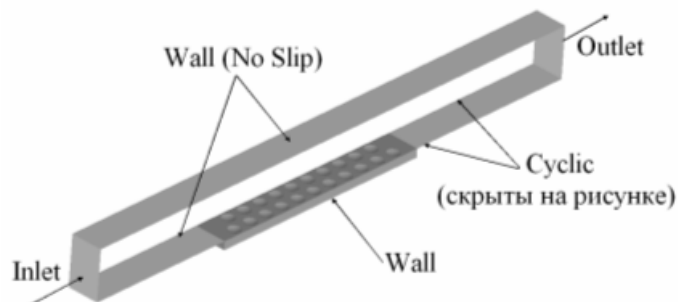


Рис. 4. Границы расчетной области

Граничные условия:

- на входе в расчетную область (Inlet) задаются постоянные скорость и температура: $u_0 = 10..90$ м/с, $T_0 = 293$ К,
- на выходе из расчетной области (Outlet) задаются условия продолжения решения:

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = \frac{\partial w}{\partial n} = \frac{\partial T}{\partial n} = 0,$$
- на боковых гранях задаются условия периодичности:

$$u|_{Cycl1} = u|_{Cycl2}, v|_{Cycl1} = v|_{Cycl2}, w|_{Cycl1} = w|_{Cycl2}, T|_{Cycl1} = T|_{Cycl2},$$
- на нижней стенке твердого тела (Wall) задается постоянная температура: $T_w = 310$ К,
- на границе сопряжения твердого тела и жидкости (Wall No Slip) задается условие прилипания и равенство потоков теплоты: $u = v = w = 0, q_f = q_c$,
- на верхней стенке задано условие прилипания, стенка адиабатная:

$$u = v = w = 0, \frac{\partial T}{\partial n} = 0.$$

Теплофизические свойства рабочей среды и материала пластины приведены в табл. 1.

Свойства	Воздух	Оргстекло
Плотность, кг/м ³	1,205	1080
Вязкость, кг/(м·с)	$1,81 \cdot 10^{-5}$	-
Теплоемкость, Дж/(кг·К)	1006	1100
Теплопроводность, Вт/(м·К)	0,026	1,07

Таблица 1. Свойства веществ, используемые при расчете

Построение сетки расчетной области

Сетка строится на основе твердотельных моделей пластин и выполняется более подробной в пограничном слое. Фрагменты сеток для пластин приведены на рис. 5.

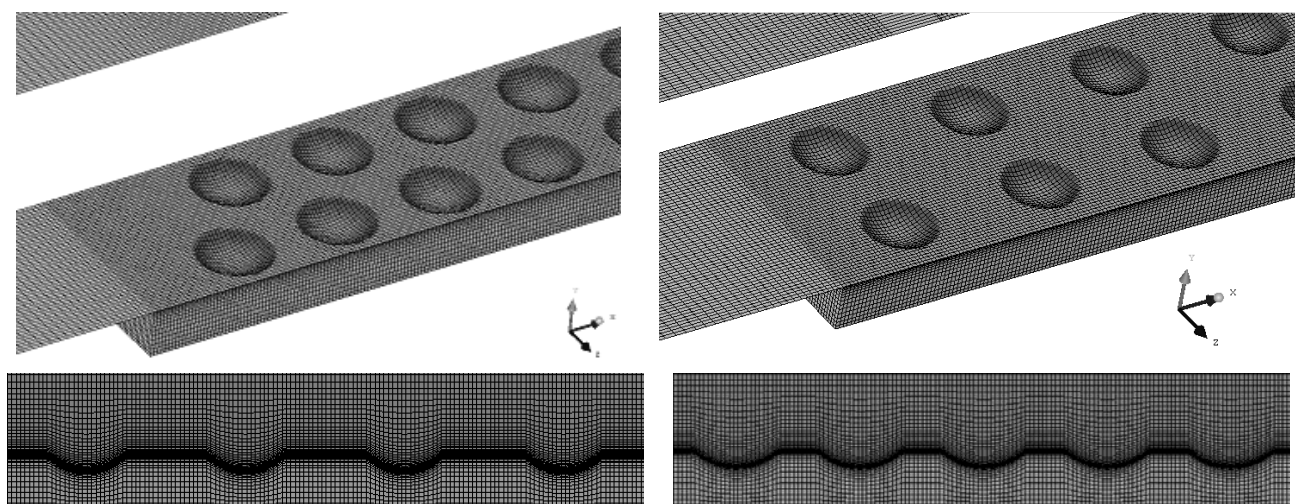


Рис. 5. Фрагменты сетки для пластин (для шахматного расположения лунок – число ячеек 1738966, число узлов 1798980, для коридорного – 1620626 и 1681650, соответственно)

Результаты численного моделирования

Расчет проводился для нескольких скоростей внешнего потока – 10, 30, 60, 90 м/с, что соответствует числам Рейнольдса $5,7 \cdot 10^4$, $1,7 \cdot 10^5$, $3,4 \cdot 10^5$, $5,1 \cdot 10^5$. На рис. 6 приведены поля температур на пластинах для $Re = 5,7 \cdot 10^4$.

Поля температур демонстрируют неравномерность полей тепловых потоков на луночной поверхности. Это объясняется тем, что в пределах луночного рельефа образуются несимметричные вихревые структуры, подобные тем, что были получены в работе [6].

На рис. 7 и 8 приведены графики температур на поверхности пластины в сечении 1, показанном на рис. 6. Для сравнения на том же графике приведено распределение температуры на плоской нагретой пластине при идентичной постановке.

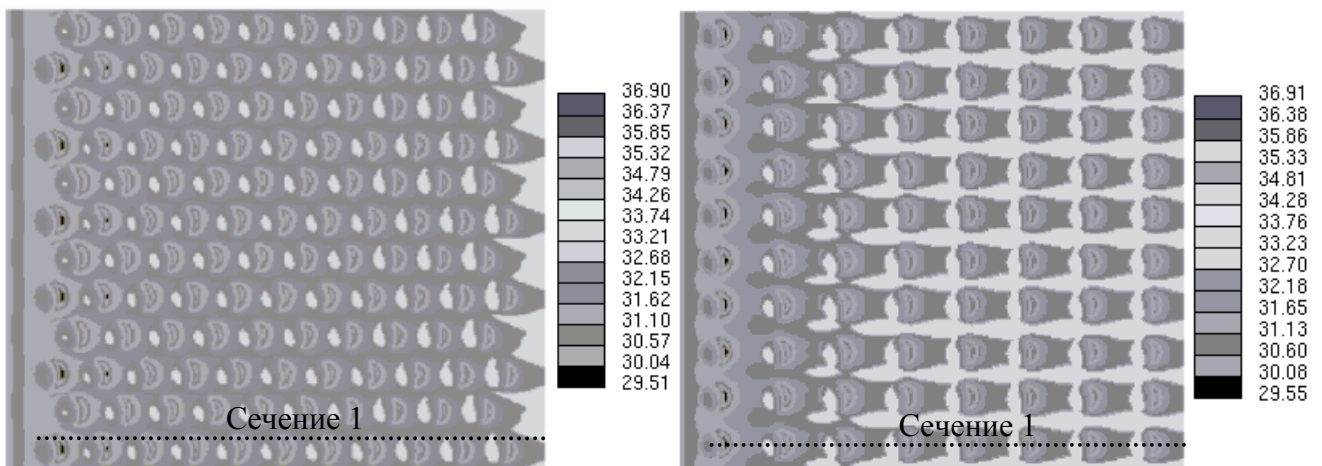


Рис. 6. Поля температур на пластинах при скорости внешнего потока 30 м/с

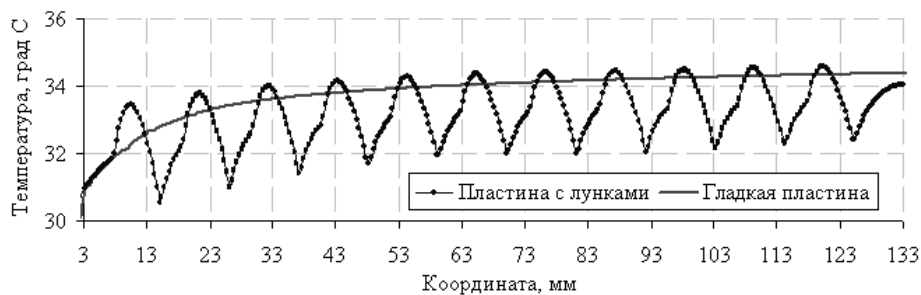


Рис. 7. Температура на пластине – сечение 1 (модель №1)

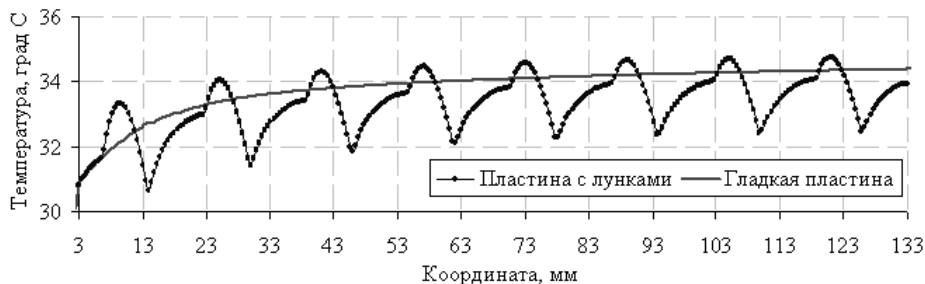


Рис. 8. Температура на пластине – сечение 1 (модель №2)

Влияние числа Рейнольдса

В работе [2] отмечалось незначительное влияние числа Рейнольдса на относительную теплоотдачу пластин с луночным рельефом на малых скоростях внешнего потока. Чтобы установить это влияние, рассматривается распределение относительных чисел Нуссельта Nu/Nu_0 у поверхности пластины в сечении 1 (Nu на пластине с лунками относительно Nu_0 на гладкой). Зависимость относительного числа Нуссельта от числа Рейнольдса и координаты представлена на рис. 9 и 10.

Как видно из графиков, влияние числа Рейнольдса незначительно на высоких скоростях внешнего потока. При тщательном рассмотрении можно заметить эффект увеличения относительной теплоотдачи при увеличении числа Рейнольдса. Однако увеличение ее незначительно по сравнению с ростом сопротивления, что будет показано далее. Таким образом, при высоких числах Рейнольдса использование таких рельефов может быть невыгодным.

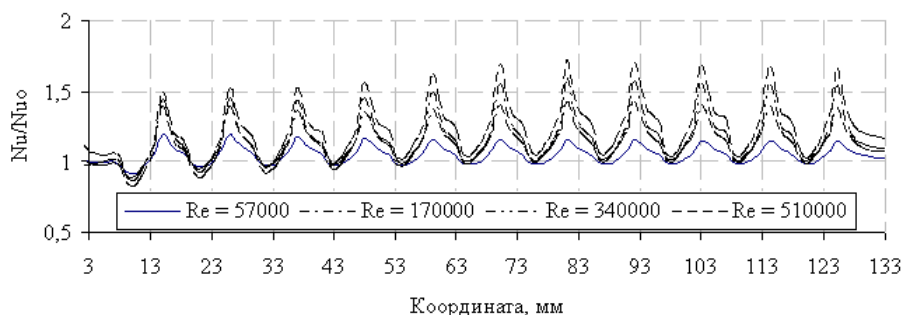


Рис. 9. Относительное число Нуссельта для различных чисел Рейнольдса Сечение 1 (модель №1)

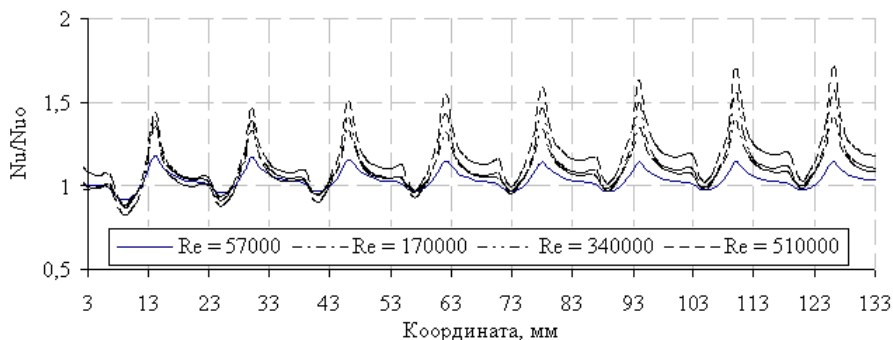


Рис. 10. Относительное число Нуссельта для различных чисел Рейнольдса Сечение 1 (модель №2)

Теплогидравлическая эффективность

Универсальным «индикатором» увеличения теплогидравлической эффективности любой поверхности является отношение числа Нуссельта к коэффициенту трения по сравнению с плоской пластиной: $\eta = \frac{Nu/Nu_0}{f/f_0}$. Для определения коэффициента трения

на плоской пластине используется формула: $f_0 = 0,00140 + 0,125 \cdot Re^{-0,32}$ [2]. Коэффициент трения на пластине с лунками определяется непосредственно по потерям статического давления: $f = \frac{H\Delta P}{\rho u^2 \Delta x}$. Полученные в результате численного моделирования ко-

эффициенты трения сопоставляются с экспериментальными данными, полученными в лаб. 108 НИИ Механики МГУ им. М.В. Ломоносова [7] для идентичной модели №1 пластины. Данные и погрешности численного исследования приведены в табл. 2.

Re	$f_{\text{эксп}}$	$f_{\text{числ}}$	Погрешность, %
170000	0,0125	0,0121	3
340000	0,0095	0,0110	15
510000	0,0087	0,098	12

Таблица 2. Сравнение экспериментальных и численных данных для коэффициента трения

Завышенное значение коэффициента трения может быть связано с тем, что для численного моделирования использовались очень точные с точки зрения геометрии модели. На экспериментальных моделях острые кромки могут быть слегка скруглены, что снижает сопротивление за счет менее интенсивного вихреобразования. Данные по теплогидравлической эффективности для различных чисел Рейнольдса представлены в табл. 3.

Re	Nu/Nu ₀	f/f_0	η
Шахматное расположение лунок в пакете			
57000	1,05	1,35	0,78
170000	1,18	2,99	0,39
340000	1,21	3,12	0,38
510000	1,3	3,0	0,40
Коридорное расположение лунок в пакете			
57000	1,04	0,97	1,15
170000	1,15	1,34	0,85
340000	1,2	1,58	0,76
510000	1,28	1,73	0,73

Таблица 3. Теплогидравлическая эффективность

Заключение

Данные по теплогидравлической эффективности указывают на то, что на низких скоростях целесообразно использовать рельефы с коридорным расположением лунок. При повышении скорости сопротивление рельефа значительно возрастает по сравнению с теплоотдачей. Столь быстрый рост сопротивления для пластины с шахматным расположением лунок можно объяснить использованием более плотной упаковки углублений и, как следствие, большего числа источников сопротивления. Для более высоких скоростей следует рассмотреть лунки со скругленными кромками и менее плотную их упаковку.

Использование численного моделирования при проектировании новых устройств позволит значительно сократить расходы на проведение дорогостоящих натурных экспериментов. Данная работа представляет собой первый этап создания методики расчета подобных задач с использованием коммерческих ПК. В дальнейшем планируется продолжить данные исследования по оптимизации рельефов и перейти к нестационарным расчетам для детального рассмотрения картины течения.

Литература

1. Халатов А.А. Теплообмен и гидродинамика около поверхностных углублений (лунок) – Киев: НАН Украины, Институт технической теплофизики, 2005. – 59 с.
2. Grenard Ph., Quintilla-Larroya V., Laroche E. Numerical Study of Heat Transfer on a Dimpled Surface with CEDRE code // 2nd European conference for aerospace sciences (EUCASS), 2007.
3. Себеси Т., Брэдшоу П. Конвективный теплообмен / Пер. с англ. М.: Мир, 1987. – 592 с.
4. Лойцянский Л.Г. Механика жидкости и газа: Учебник для вузов. – 7-е изд., испр. – М.: Дрофа, 2003. – 840 с.
5. Methodology STAR-CD version 3.26 // STAR-CD Version 3.20 Documentation, CD adapco Group, 2004.
6. Быстров Ю.А., Исаев С.А., Кудрявцев Н.А., Леонтьев А.И. Численное моделирование вихревой интенсификации теплообмена в пакетах труб – СПб: Судостроение, 2005. – 392 с.
7. Виноградов Ю.А., Здитовец А.Г., Стронгин М.М., Титов А.А. Экспериментальное исследование нарушения аналогии Рейнольдса на поверхностях с вихреобразующим рельефом // Тезисы докладов XV школы-семинара под руководством акад. РАН Г.Г.Черного – М.: Изд-во Московского университета, 2007. – С. 92–93.

МЕТОД ИЗМЕРЕНИЯ ТЕПЛОПРОВОДНОСТИ АНИЗОТРОПНЫХ ВЫСОКОТЕПЛОПРОВОДНЫХ ТЕЛ В ВИДЕ СТЕРЖНЕЙ И ПЛАСТИН

А.Д. Исламова

Научный руководитель – к.т.н., доцент В.А. Кораблев

Приведена методика измерения теплопроводности высокотеплопроводных образцов, изготовленных в форме стержней и пластин неразрушающим методом. На свободной поверхности образцов имеет место вынужденная конвекция для создания равномерного коэффициента теплоотдачи и интенсификации теплообмена.

Введение

Современные способы изготовления конструкционных материалов на основе нанотехнологий позволяют получать высокотеплопроводные материалы в виде стержней и пластин. Они используются, в частности, для обеспечения теплового режима электронных элементов в разрабатываемых в настоящее время радиоэлектронных устройствах. Образцы этих материалов обладают значительной анизотропией теплопроводности, причем ее наибольшее значение обычно лежит вдоль стержня или пластины. Среди существующих на сегодняшний день методов измерения теплофизических свойств веществ [1] не было найдено подходящего, для измерения теплопроводности твердых тел в диапазоне от 200 до 800 Вт/м·К. Такие измерения необходимы для контроля качества изготавливаемых материалов и для физических исследований.

Теория метода

Измерение теплопроводности длинных образцов осложнено наличием теплообмена на их боковой поверхности, и этот фактор был одним из основных источников погрешности измерений. Рассматриваемый метод основан на тщательном учете лучисто-конвективного теплообмена на поверхности исследуемого образца, нагреваемого с одного конца и охлаждаемого с боковой поверхности. Уравнение теплопроводности для стержня постоянного сечения в случае, когда его теплопроводность λ не зависит от температуры, в стационарном тепловом режиме имеет вид

$$\frac{d^2\vartheta(x)}{dx^2} = \frac{\alpha P}{A\lambda} \vartheta(x). \quad (1)$$

Здесь x – координата по длине стержня, P , A – периметр и площадь его поперечного сечения, ϑ – перегрев стержня относительно среды, $\alpha(\vartheta(x))$ – коэффициент теплоотдачи.

Граничные условия к уравнению (1) в случае, когда начало оси x совпадает с местом крепления стержня, в котором к нему подводится тепло, а его свободный конец находится достаточно далеко от места крепления, так что стержень полагается полубесконечным, имеют вид

$$\vartheta|_{x=0} = \vartheta_0, \quad \vartheta|_{x=l} = 0, \quad \left. \frac{d\vartheta}{dx} \right|_{x=l} = 0. \quad (2)$$

Решение этого уравнения имеет вид [4]

$$\vartheta(x) = \vartheta_0 \frac{\operatorname{ch} \left[\sqrt{\frac{\alpha \Pi}{\alpha A}} (x-l) \right]}{\operatorname{sh} \left(\sqrt{\frac{\alpha \Pi}{\lambda A}} \cdot l \right)}. \quad (3)$$

Если измерить профиль температур $\vartheta(x)$ нагретого до стационарного состояния стержня и аппроксимировать его в виде формулы (3), то, зная величину лучисто-конвективного теплообмена α и размеры образца A , Π , можно получить величину теплопроводности λ , решив относительно нее уравнение (3). Решение этого уравнения возможно только численно, методом последовательных приближений в общем случае.

Если стержень настолько длинный, что его холодный конец имеет температуру среды, т.е. выполняется граничное условие (2), то выражение (3) можно привести к виду

$$\vartheta(x) = \vartheta_0 \exp \left(- \sqrt{\frac{\alpha \Pi}{\lambda A}} \cdot x \right), \quad (4)$$

и решение этого уравнения относительно λ может иметь вид

$$\lambda = \frac{\alpha \Pi x^2}{A \left[\ln \left(\frac{\vartheta_0}{\vartheta(x)} \right) \right]^2}. \quad (5)$$

Для такого определения теплопроводности λ необходимо знать величину коэффициента теплообмена на поверхности образца со средой α . Значение α должно быть равномерно распределено по всей длине образца. Этого можно достигнуть, обдувая образец поперечным потоком воздуха с помощью вентилятора. Величину α можно определить экспериментально, изготовив специальный тепломер в виде образца с равномерно распределенным по длине нагревателем и рядом датчиков температур, расположенных на его поверхности. После выхода термометра, нагреваемого в потоке воздуха, на стационарный тепловой режим измеряем перегрев его поверхности относительно набегающего потока воздуха, и коэффициент теплообмена рассчитываем по формуле:

$$\alpha = \frac{\Phi}{\bar{\vartheta} \Pi l}, \quad (6)$$

где Φ – электрическая мощность, подаваемая на тепломер, $\bar{\vartheta}$ – средний перегрев поверхности тепломера относительно потока воздуха, Π и l – периметр поперечного сечения тепломера и его длина.

Методика проведения измерений

На рис. 1 представлена схема установки для измерения теплопроводности стержня вдоль его оси. Образец материала 1 в виде стержня устанавливался в штативе над вентилятором, обеспечивающим равномерный поток воздуха со скоростью U . На одном из его концов установлен нагреватель 2, закрытый теплоизоляцией 3. На поверхности образца установлено несколько термоэлектрических термометров 4, измеряющих перегрев образца относительно набегающего потока воздуха. Нагреватель питался от стабилизированного источника постоянного тока. Измерение сигналов от термометров проводилось при помощи цифрового вольтметра.

Для градуировочных опытов были изготовлены также тепломер, имеющий форму образца, для измерения коэффициента лучисто-конвективного теплообмена α и образцы из высокотеплопроводных материалов (медь МЗ, сплав АМГ) для проверки предлагаемого способа измерения теплопроводности. Были проведены исследования лучисто-конвективного теплообмена на поверхности тепломера при различных скоростях воз-

духа и температурах поверхностей при помощи тепломера, и получили зависимости для определения величин α для расчетов по формулам (3) и (5).

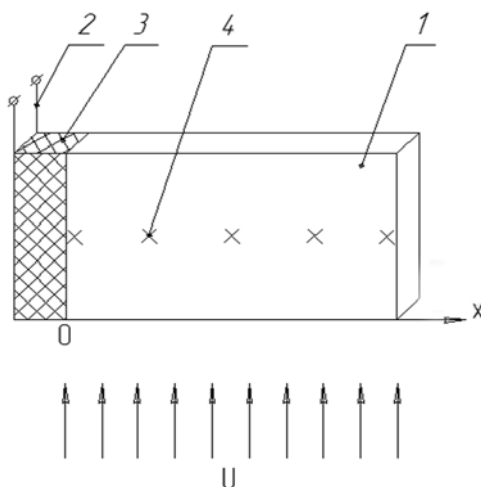


Рис. 1. Схема установки для измерения теплопроводности стержня вдоль его оси

Исследования образцов, выполненных из материалов с известными свойствами, показали, что суммарная погрешность метода определения теплопроводности составляет 12 % при доверительной вероятности 0,9. Результаты измерений теплопроводности композиционных материалов, полученных с использованием нанотехнологий на основе углерода и пироуглерода, показали, что в отдельных направлениях их теплопроводность может достигать значений $500\text{--}600 \frac{\text{Вт}}{\text{м}\cdot\text{ЧК}}$.

Заключение

Метод измерения теплопроводности, разработанный для образцов, имеющих форму пластины или стержня, может быть использован для исследований материалов, имеющих существенную анизотропию свойств и аномально высокую теплопроводность. Также метод может быть использован для текущего контроля изделий на производстве, так как он не требует изготовления образца с заданными геометрическими параметрами, что выгодно его отличает от существующих калориметрических методов.

Литература

1. Кондратьев Г.М., Дульнев Г.Н., Платунов Е.С., Ярышев Н.А. Прикладная физика: Теплообмен в приборостроении. – СПб: СПбГУ ИТМО, 2003. – 560 с.
2. Буравой С.Е., Куренин В.В., Петров Г.С., Платунов Е.С. Теплофизические измерения и приборы. – Л.: Машиностроение, 1986. – 256 с.
3. Осипова В.А. Экспериментальное исследование процессов теплообмена – М.: Энергия, 1979. – 320 с.
4. Дульнев Г.Н., Семяшкин Э.М. Теплообмен в радиоэлектронных аппаратах. – Л.: Энергия, 1968. – 360 с.

ВЫДЕЛЕНИЕ СТРУКТУРНЫХ ПРИЗНАКОВ РАСПОЗНАВАЕМЫХ СИМВОЛОВ НА ОСНОВЕ НАБОРА КЛЕТОЧНЫХ АВТОМАТОВ

Д.И. Суясов

Научный руководитель – д.т.н., профессор А.А. Шалыто

Статья содержит описание решения задачи выделения структурных признаков из изображений символов. Данная задача ставится в системах распознавания текста и напрямую влияет на качество распознавания. В работе предлагается использовать набор из клеточных автоматов с метками. Строится последовательность автоматов для определения концов, пересечений и циклов в символах.

Введение

Важной задачей современных интеллектуальных систем является задача распознавания текста. Распознавание текста используется в системах анализа документов, при переводе в электронный вид книжных архивов, для анализа анкет результатов при тестировании и во многих других сферах.

Проблемой при распознавании текста является недостаточная точность распознавания. Работы в этой области ведутся в основном специальными научными группами и коммерческими компаниями.

Большинство работ по решению задач в области систем распознавания предполагает модульный подход: отдельные модули решают задачи по фильтрации, сегментированию, выделению признаков и классификации изображения. Точность результатов работы каждого модуля, зачастую, в значительной степени сказывается на общем результате распознавания текста [1]. Разработка перспективных методов, лежащих в основе модулей системы распознавания и, в частности, в основе модуля выделения признаков символов из изображений, позволяет повысить общее качество распознавания и найти новые альтернативные пути решения задачи.

Стоит отметить, что к сегодняшнему дню предложены варианты решения задачи по выделению признаков символов. К ним относятся выделение скелетов символов [2, 3], выделение признаков на основе преобразующих функций [4] и другие.

Решение, предложенное в данной работе, определяет способ поиска и идентификации признаков символов из их изображений. Подход на основе клеточных автоматов позволяет использовать простой математический аппарат без операций с плавающей точкой. Введенное в работе понятие клеточного автомата с метками позволяет без модификации исходного изображения выделять признаки символов.

Клеточные автоматы и их польза

Понятие клеточного автомата введено в работах Джона фон Немана (John von Neumann) и Конрада Цузе (Konrad Zuse). В начале 40-х годов XX века они разработали идею универсальной вычислительной среды для построения, анализа и сравнения характеристик алгоритмов [5].

Формально клеточный автомат можно определить как набор: $\{G, Z, N, f\}$, где G – метрика поля, на котором действует клеточный автомат; Z – множество состояний каждой клетки; N – окрестность клетки, которая влияет на состояние данной клетки; f – правила клеточного автомата, которые в математическом виде могут быть записаны как $Z \times Z^{|N|} \rightarrow Z$.

Свойствами клеточного автомата являются: локальность правил, однородность системы, конечность множества состояний клетки, одновременность изменений для всех клеток. Более подробно и формально основные свойства классического клеточного автомата приведены в работе [6].

Существует несколько разновидностей клеточных автоматов. Большинство из них приведено в работе Стивена Вольфрама (Stephen Wolfram) «A New Kind of Science» [7]. Выделяют основные типы клеточных автоматов.

(1) Классические клеточные автоматы – автоматы, где клетки поля изменяют состояние в зависимости от состояний соседних клеток.

(2) Мобильные автоматы характеризуются наличием одной или нескольких активных клеток: правила переходов преобразуют только активные клетки.

(3) Тьюринговая машина – это усложнение мобильного автомата, где активная клетка имеет различные состояния.

(4) В системах подстановок автоматы заменяют клетку набором из других клеток.

(5) Циклические системы характеризуются чередованием применения правил клеточного автомата.

(6) Регистровые машины – это автоматы, выполняющие заданные программы по изменению состояний клеток.

(7) В символьных системах клеточный автомат работает с системами символов.

Наиболее известным приложением клеточных автоматов можно считать игру «Жизнь» Джона Конвея (John Conway) [5, 8]. Также некоторые работы по применению клеточных автоматов открыты для свободного доступа на сайте кафедры «Технологии программирования» СПбГУ ИТМО [9].

Клеточные автоматы с метками

Расширением понятия клеточного автомата может служить клеточный автомат с метками. Принцип клеточной системы с метками заключается в ассоциации каждой клетки поля с одним или несколькими метками. При этом для клетки может быть введена как одна метка, так и несколько. Они исполняют роль маркеров и используются в правилах клеточного автомата. Пример правила клеточного автомата с метками представлен на рис. 1.

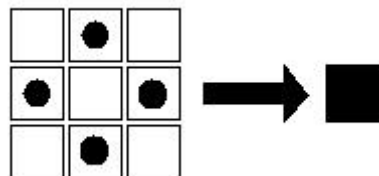


Рис. 1. Пример правила перехода клеточного автомата с метками: правило перекрашивает белую клетку в черный цвет, если четыре соседних клетки содержат метки

Формально клеточный автомат с метками может быть описан следующим образом. Клеточный автомат с метками – это набор $\{G, M, Z, N, f\}$, где G – конечное дискретное метрическое множество, гарантирующее конечность расстояний между клетками; M – конечное множество меток, определенное для каждой клетки; Z – конечный набор состояний клеток; N – конечное множество, определяющее окрестность клетки таким образом, что каждый элемент множества позволяет определить соседа для каждой клетки ($|N|$ – количество соседних клеток, которые влияют на состояние данной клетки); f – правила клеточного автомата, соответствующие математической функции переходов $Z \times C \times Z^{|N|} \times C^{|N|} \rightarrow Z \times C$, $C \subset M$.

Стоит отметить, что клеточный автомат с метками может интерпретироваться как расширение понятий мобильного клеточного автомата и клеточного автомата типа «Тьюринговая машина», в котором определена одна активная клетка. Клеточный автомат с метками сводится к данным типам клеточных автоматов введением ограничений

на правила переходов: правила переходов действуют только для клетки, которая содержит метку (для мобильного клеточного автомата определена одна единственная метка), клеточный автомат с метками данное ограничение не содержит.

Дополнительно можно провести аналогию клеточного автомата с метками с понятием памяти в тьюринговых машинах [10]. Головка тьюринговой машины может хранить некоторое конечное число данных. Правила автомата машины используют эти данные для изменения ленты машины или передвижения ее головки.

Клеточные автоматы с метками могут быть применены в системах со сложной логикой, где клеткам необходимо хранить информацию в процессе работы автомата. В частности, данный тип автоматов применяется при выделении структурных признаков графических изображений символов.

Набор клеточных автоматов

Клеточный автомат является мощным инструментом по выполнению задач моделирования. Четко определив все его составляющие (метрику поля, состояния клеток, количество влияющих на клетку соседей и правила работы автомата), можно решить огромное число задач, многие из которых имеют нетривиальные решения [7].

К сожалению, не все задачи, решения которых возможны на основе клеточных автоматов, могут быть решены одним автоматом. Для решения может потребоваться набор (последовательность) из клеточных автоматов, каждый из которых будет решать специализированную задачу. Набор клеточных автоматов – это совокупность клеточных автоматов, связанных единой метрикой поля и одинаковым набором состояний клеток, но имеющих разный набор правил переходов. Клеточные автоматы в наборе запускаются последовательно, каждый очередной клеточный автомат начинает работу на поле, которое было сформировано предыдущим клеточным автоматом в последовательности.

Как было отмечено, каждый клеточный автомат в наборе запускается после завершения работы предыдущего автомата. Соответственно, должны быть определены правила завершения работы автомата. Подобные правила могут быть заданы несколькими способами.

(1) Клеточный автомат запускается на выполнение четко заданного количества итераций.

(2) Выполнение клеточного автомата происходит до тех пор, пока к полю применимо хотя бы одно правило автомата.

(3) Автомат выполняется до тех пор, пока на поле происходят изменения клеток.

Кроме правил завершения работы клеточных автоматов, набор может также характеризоваться наличием управляющих элементов – не клеточных автоматов. Такими элементами могут быть условные операторы, операторы переходов, операторы циклов и другие. Операторы, заданные в наборе клеточных автоматов, позволяют расширить применимость понятия набора клеточных автоматов. Сложные системы могут быть разбиты на небольшие подзадачи, решаемые в рамках отдельных клеточных автоматов и объединенные единой логикой управления.

Логика, определяющая правила завершения работы клеточных автоматов и порядок и последовательность их запуска, является идеальным примером возможности внедрения конечного автомата [9, 11].

Схема по выделению признаков

Последовательность клеточных автоматов также применима в таких сложных системах, как системы распознавания символов. Системы OCR (optical character recognition – англ. оптическое распознавание символов) или системы распознавания текста ис-

пользуют разные стратегии работы. Наиболее распространенной стратегией является разбиение процесса распознавания текста на этапы.

- (1) фильтрация изображения текста: очистка от шумов и помех;
- (2) разбиение изображения на текстовые блоки;
- (3) выделение в каждом блоке текстовых строк и отдельных символов;
- (4) сбор информации о каждом символе и/или тексте целиком;
- (5) классификация символов на основе выделенных признаков.

Каждый этап распознавания очень важен и влияет на качество результата работы системы. В данной статье рассматривается четвертый этап, а именно выделение структурных признаков символов.

Человек в процессе чтения определяет символ текста на основе целого ряда признаков: это и наклон, и изгибы, и шрифт, и положение, и многие другие. Для качественной работы машины в некоторых случаях достаточно малого количества наиболее характерных признаков. Такими признаками могут быть наличие и положение концов символов, количество петель, положение пересечений отрезков в символе. Для обработки изображений русских букв данных признаков в большинстве случаев достаточно. На рис. 2 указаны признаки для первых четырех букв русского алфавита.

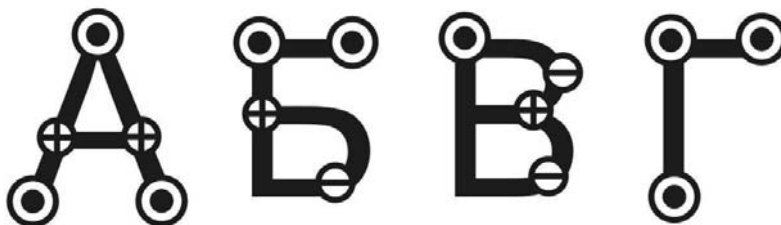


Рис. 2. Выделенные признаки (концы, петли и пересечения) первых четырех букв русского алфавита

Выделение указанных признаков может быть с успехом выполнено с помощью последовательности клеточных автоматов с метками. Стратегия выделения признаков основана на распространении волны.

От верхнего края символа вдоль точек, составляющих данный символ, пускается волна. Эта волна может разделяться на составляющие, повторяя контур изображения символа. В определенный момент составляющие волны могут встретиться или затухнуть на конце символа. На рис. 3 показано направление движения волны вдоль изображения символа «А» и показаны места старта волны, затухания на концах символа и в месте встречи ее двух составляющих. В процессе распространения волны в момент, когда определяется клетка конца символа или клетка встречи двух составляющих волны, генерируется ответная эхо-волна вдоль уже пройденных клеток. Эхо проходит путь в обратном направлении и отмечает клетки, в которых начальная волна была разделена на составляющие. Это позволяет найти позиции пересечений отрезков, из которых состоят символы.

В понятие волны в данном алгоритме вкладывается несколько составляющих. Фронт волны – точки символа, которые передвигаются от одного конца изображения символа к другому. Шлейф волны – это точки изображения, в которых в предыдущий момент времени находился фронт волны. Точки пройденного пути – это точки изображения, где присутствовал фронт волны, а затем и шлейф волны, в этих точках процесс не возобновляется.

В начальный момент времени все точки символа не помечены метками. Затем первая точка волны помечается, и алгоритм начинает свою работу.

Алгоритм основан на идее, что в процессе прохождения волны в какой-то момент времени фронт волны угаснет, в то время как шлейф все еще будет присутствовать.

Данное событие может случиться только на конце символа либо на месте встречи двух составляющих волны. Позиция шлейфа волны в этот момент запоминается.



Рис. 3. Направление распространения волны вдоль символа «А» с фиксированием позиций концов символов и петель, а также распространение эхо с фиксированием позиции пересечений (1, 2, 3 – шаги распространения волны и эхо)

Дополнительно, событие встречи двух составляющих волны регистрируется на основании факта, что шлейфы двух составляющих волны в момент встречи не связаны между собой. Таким образом, запоминается позиция петли символа.

Роль клеточных автоматов с метками в процессе работы алгоритма заключается в выделении информации без изменения изображения символа. Автоматы работают в качестве сборщиков данных об изображении и никак не влияют на входные данные.

Реализация алгоритма

Для запуска алгоритма и анализа результатов его работы была разработана программа. Программа позволяет создавать отдельные клеточные автоматы, в том числе клеточные автоматы с метками и последовательности клеточных автоматов. Также программа позволяет запускать клеточные автоматы в пошаговом или автоматическом режимах.

На основе разработанной программы был смоделирован алгоритм по выделению концов, петель и пересечений в изображениях символов. Эксперименты с запуском алгоритма для различных вариантов текста, с варьированием шрифта, его размера, наклона, толщины показали работоспособность алгоритма. Характерные результаты представлены ниже.

(1) Позиции и количество выделенных признаков согласуется с теоретически предсказанными значениями.

(2) Результат работы алгоритма не зависит от выбранных размеров и толщины символов.

(3) Наклон символов смещает положение структурных признаков относительно друг друга, что должно учитываться в дальнейших шагах процесса распознавания символов.

(4) Результат работы алгоритма меняется в зависимости от выбранного шрифта. Количество и положение петель, пересечений, концов символов варьируется при обработке изображений символов различных шрифтов.

(5) Скорость работы алгоритма зависит от размеров обрабатываемого изображения. Предполагается использование оптимизационных подходов при увеличении разрешения изображений символов.

(6) Выделенные признаки символов представлены в виде меток в клетках изображения. Исходное изображение символа не изменяется.

Результаты работы и развитие идеи

Разработанный алгоритм выделения признаков символов из их изображений на основе последовательности клеточных автоматов с метками показал стойкость к различным вариантам начертания символов и зависимость от выбранного шрифта. Дополнительно стоит отметить, что результат алгоритма чувствителен к помехам на изображении символа. Сильными сторонами алгоритма являются:

- (1) простота вычислений;
- (2) отсутствие операций с плавающей точкой;
- (3) возможность одновременной обработки всех символов;
- (4) возможность распараллеливания процесса выполнения клеточных автоматов;
- (5) устойчивость к размерам и толщине символов.

Относительно невысокие скоростные характеристики работы моделирующей программы могут быть компенсированы внедрением оптимизационных методов (введение обобщенных координат, исключение из обработки неиспользуемых частей полей, объединения точек изображения) и распараллеливанием процесса обработки всех символов текста. Алгоритм может использоваться в модульных системах по распознаванию текста в качестве системы по выделению характерных признаков символов. Выделенные признаки могут использоваться в нейронных сетях, статистических методах как входные параметры.

Дальнейшие исследования в области применимости клеточных автоматов в процессе распознавания текста заключаются в оптимизации моделирования работы клеточных автоматов, распараллеливании процесса обработки нескольких символов текста, исследовании методов классификации признаков на основе клеточных автоматов.

Заключение

Задача выделения структурных признаков символов в системах распознавания текстов поставлена давно. Повышение качества и выделение характерных признаков символов – на сегодняшний день все еще актуальная проблема. Существующие работы не решают задачу полностью, и появление новых методов и механизмов выделения признаков способствует развитию OCR области.

Алгоритм выделения структурных признаков символов из их изображений на основе набора клеточных автоматов позволяет по новому подойти к решению данной задачи и предоставляет инструментарий для дальнейших исследований.

Представленный алгоритм характеризуется простотой вычислений без операций с плавающей точкой, возможностью параллельного вычисления, независимостью от ряда характеристик символов. Данные признаки выделяют предложенный алгоритм среди существующих аналогов. В то же время, результаты алгоритма зависят от шума в символах и некоторых особенностей шрифтов.

Литература

1. Бондаренко А.В., Галактионов В.А., Горемычкин В.И., Ермаков А.В., Желтов С.Ю. Исследование подходов к построению систем автоматического считывания символьной информации – М.: ИПМ им. М.В.Келдыша РАН, 2003.
2. Фролов А.Б., Четрафилов И.Д. О некоторых подходах к распознаванию оптических образов текстов // Интеллектуальные системы. – 1997. – Т. 2. – Вып. 1–4. – С.189–200.
3. Zhang T.Y., Suen C.Y. A Fast Parallel Algorithm for Thinning Digital Patterns. Image Processing and Computer Vision // Communications of the ACM. – March 1984. – Volume 27. – № 3. – P. 236–239.

4. Вершок Д.А. Алгоритмические средства обработки и анализа изображений на основе преобразования Хафа. Автореф. дис. ... канд. техн. наук. – Минск, 2002. – 22 с.
5. Астафьев Г.Б., Короновский А.А., Храмов А.Е. Клеточные автоматы. Учебное пособие. – Саратов: изд-во ГосУНЦ «Колледж», 2003. – 25 с.
6. Наумов Л.А. Разработка среды и библиотеки CAME&L для решения задач с использованием клеточных автоматов. – СПб, СПб ГУ ИТМО, 2003. – Режим доступа: <http://is.ifmo.ru/papers/camel/>
7. Wolfram S. A New Kind of Science. Wolfram Media, Inc., 2002.
8. Gardner M. The Fantastic Combinations of John Conway's New Solitaire Game "Life"// Scientific American. –1970. – №223. – P. 120–123.
9. Сайт по автоматному программированию и мотивации к творчеству. Кафедра «Технологии программирования» Санкт-Петербургского государственного университета информационных технологий, механики и оптики [Электронный ресурс]; ред. Шалыто А.А.; web-master Царев М. – Режим доступа: <http://is.ifmo.ru/> свободный. –Загл. с экрана. – Яз. рус., англ.
10. Хопкрофт Д., Мотвани Р., Ульман Д. Введение в теорию автоматов, языков и вычислений. – М.: Вильямс, 2002.
11. Шалыто А.А. Switch-технология. Алгоритмизация и программирование задач логического управления. – СПб: Наука, 1998. – 628 с.

СРАВНЕНИЕ ПЕРЕХОДНЫХ ХАРАКТЕРИСТИК СТАНДАРТНЫХ СИГНАЛОВ И ПОЛИНОМОВ УОЛША

С.А. Родинков

Научный руководитель – д.т.н., профессор В.М. Мусалимов

Данная статья содержит сравнение переходных характеристик различных стандартных сигналов. Рассмотрена возможность исследования системы с помощью новых методов на базе полиномов Уолша. Даны основные понятия о полиномах Уолша и выделены характерные по сравнению с синусоидальными волнами особенности.

Введение

Переходные характеристики представляют собой некоторую реакцию исходной системы на стандартный сигнал или воздействие, например, единичное ступенчатое воздействие, единичный импульс, синусоидальный сигнал. В данной статье показано, что такие сигналы, как полиномы Уолша, также можно рассматривать как некоторый инструмент для исследования различных систем.

Переходные характеристики стандартных сигналов

Рассмотрим, какие же реакция возникают на все вышеперечисленные стандартные сигналы. В качестве исходной функции, на которой мы будем изучать воздействие сигналов, выберем следующую:

$$y = \ddot{x} + 0.75\dot{x} + 100x.$$

В приложении Simulink программного пакета Matlab составим блок-схему, как показано на рис. 1.

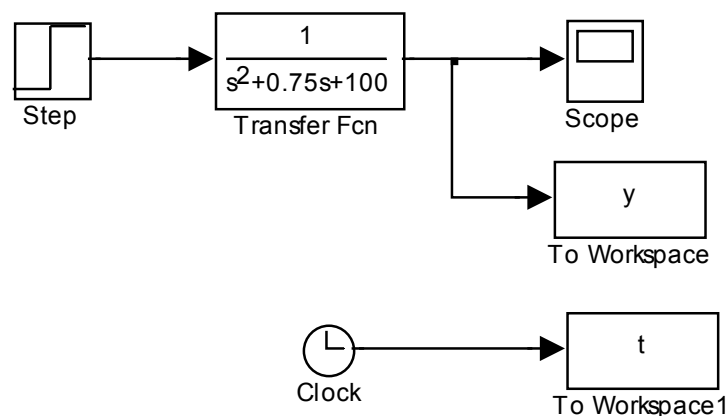


Рис. 1. Типовая блок-схема исследования реакции воздействия стандартного сигнала

На рис. 1 показана блок-схема воздействия единичным сигналом. Отклик показан на рис. 2. Другие виды воздействия указаны на рис. 3 и 4.

Также следует отметить, что в новых версиях приложения Simulink программного пакета Matlab появилась возможность работы с полиномами Уолша. На рис. 5 представлен отклик на несколько различных полиномов Уолша.

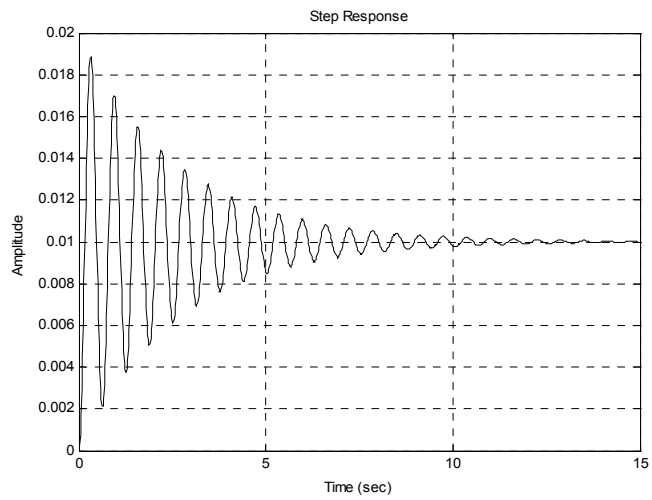


Рис. 2. Отклик на единичное воздействие на систему

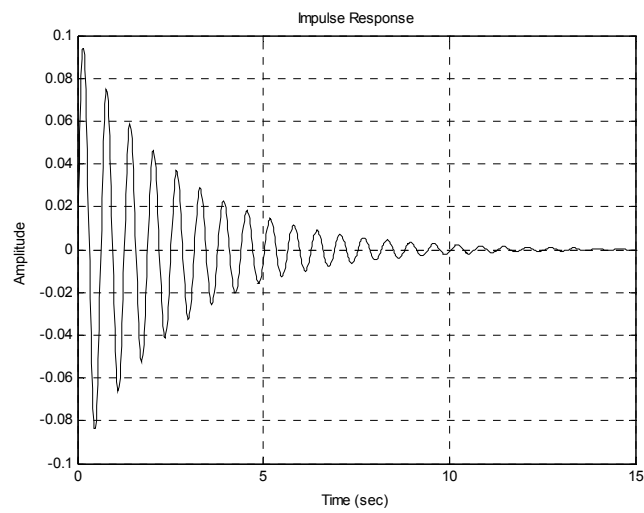


Рис. 3. Отклик на импульсное воздействие

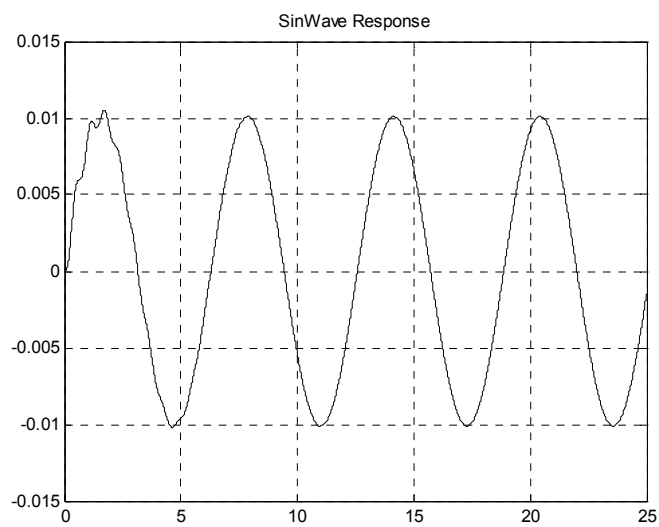


Рис. 4. Отклик на синусоидальное воздействие

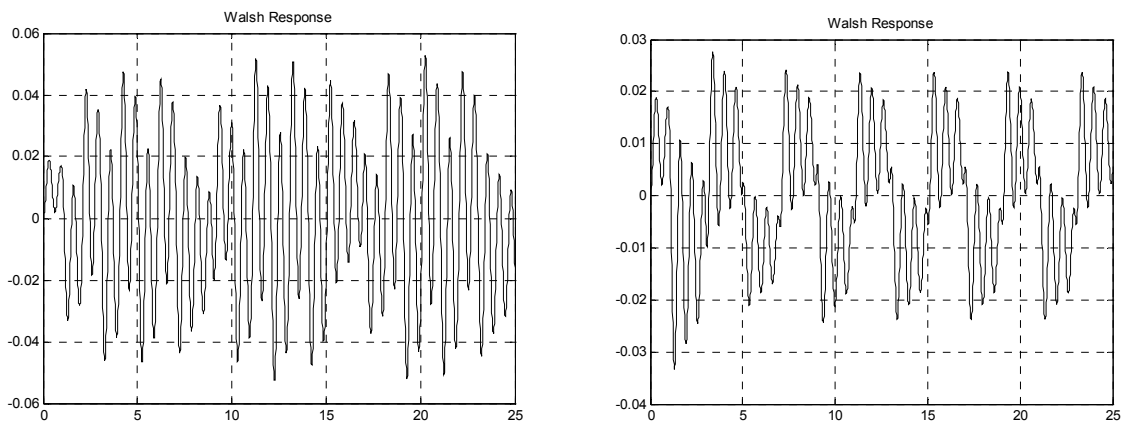


Рис. 5. Отклик на полиномы Уолша с различными коэффициентами

Отличие синусоидальных волн от волн Уолша и от других несинусоидальных волн

Все различия между волнами Уолша и синусоидальными волнами можно свести к следующим математическим утверждениям.

- Производная или интеграл от синусоидальной функции является той же синусоидальной функцией, которая сдвинута во времени и имеет другую амплитуду. С другой стороны, функции Уолша изменяются, если их проинтегрировать или проинтегрировать.
- Сумма синусоидальных функций с одной и той же частотой, но с различными амплитудами и фазами, равна синусоидальной функции с той же частотой. Сложение двух волн происходит иначе.
- Эффект Доплера проявляется двойко. Одиночная волна Уолша дает более сильный эффект Доплера, чем одиночная синусоидальная волна. С другой стороны, эффект Доплера у ортогональной последовательности Уолша слабее, чем у ортогональной последовательности синусоидальных волн. Чтобы преобразовать волну Уолша в другую, необходимо, чтобы относительная скорость удовлетворяла условию $|v/c| > 3/5$. Для синусоидальных волн никакой пороговой скорости для такого преобразования не существует.
- Изменение знака синусоидальных функций на обратный эквивалентно сдвигу их во времени. Для волн, которые не обладают симметрией по знаку и не являются полярно-симметричными, такой эквивалентности не существует.

Функции Уолша

В теории синусоидальных функций используются такие обозначения, как $\sin \omega t$, $\cos \omega t$ или $\exp(i\omega t)$. Экспоненциальные функции более удобны при ведении расчетов, а функции \sin и \cos подходят лучше для наблюдаемого представления. Подобным образом вводятся функции Уолша $cal(i, \theta)$ и $sal(i, \theta)$ по аналогии с $\sqrt{2} \cos(2\pi i\theta)$ и $\sqrt{2} \sin(2\pi i\theta)$. Число $2i$ характеризует число пересечений нуля на полуоткрытом интервале, например, $-\frac{1}{2} \leq \theta < \frac{1}{2}$. Преимущество обозначений такого рода состоит в том, что секвента i может быть легко измерена различного рода счетчиками. (Секвента – половина среднего числа пересечений нуля за единицу времени).

Однако в некоторых случаях намного удобнее использование другого представления функций, а именно, одной системы функций $wal(j, \theta)$. Аналогично данное обозна-

чение более удобно для расчетов, а не для измерений. Здесь параметр j дает число пересечений нуля на открытом интервале, например $-\frac{1}{2} < \theta < \frac{1}{2}$. Данное представление лучше всего позволяет увидеть самые важные свойства функций Уолша.

Также существуют и другие представления функций Уолша (рис. 6).

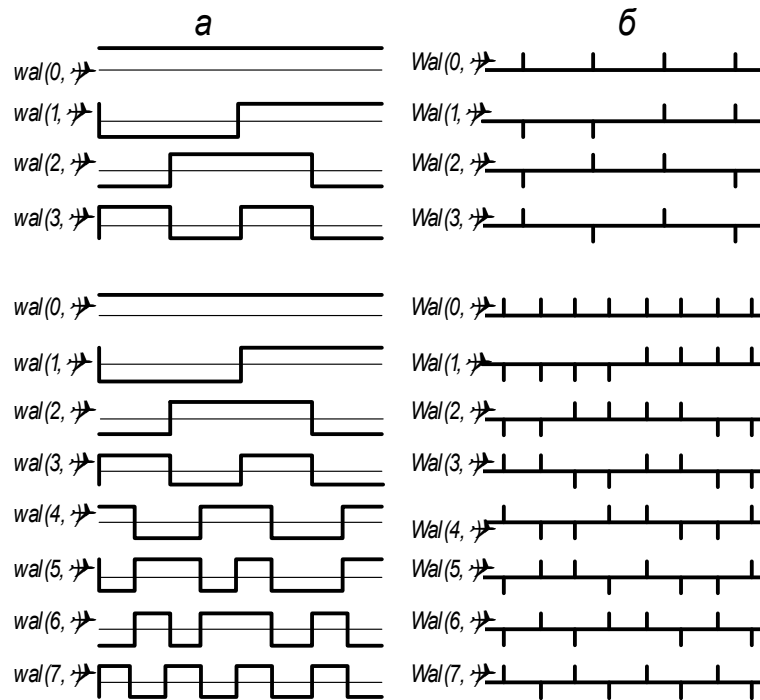


Рис. 6. Функции Уолша в двух представлениях:
 а) функции Уолша, всюду непрерывные и дифференцируемые,
 б) дискретные функции Уолша, представленные в виде отсчетных функций

На рис. 6, а, слева функции Уолша показаны в виде функций, однозначно определенных для всех вещественных значений θ , кроме точек, где функции имеют скачок. Это непрерывные функции Уолша с конечным числом разрывов. На рис. 6, б, справа показаны те же функции в виде отсчетов. С математической точки зрения это наилучшая форма представления функций Уолша. Отсчетные функции Уолша определены только там, где показаны значения $+1$ и -1 . Для технических целей иногда полезно определять эти функции так, чтобы они были равны нулю в промежутке между отсчетами. Функции $wal(j, \theta)$ и $Wal(j, \theta)$ имеют одинаковое число пересечений нуля.

Функции $wal(j, \theta)$, представленные в левой части рис. 6, а, являются взаимно ортогональными, но ортогональность в общем случае теряется, если эти функции сдвинуть по времени друг относительно друга. Замечательным свойством синусоидальных функций является то, что ортогональность любых периодических функций \sin или \cos с различными частотами не нарушается при временном сдвиге. Рассмотрим теперь отсчетные функции $Wal(j, \theta)$ в правой части рис. 6, б, которые тоже являются ортогональными; но их ортогональность не нарушается при временном сдвиге. Ортогональность может быть нарушена, если сдвиги целочисленно кратны расстоянию между отсчетами. Однако вероятность того, что это возникнет случайно, равна нулю, так как отсчеты являются бесконечно короткими, а расстояние между отсчетами конечно. Неинвариантность ортогональности к сдвигу не является уникальным свойством функций Уолша $Wal(j, \theta)$. Этим свойством обладают любые отсчетные функции. Непрерывную функцию можно рассматривать как отсчетную функцию, имеющую несчетное множество отсчетов. Однако в экспериментальных исследованиях может быть получено, на-

коплено или обработано только конечное число отсчетов. Поэтому математические положения, которые верны для несчетного или счетного множества отсчетов, но неверны для конечного числа отсчетов, не могут быть использованы при анализе данных экспериментов. Примером такого достаточно абстрактного положения является уникальное свойство синусоидальных функций сохранять ортогональность при сдвиге.

Заключение

На основании вышеизложенного можно говорить об универсальности использования функций Уолша и возможности использования их для обработки разного рода сигналов, в частности, акустических сигналов. В дальнейшем, возможно, на кафедре мехатроники будет предпринята попытка использования функций Уолша для конструирования приемо-передающего акустического устройства.

Литература

1. Бесекерский В.А., Попов Е.П. Теория систем автоматического управления. – СПб: Профессия, 2004. – 752 с.
2. Бишоп Р.Х., Дорф Р.К. Современные системы управления: Пер. с англ. – М.: Лаборатория базовых знаний, 2004 – 832 с.
3. Гудвин Г.К., Греббе С.Ф., Сальгадо М.Э. Проектирование систем управления: Пер. с англ. – М.: Бином, 2004. – 911 с.
4. Дьяконов В.П. Вейвлеты. От теории к практике. Изд. 2-е, перераб. и доп. – М.: СОЛОН-Пресс, 2004. – 400 с.
5. Хармут Х.Ф. Теория секвентного анализа. Пер. с англ. – М.: Мир, 1980. – 576 с.

ЗАКОН ДВИЖЕНИЯ МЕХАНИЗМА ПЛЕТЕНИЯ СПИРАЛЬНО-АНИЗОТРОПНЫХ НИТЕЙ

А.Д. Перечесова, П.А. Сергушин

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В работе рассмотрена задача определения параметров движения основного узла механизма для изготовления спирально-анизотропных нитей. Применен метод планов. Полученный результат будет использован для усовершенствования формы деталей устройства с целью увеличения быстродействия и улучшения качества получаемого продукта (спирально-анизотропных нитей (САН) в виде «косы»).

Введение

Существует ряд устройств, в чувствительных элементах (ЧЭ) которых используются нити, подвесы или растяжки. Это торсионные приборы, крутильные весы, струнные акселерометры, сейсмоприемники, магнитовариометры и т.д.

Известно, что к ЧЭ предъявляются следующие требования: линейность и однозначность статической характеристики, высокая чувствительность (крутизна) и разрешающая способность; стабильность характеристик во времени, отсутствие влияния нагрузки на статические характеристики, минимальная инерционность; минимальное влияние внешних факторов (температуры, вибраций и т. д.); устойчивость к химическим воздействиям; простота и технологичность конструкции; взаимозаменяемость (повторяемость характеристик); удобство монтажа и обслуживания. Выбор типа ЧЭ определяется требованиями по точности системы, ее назначением и условиями эксплуатации [1].

Для достижения этих требований предложено использовать достижения теории спирально-анизотропного тела (САТ) и изготавливать ЧЭ в виде косы из филаментов арамидной нити.

В работе исследуется плоский рычажный механизм, применяемый в качестве рабочего узла механизма для изготовления подвесов чувствительных элементов измерительных приборов. Поставлена задача определения параметров движения его звеньев, в частности, кулачковой стрелки, перемещающей пряжи сплетаемого изделия по заданному закону. Для этого применены стандартные методы – план положений, скоростей и ускорений, рассчитан план сил. Полученный результат планируется использовать для усовершенствования формы деталей устройства с целью увеличения быстродействия и улучшения качества получаемого продукта.

№ п/п	Наименование показателя	Значение
1	Удельная разрывная нагрузка нити, сН/текс, не менее	210
2	Коэффициент вариации по разрывной нагрузке, %, не более	10
3	Относительное отклонение фактической линейной плотности от номинальной, %, не более	+5,0 –3,0
4	Удлинение нити при разрыве, %, не менее	2,6
5	Число кручений на 1м нити, кр/м	50+10; 100+10
6	Массовая доля замасливателя, %	1,0 – 2,0
7	Модуль упругости, ГПа, не менее	135
8	Число филаментов	300 (600)

Таблица 1. Физико-механические свойства арамидных нитей Русар 100 текс

В качестве сырья для изготовления САН предложено использовать филаменты арамидных нитей Русар производства ОАО «КАМЕНСКВОЛОКНО». Синтетические высокомодульные нити Русар относятся к классу параарамидных волокон.

В табл. 1 приведены физико-механические свойства арамидных нитей Русар 100 текс, из которых изготавливают САН в СПбФ ИЗМИРАН [2].

Принцип работы устройства для изготовления САН

Предлагается изготавливать ЧЭ с помощью устройства для изготовления САН из трех прядей. Используется по одной мононити в пряди, диаметр каждой нити не превышает 2 мкм, максимальный эквивалент диаметра косы при этом – 7 мкм, плотность плетения – 5–10 узлов/мм (в зависимости от натяжения прядей и угла плетения), длина получаемого изделия не более 100 мм. Производительность устройства в среднем – 100 мм/час (учитывая подготовительные операции).

Внешний вид и кинематическая схема устройства приведена на рис. 1.

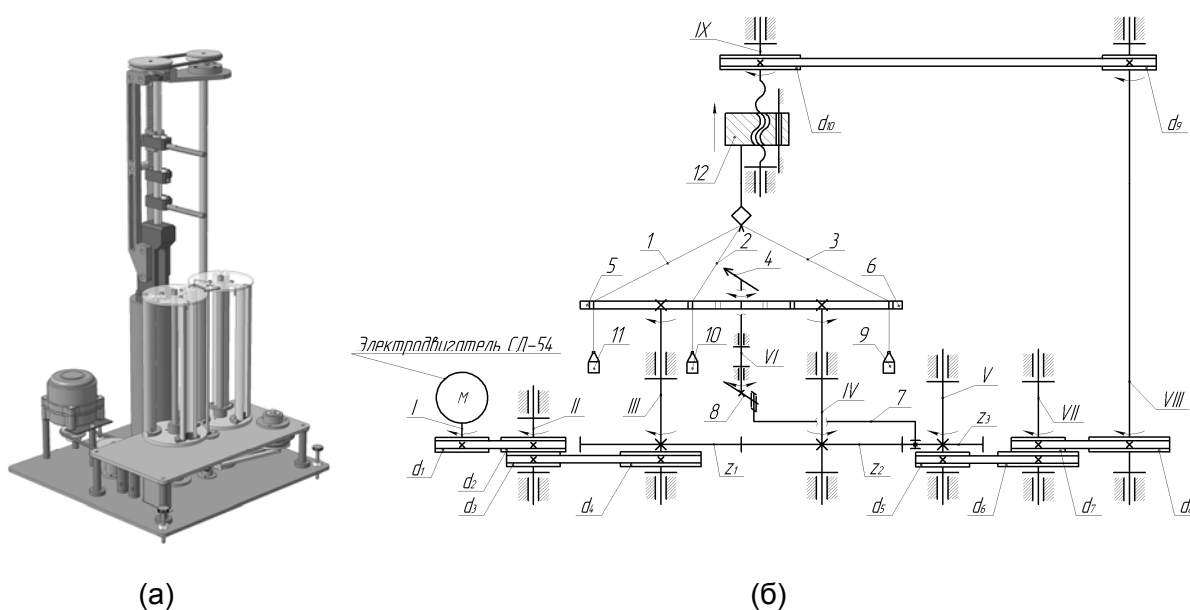


Рис. 1. Внешний вид устройства для изготовления САН (а); кинематическая схема (б):
 1–3 – пряди, 4 – стрелка, 5, 6 – диски, 7 – шатун, 8 – рычаг стрелки,
 9..11 – грузы, 12 – гайка, I..IX – валы, $z_1..z_3$ – зубчатые колеса, $d_1..d_{10}$ – шкивы

Процесс плетения реализован следующим образом: при помощи электродвигателя М через промежуточное передаточное устройство (d_1-d_4) приводятся в движение два диска 5 и 6, в пазы которых пропущены три пряди (1–3) будущей косы. При помощи зубчатых колес z_1 и z_2 реализовано вращение этих дисков во взаимно-встречном направлении. Синхронно с вращением дисков через кривошипно-шатунный механизм привода стрелки 4 ($z_3, 7, 8$) реализуется ее возвратно-вращательное движение. Таким образом, стрелка перемещает пряди из пазов одного диска в свободные пазы другого диска. Механизм подачи ($d_1..d_{10}, IX, 12$) осуществляет равномерный подъем узла плетения по мере скрещивания прядей механизмом плетения.

На рис. 2 изображен эскиз исполнительного механизма, схема одного цикла работы устройства и вид получаемой САН, соответственно [3].

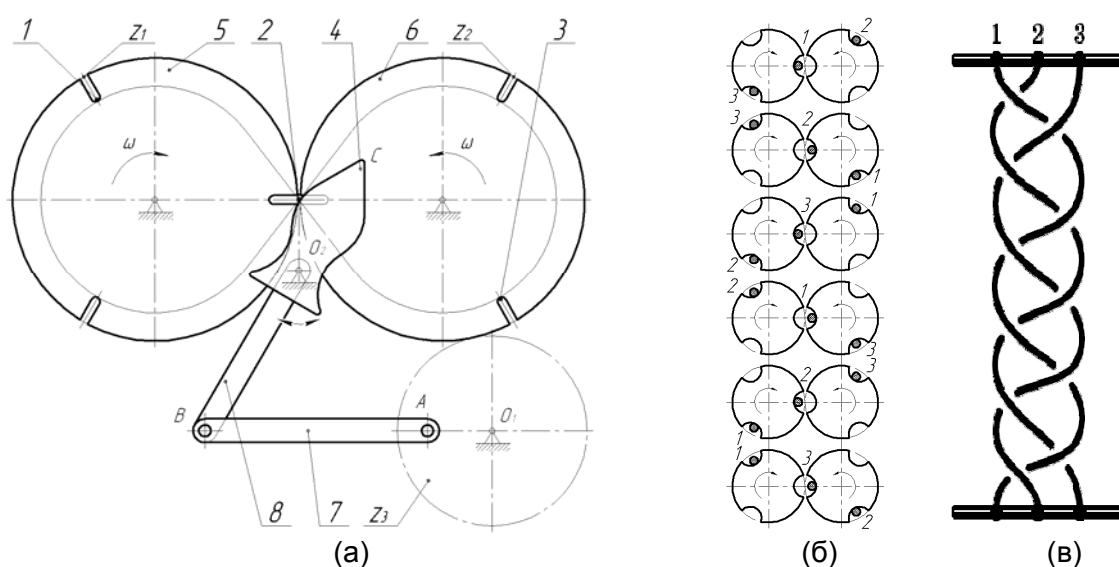


Рис. 2. Исполнительный механизм устройства для изготовления САН (а), схема одного цикла его работы (б) и результат работы (косы) (в): 1..3 – пряжи, 4 – стрелка, 5, 6 – диски, 7 – шатун, 8 – рычаг стрелки, $z_1 .. z_3$ – зубчатые колеса

В табл. 2 приведены характеристики сплетенных САН. Производительность прибора – в среднем 100 мм/час (учитывая подготовительные операции). Для усовершенствования формы деталей устройства с целью увеличения быстродействия и улучшения качества получаемого продукта необходимо определить характер движения основного звена – кулачковой стрелки, перемещающей пряжи сплетаемого изделия по заданному закону.

№ п/п	Наименование показателя	Значение
1	длина, мм, не более	100
2	минимальный эквивалент диаметра косы, мкм, не более	7
3	плотность плетения, узлов/мм	5..10

Таблица 2. Характеристики сплетенных САН

Кинематика механизма перемещения нитей

Передаточные отношения

$$d_1 = 0,035 \text{ м}; \quad d_{z1} = 0,096 \text{ м};$$

$$d_2 = 0,050 \text{ м}; \quad d_{z2} = 0,096 \text{ м};$$

$$d_3 = 0,040 \text{ м}; \quad d_{z3} = 0,064 \text{ м};$$

$$d_4 = 0,067 \text{ м};$$

$$i_{n, n+1} = \frac{d_n}{d_{n+1}};$$

$$i_{I, II} = 0,70; \quad i_{III, IV} = 1,00;$$

$$i_{II, III} = 0,60; \quad i_{IV, V} = 1,50;$$

$$i_{I-V} = i_{I, II} \cdot i_{II, III} \cdot i_{III, IV} \cdot i_{IV, V};$$

$$i_{I-V} = 0,63.$$

Угловые скорости

В качестве привода устройства выбран электродвигатель с редуктором СД-54:

$$\omega (I) = 2,24 \frac{\text{об}}{\text{мин}}; = 0,23 \frac{\text{рад}}{\text{с}};$$

$$\omega (V) = \omega (I) \cdot i_{I-V};$$

$$\omega (V) = 0,15 \frac{\text{рад}}{\text{с}}.$$

На рис. 3. изображен план положений исполнительного механизма.

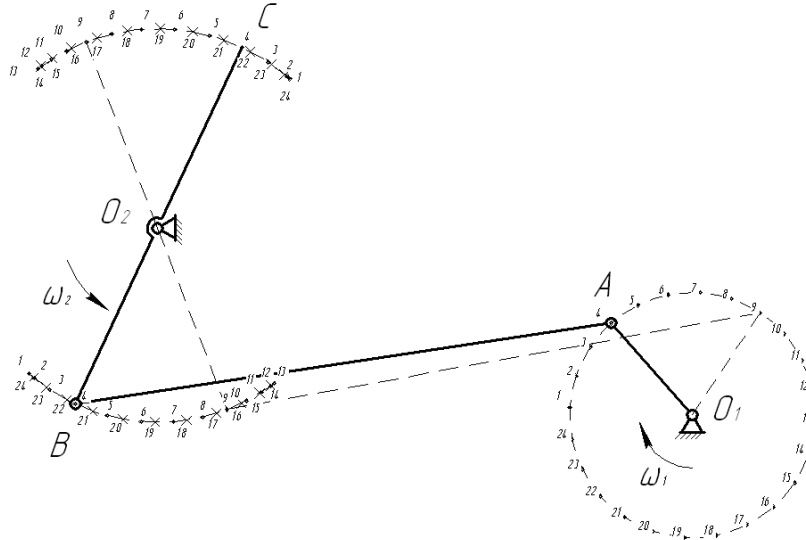


Рис. 3. План положений исполнительного механизма

$$O_1A = 0,024\text{м}; AB = 0,106\text{м}; BO_2 = 0,038\text{м}; O_2C = 0,039\text{м}.$$

На рис. 4 приведены планы скоростей исполнительного механизма.

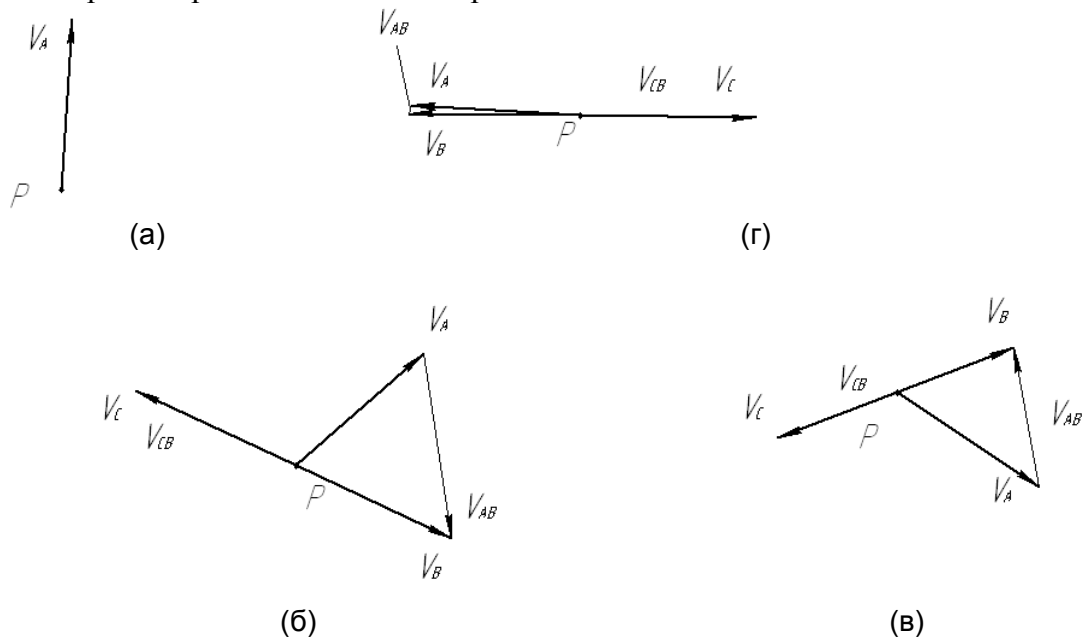


Рис. 4. План скоростей исполнительного механизма

(а) – в положении 1; (б) – в положении 4; (в) – в положении 9; (г) – в положении 19

Здесь P – полюс, а масштаб $K_V = 10^4 \frac{\text{м}}{\text{с}} \cdot \text{мм черт}$.

На рис. 5 изображен план ускорений исполнительного механизма.

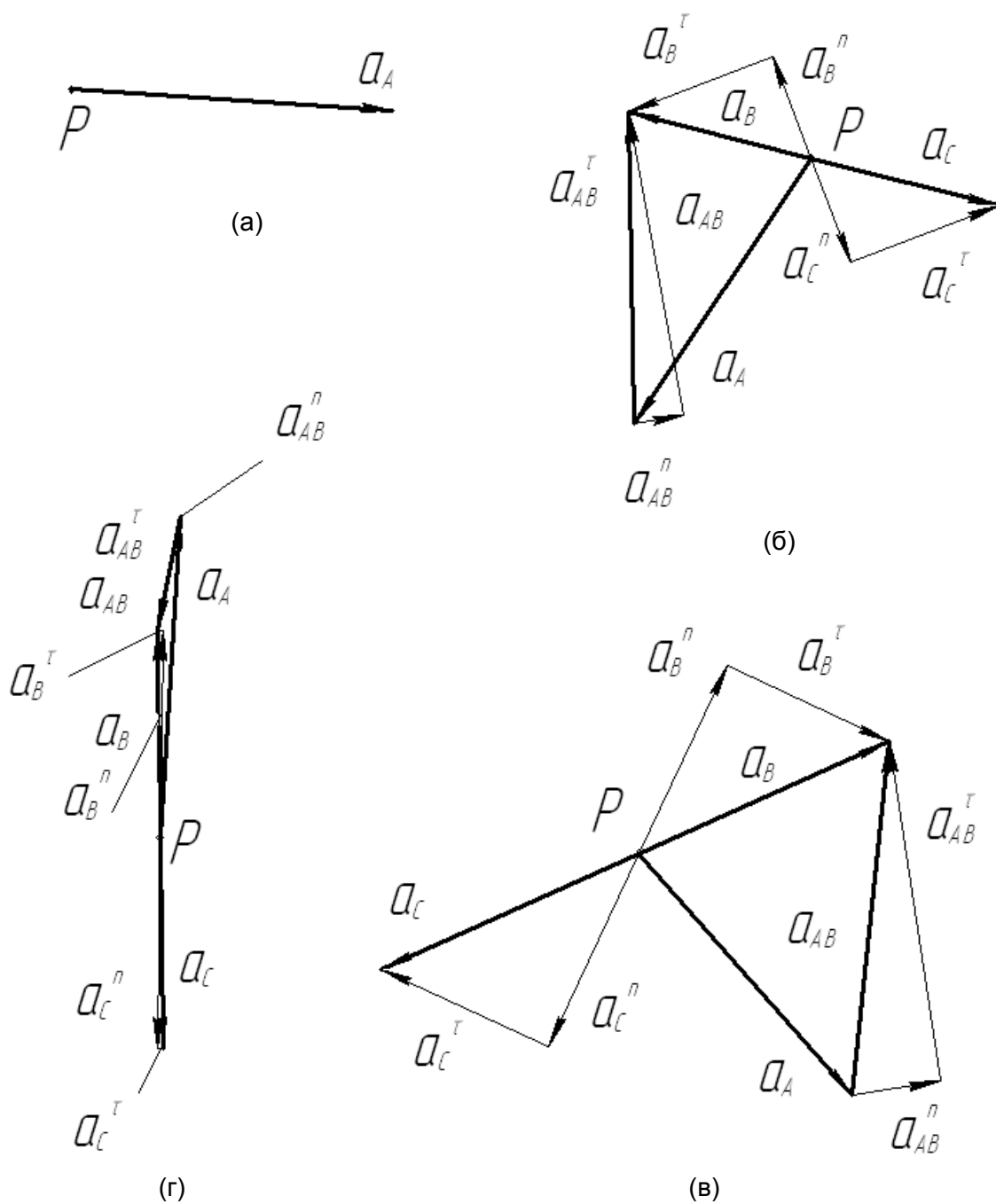


Рис. 5. План ускорений исполнительного механизма: (а) – в положении 1; (б) – в положении 4; (в) – в положении 9; (г) – в положении 19

Здесь масштаб $K_a = 10^5 \frac{\text{м/с}^2}{\text{мм черт}}$.

Динамическая модель исполнительного механизма

Считаем, что звено O_1A – звено привода. Динамическую модель механизма (рис. 6) представим в виде звена, которое вращается вокруг оси O_1 с угловой скоростью ω_1 , и к нему приложен движущий момент M_1 . К звену приложена приведенная сила сопротивления P_{np} , перпендикулярная радиусу O_1A , и приведенная масса звена

m_{np} сосредоточена в точке A . На рис. 6 изображена динамическая модель исполнительного механизма.

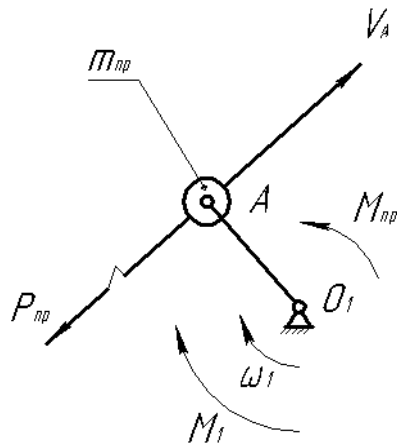


Рис. 6. Динамическая модель исполнительного механизма

Основные действующие силы

$$F = a \cdot m$$

$$m_{O_1A} = 122.07 \cdot 10^{-3} \text{ кг};$$

$$m_{AB} = 10.08 \cdot 10^{-3} \text{ кг};$$

$$m_{BO_2} = 10.19 \cdot 10^{-3} \text{ кг};$$

$$m_{O_2C} = 3.13 \cdot 10^{-3} \text{ кг}.$$

На рис. 7 показаны основные действующие силы (положение 4).

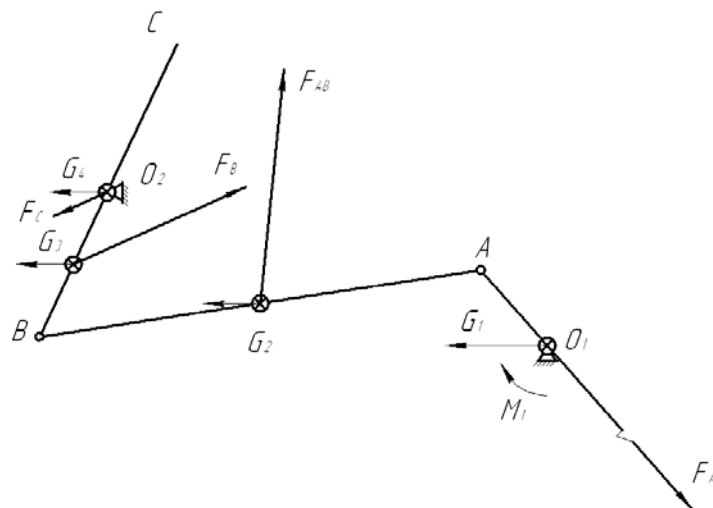


Рис. 7. Основные действующие силы (положение 4)

$$F_{O_1A} = 633.54 \cdot 10^{-7} \text{ Н};$$

$$F_{AB} = 55.94 \cdot 10^{-7} \text{ Н};$$

$$F_{BO_2} = 44.94 \cdot 10^{-7} \text{ Н};$$

$$F_{O_2C} = 14.18 \cdot 10^{-7} \text{ Н}.$$

Заключение

В работе исследован характер движения звеньев механизма для плетения САН, получены: план положений, скоростей, ускорений и сил для основных его звеньев. На сегодняшний день САН применяются в качестве подвеса ЧЭ магнитостатического вариометра (МВ) – прибора, предназначенного для проведения магнитных измерений.

Изготовленные САН по сравнению с существующими на рынке нитями обладают большей разрывной прочностью, меньшим коэффициентом линейного расширения при том же диаметре поперечного сечения. Кроме того, МВ, изготовленные на основе САН, обладают более высокой чувствительностью (0.1 нТл), менее чувствительны к климатическим факторам.

Литература

1. Ахромеев Ж.П. Робототехника и гибкие автоматизированные производства. / В 9-ти кн. Кн. 2. Приводы робототехнических систем. – 1986. – 175 с.
2. Параарамидные волокна и нити [Электронный ресурс]: официальный сайт ОАО «Каменскволокно» – Режим доступа: <http://www.aramid.ru>, свободный.
3. Копытенко Ю.А., Леваненко В.А., Петрищев М.С., Сергушин П.А. Устройство для изготовления спирально-анизотопных нитей // Программа и тезисы докладов Восьмой сессии Международной научной школы «Фундаментальные и прикладные проблемы надежности и диагностики машин и механизмов» VPB-07, Санкт-Петербург, 22–27 октября 2007 г. – СПб: ИПМаш РАН, 2007. – С. 112–114.

КОНСТРУКТИВНАЯ РЕАЛИЗАЦИЯ ДИАМЕТРАЛЬНОЙ СХЕМЫ ГИРОИНКЛИНОМЕТРА С ДОПОЛНИТЕЛЬНЫМ РАЗВОРОТОМ КОРПУСА СКВАЖИННОГО ПРИБОРА

Д.А. Соколов

Научный руководитель – к.т.н., доцент Я.И. Биндер

Анализируются характеристики так называемой диаметральной схемы инклинометра при использовании дополнительных разворотов корпуса вокруг продольной оси. Приводятся алгоритмы определения навигационных параметров. Предлагается конструктивное решение для реализации схемы.

Введение

В процессе разработки серии универсальных гироскопических инклинометров, предназначенных для работы в скважинах с малым диаметром и произвольной ориентацией, был предложен и проанализирован целый ряд новых технических решений бесплатформенных схем. Напомним их основные особенности. В так называемой «продольной» схеме ось кинетического момента двухосного ДУС совпадает с продольной осью скважинного прибора или, что то же самое, оси чувствительности двухосного измерителя, в том числе и в случае отсутствия механического носителя кинетического момента, находятся в поперечной плоскости. Данная схема позволяет производить промерочные работы как в точечном (во время остановок скважинного прибора), так и – что приобрело особую актуальность в последние годы – в непрерывном режимах.

В первом случае в процессе отыскания плоскости меридиана места используется хорошо известная процедура гироскопирования. Исследование продольной схемы подтвердило, что она имеет принципиальное ограничение, затрудняющее, а в ряде случаев делающее невозможным ее применение в точечном режиме при измерении угловых параметров, расположенных близко к линии «запад-восток» [1].

В непрерывном режиме с помощью уравнений Пуассона при неких начальных условиях, заданных извне или полученных при помощи того же компасирования, интегрируются показания двухосного ДУС. Такой режим получил также название бесплатформенного гироскопа направления (ГН).

Следующим этапом эволюции явилась «поперечная» схема. В ней главная ось двухосного датчика угловой скорости ориентирована в поперечной плоскости скважинного прибора, что позволяет в рамках схемы с одним ДУС осуществить компасирование при любой ориентации скважины (так называемая адаптивность к траектории) [2]. Однако это техническое решение принципиально не сочетается с режимом бесплатформенного ГН, что резко ограничивает востребованность ГИ, построенного по поперечной схеме. В то же время следует отметить принципиальную возможность развития на базе этой схемы режима физической стабилизации ГИ относительно оси скважинного прибора (СП), т.е. создания ГН на базе одноосного гиросtabilизатора.

Так называемая «диаметральная» схема, предложенная некоторое время назад [3], представляется наиболее совершенной в ряду технических решений для бесплатформенных инклинометров. Она предусматривает разворот корпуса гироскопа вокруг поперечной оси скважинного прибора. При этом вектор кинетического момента перпендикулярен оси разворота и находится в диаметральной плоскости скважинного прибора. Эта схема совмещает в себе достоинства всех предыдущих, позволяя создать на основе одного двухосного ДУС полностью универсальный ГИ, способный эффективно работать как в точечном, так и в непрерывном режимах.

С того времени, как была предложена диаметральная схема (т.е. буквально в последние 2 года), особую актуальность, как уже было сказано, приобрели разработка и исследование возможностей применения инклинометра в непрерывном режиме. Это

связано как с желанием заказчиков получать подробные данные о «высокочастотном» пространственном спектре траектории скважин, так и с общей тенденцией повышения производительности съемки. Последнее обусловлено тем, что время съемки скважины является самостоятельным и, притом, весьма существенным технико-экономическим фактором, в значительной степени влияющим на выбор используемой при измерениях инклинометрической системы.

Алгоритмы определения навигационных параметров

Кратко коснемся основных особенностей работы схемы в точечном режиме.

Введем следующие системы координат (рис. 1): ENh – горизонтная с географической ориентацией осей (соответственно на восток, север и по вертикали места); $ox_0y_0z_0$ – связанная с корпусом гироинклинометра, ось oz_0 которой ориентирована вдоль продольной оси, а оси ox_0, oy_0 лежат в плоскости поперечного сечения СП. Положение системы координат $ox_0y_0z_0$ относительно географического трехгранника ENh определяется тремя углами – A, θ и ψ .

Для анализа возможностей диаметральной схемы введем систему координат, связанную с корпусом ДУС – $oX_\Gamma Y_\Gamma Z_\Gamma$, ось oZ_Γ которой совпадает с вектором кинетического момента, а оси oX_Γ, oY_Γ – с измерительными осями ДУС.

Положение системы координат $oX_\Gamma Y_\Gamma Z_\Gamma$ относительно $ox_0y_0z_0$ определяется углом χ (см. рис. 1) разворота вектора кинетического момента гироскопа относительно оси скважинного прибора. Варьируя этот угол, мы получаем возможность перейти от рассмотрения обобщенной схемы ГИ к конкретным вариантам ее реализации. В частности, продольной и поперечной схемам соответствуют значения $\chi = 0^\circ$ и 90° , соответственно.

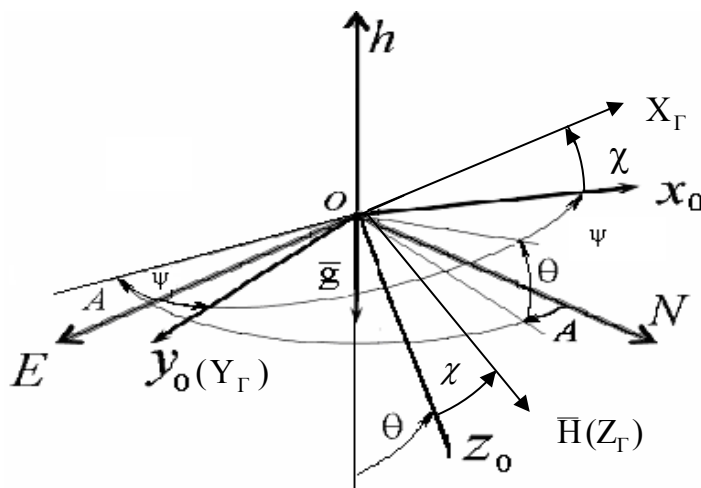


Рис. 1. Системы координат

Рассмотрим выражения для сигналов измерительных каналов ДУС в режиме компасирования, приняв во внимание выражения для проекций кажущегося ускорения на измерительные оси ДУС, а также на оси системы координат, связанной с корпусом гироинклинометра (измеряемые блоком акселерометров).

$$\begin{aligned} n_{X_\Gamma} &= n_{x_0} \cos \chi - n_{z_0} \sin \chi, \\ n_{Y_\Gamma} &= n_{y_0}. \end{aligned} \tag{1}$$

причем

$$n_{x0} = -g \sin \theta \cos \psi, \quad n_{y0} = -g \sin \theta \sin \psi, \quad n_{z0} = g \cos \theta.$$

$$\begin{aligned} \omega_{X_r} = & \Omega \sin \varphi (\sin \theta \cos \psi \cos \chi + \cos \theta \sin \chi) + \\ & + \Omega \cos \varphi \cdot [\cos A (\cos \theta \cos \psi \cos \chi - \sin \theta \sin \chi) + \sin A \sin \psi \cos \chi] + \tau_{X_r} - \end{aligned} \quad (2)$$

$$\begin{aligned} \omega_{Y_r} = & \Omega \sin \varphi \sin \theta \sin \psi - \Omega \cos \varphi \cdot (\sin A \cos \psi - \cos A \cos \theta \sin \psi) + \tau_{Y_r} - \\ & - \rho_{Y_r} \sin \theta \sin \psi - \varepsilon_{Y_r} (\sin \theta \cos \psi \cos \chi + \cos \theta \sin \chi). \end{aligned} \quad (3)$$

Уравнения (2) и (3) целесообразно рассматривать относительно неизвестных $\sin A$, $\cos A$, так как только одновременное вычисление этих тригонометрических функций позволяет определить азимут при любом его значении. Определитель Δ системы уравнений имеет вид:

$$\Delta = -\cos \theta \cos \chi + \sin \theta \sin \chi \cos \psi. \quad (4)$$

Его обнуления можно избежать, если соблюдать условие $\operatorname{tg} \chi_0 \neq \frac{\operatorname{ctg} \theta}{\cos \psi}$.

Адаптивности же к траектории, т.е. определения азимута при любом его значении с одной и той же точностью, можно добиться при $\Delta = \max$, когда выполняется условие

$$\operatorname{tg} \chi_0 = -\operatorname{tg} \theta \cos \psi. \quad (5)$$

В работе [3] предлагалась и исследовалась возможность реализации диаметральной схемы и доказывалось наличие у этой схемы возможности инвариантного компасирования (независимости ошибки определения азимута от пусковых погрешностей гироскопа) – важнейшего свойства гироскопа, работающего практически в условиях старт-стоп. Для достижения одинаковой эффективности (равной точности) инвариантного компасирования при любых траекториях скважин в диаметральной схеме необходимо управление углом ψ . Дальнейший анализ показывает, что максимума определитель достигает независимо от параметров траектории, при $\psi = 0, 180^\circ$:

$$|\Delta \max| = \sqrt{1 - \sin^2 \psi \cdot \cos^2 \theta}. \quad (6)$$

При этом достигается свойство инвариантного компасирования.

Проиллюстрируем сформулированный вывод. Действительно, при использовании разворотов по углам χ и ψ достаточно двух положений ДУС для проведения инвариантного компасирования. В первом положении $\psi = 0^\circ, \chi = -\theta$, а во втором $\psi = 180^\circ, \chi = \theta$.

Проанализируем выражения для сигналов измерительных каналов ДУС, с учетом разворотов в эти положения.

В первом положении

$$\omega_{X1_r} = \Omega \cos \varphi \cdot \cos A + \tau_{X_r}, \quad (7)$$

$$\omega_{Y1_r} = -\Omega \cos \varphi \cdot \sin A + \tau_{Y_r}. \quad (8)$$

Во втором положении

$$\omega_{X2_r} = -\Omega \cos \varphi \cdot \cos A + \tau_{X_r}, \quad (9)$$

$$\omega_{Y2_r} = \Omega \cos \varphi \cdot \sin A + \tau_{Y_r}. \quad (10)$$

Как видно, неопределенности дрейфов, пропорциональные проекциям ускорения – осевой дебаланс ротора гироскопа и квадратурные составляющие скорости ухода, являющиеся для механического носителя вектора кинетического момента основными возмущающими факторами – не оказывают влияние на точность компасирования.

Устанавливая ДУС последовательно в положения $\psi = 0^\circ, \chi = -\theta$ и $\psi = 180^\circ, \chi = \theta$, мы можем исключить влияние прочих составляющих дрейфов гироскопа, вычисляя

$$\cos A = \frac{\omega_{X1r} - \omega_{X2r}}{2\Omega \cos \varphi}, \quad (11)$$

$$\sin A = \frac{\omega_{Y1r} - \omega_{Y2r}}{2\Omega \cos \varphi}. \quad (12)$$

Проводимый анализ процедуры компасирования, базирующейся на принудительных разворотах корпусов ДУС и скважинного прибора, был бы неполон без соответствующей оценки влияния точности этих разворотов. Опуская промежуточные преобразования, приведем выражение для погрешности определения азимута в зависимости от неопределенности углов χ и ψ при проведении выставки:

$$\Delta A = \Delta \psi \cos \varphi + \operatorname{tg} \varphi (\Delta \psi \sin \theta \cos A + \Delta \chi \sin A). \quad (13)$$

Полученные линейные соотношения позволяют судить о заведомой выполнимости требований к точности углов, обеспечиваемых, прежде всего, акселерометрами ГИ работающими по прямому назначению.

Непрерывный режим

Анализ использования предлагаемой схемы ГИ в непрерывном режиме в принципе не является предметом исследования в данной работе. Однако, принимая во внимание максимальную универсализацию использования прибора как одну из основных предпосылок к его исследованию и созданию, необходимо кратко затронуть новые возможности, возникающие при управлении дополнительным разворотом корпуса.

Во-первых, как показал анализ практических результатов проведения непрерывной съемки в скважинах различного типа, наиболее существенным внешним фактором, во многом определяющим точностные характеристики этого режима, является так называемый «канатный эффект», т.е. вращательное движение кабеля, обусловленное, с одной стороны, состоянием его упругой деформации, а с другой – вертикальным линейным удлинением (укорочением) его свободного конца в процессе спускоподъемных операций (СПО). Считать воздействие указанного фактора на работу ГИ в непрерывном режиме однозначно отрицательным было бы неправильно: в частности, это вращение может быть использовано в качестве мощного модулятора ряда инструментальных погрешностей ГИ. Однако параметры, да и само наличие этого вращения, весьма нерегулярны и зависят от интенсивности изгиба и кручения скважины, скорости проведения и интервалов непрерывности СПО и т.д. В силу этого возможным образом «скорректировать» естественное крутильное движение кабеля принудительным движением корпуса представляется вполне целесообразной. Например, на состояние покоя или нерегулярное слабое вращение кабеля в начальной фазе интервалов движения может быть наложено принудительное однонаправленное или реверсивное вращение СП. То же самое относится к движению скважинного прибора при больших зенитных углах (θ) когда скорость СПО резко падает. В некоторых режимах бесплатформенного ГИ вращение кабеля, напротив, требуется скомпенсировать или, во всяком случае, ослабить.

Во-вторых, последнее замечание вызывает вполне естественный вопрос: нельзя ли от ослабления, а тем более компенсации, влияния вращения кабеля перейти к стабилизации вращательного движения СП или, во всяком случае, той его части, которая содержит ЧЭ, т.е., используя структурную гибкость диаметральной схемы, совместить бесплатформенный ГИ с одноосным гиросtabilизатором? Целесообразность использования последнего при непрерывной съемке, в качестве альтернативы режиму ГИ, связана с тем, что при прохождении вертикальных (и близких к ним) участков скважин ГИ не в состоянии работать в режиме гироазимута, т.е. сохранить точность, пусть даже очень хорошей начальной выставки в устье скважины. Это не сказывается существенно на точности построения координат вертикальной части ствола, однако всецело определяет

погрешность начальной выставки при переходе к наклонному участку траектории. Что же касается аппаратной реализации одноосного гиостабилизатора, то, как следует из дальнейшего изложения, все основные технические решения для этого уже определены необходимостью реализации диаметральной схемы с дополнительным поворотом как таковой.

Конструктивные требования

Предполагаемые к реализации конструктивно-технические решения базируются:

- на сочетании габаритов используемого датчика угловой скорости и принципиально-го для всей разработанной серии универсальных инклинометров (УГИ) ограничения диаметра корпуса СП на уровне 42 мм;
- на реализации вращения блока БЧЭ относительно основной части скважинного прибора с помощью принципа так называемого «мокрого» вала;
- на опыте герметизации быстро вращающихся валов в скважинах с помощью цилиндрических уплотнений высокого давления. Здесь следует отметить, что линейные скорости вращательного движения по углу ψ на один-два порядка ниже допускаемых для конструктивных элементов такого рода, что позволяет оптимистично отнестись к такому нестандартному для гироскопов решению, как «мокрый» вал;
- на реализации токоподвода от БЧЭ к основной части прибора с помощью полого вала;
- на свободном размещении в основной части прибора достаточно мощного исполнительного элемента и обеспечивающих элементов привода.

На рис. 2 представлена кинематическая схема с учетом конструктивной компоновки диаметральной схемы с возможностью дополнительного разворота.

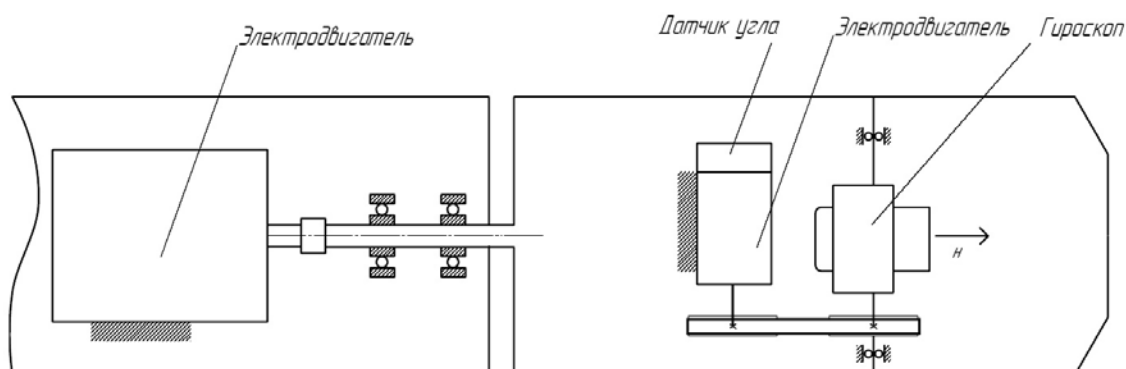


Рис. 2. Кинематическая схема

Заключение

Рассмотренные принципы управления разворотом корпуса скважинного прибора для ГИ построенного по диаметральной схеме обеспечивают:

- реализацию точечного компасирования, инвариантного к пусковым погрешностям ДУС при любых траекториях скважин;
- дальнейшее снижение погрешностей «инвариантного» компасирования обусловленных неопределенностью моделей дрейфов ДУС;
- осуществление при непрерывной инклинометрической съемке в режиме бесплатформенного ГН принудительного вращения корпуса ГИ вокруг оси скважины с целью достижения компенсационных или модуляционных эффектов по отношению к основным возмущающим факторам;

- возможность реализации одноосного индикаторного гиросtabilизатора в целях осуществления непрерывной инклинометрической съемки скважин при большой протяженности их вертикальных стволов.

Проведен предварительный анализ тактико-технических требований к конструкции и отдельным ее элементам, подтверждающий принципиальную реализуемость ГИ, выполненного по диаметральной схеме с дополнительным приводом управления по углу ψ . Есть основания полагать, что в классе наиболее жестких геометрических, виброударных и термобарических характеристик скважинных приборов, диктуемых задачами подземной навигации, рассматриваемая схема ГИ может оказаться оптимальной с точки зрения целого ряда принципиально противоречивых технических требований.

Литература

1. Биндер Я.И. Аналитическое компасирование в инклинометрии скважин малого диаметра. // Гироскопия и навигация. – 2003. – № 2 (41). – С. 38–46.
2. Биндер Я.И., Падерина Т.В. Бесплатформенный гиросинклинометр с ориентацией главной оси двумерного датчика угловой скорости в плоскости поперечного сечения скважины. // Гироскопия и навигация. – 2004. – № 1 (44). – С. 5–16.
3. Биндер Я.И. Универсальный гиросинклинометр с ориентацией главной оси двухосного датчика угловой скорости в диаметральной плоскости скважины. // Гироскопия и навигация. – 2005. – № 4 (51). – С. 23–32.

ВЛИЯНИЕ ТЕМПЕРАТУРЫ НА ЧАСТОТНЫЕ ХАРАКТЕРИСТИКИ МИКРОАКУСТИЧЕСКИХ ДАТЧИКОВ НА ПОВЕРХНОСТНЫХ И ОБЪЕМНЫХ АКУСТИЧЕСКИХ ВОЛНАХ

А.Л. Ткачев

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В статье рассматривается влияние температуры на коэффициент задержки и частотные характеристики микроакустических датчиков на поверхностных и объемных акустических волнах, а также такой эффект, как температурный удар, и его влияние на характеристики устройства.

Введение

В современных системах остро стоят вопросы миниатюризации, качества, сохранения и улучшения показателей и цены, что является причиной постоянных разработок в области уменьшения и улучшения компонентов, используемых в этих системах. Микроакустические датчики на поверхностных и объемных акустических волнах представляют собой один из видов таких компонентов.

Типичные акустические материалы для устройств, работающих на объемных волнах (так называемых линий задержки), – монокристаллы NaCl и других галогенидов щелочных металлов, кварца, гранатов, кварцевое стекло, а также специальные стекла и металлические сплавы. Для устройств, работающих на поверхностных акустических волнах (полосовые фильтры, устройства акустической памяти, корреляционной обработки и др.), используют пьезо-, сегнетоэлектрики и полупроводники. Свойства основных акустических материалов для этих устройств приведены в табл. 1 и 2.

Материал	Направление распр. или нормаль, угол поворота	Затухание на 1 ГГц, дБ/мкс	Температурный коэффициент $\alpha_T \times 10^6$ (°C) ⁻¹
LiNbO ₃	Y, Z	1,07	94
	128°	-	75
Кварц	Y, X	2,6	-24
	ST, X	3,1	0
LiTaO _a	Y, Z	1,14	35
	167°	-	64
Bi ₁₂ GeO ₂₀	(001), [110]	1,64	120
AlPO ₄	80,4°	-	0
GaAs	(001), [110]	-	52

Таблица 1. Кристаллические материалы для устройств на ПАВ

Материал	Срез кристалла, направление распространения звуковой волны	Направление смещения в звуковой волне	Затухание, дБ × 10 ¹⁷ /(мкс* Гц ²)
Хлорид натрия NaCl (монокристалл)	[110]	[001]	0,22
Фторид лития LiF	[110]	[ПО]	1,23
Плавленый кварц	-	-	0,71
Свинцоводержащее стекло	-	-	21,1

Таблица 2. Свойства монокристаллов для устройств, работающих на ОАВ

В таблицах [100], [010] и [001] – направления соответственно вдоль осей X , Y и Z ; [110] – под углом 45° к осям X и Y ; [111] – вдоль пространственной диагонали куба.

Температурная зависимость времени задержки

Температурные эффекты можно исследовать с помощью модели линейного источника и линейного приемника, расположенной на свободной поверхности [1]. Если источник и приемник размещены на расстоянии l , то, пренебрегая дифракцией, потерями распространения и дисперсией, время задержки $T = l/v_0$. Скорость v_0 зависит от плотности, а также от упругих и пьезоэлектрических свойств материала, и, если температурные зависимости этих величин известны, можно определить температурную зависимость v_0 . Длина материала l также изменяется с температурой. Поэтому можно ввести температурный коэффициент задержки:

$$\alpha_T = \frac{1}{T} \cdot \frac{dT}{d\Theta} = \frac{1}{l} \cdot \frac{dl}{d\Theta} - \frac{1}{v_0} \cdot \frac{dv_0}{d\Theta}, \quad (1)$$

где Θ – температура. Величина α_T не зависит от расстояния между преобразователями l .

Относительное изменение времени T невелико, а зависимость $T(\Theta)$ линейна, поэтому величина α_T практически постоянна. В работе [2] приводятся данные о величинах α_T и температурных коэффициентах скорости для ряда материалов. Кварц обладает особенностью: при комнатной температуре и определенных углах среза (ST , X -срез) $\alpha_T = 0$. Экспериментально было доказано, что задержка T зависит от температуры практически по квадратичному закону [3]:

$$T(\Theta) \approx T(\Theta_0) \cdot [1 + c \cdot (\Theta - \Theta_0)^2], \quad (2)$$

где $c = 32,3 \cdot 10^{-9} (^\circ\text{C})^{-2}$, $\Theta_0 = 21,1^\circ\text{C}$. Задержка минимальна при температуре Θ_0 , которую называют температурой нулевого коэффициента задержки. Изменение времени задержки проиллюстрировано на рис. 1, где $\Delta T = T(\Theta) - T(\Theta_0)$ и $T_0 = T(\Theta_0)$.

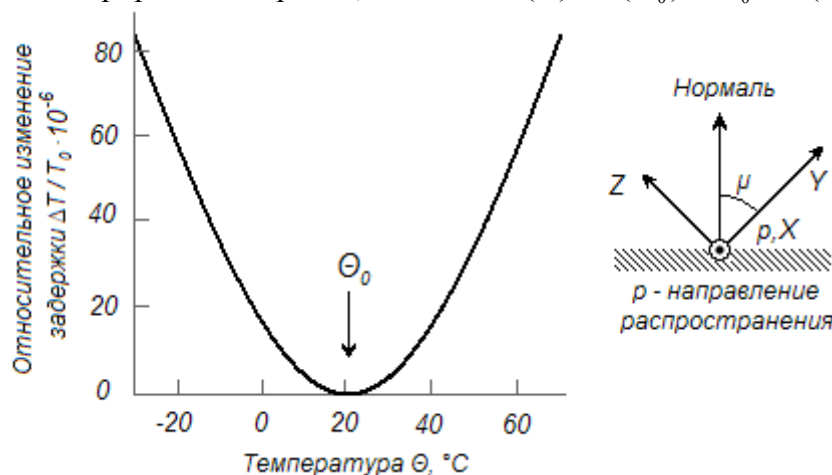


Рис. 1. Температурная зависимость времени задержки для кварца ST -среза, $\mu=42,75^\circ$

Из (2) следует, что температурный коэффициент задержки равен

$$\alpha_T = 2c(\Theta - \Theta_0) \quad (3)$$

и обращается в нуль при температуре Θ_0 .

Исследования образцов показали, что при изменении угла среза общий характер температурной зависимости остается прежним, но температура нулевого коэффициента задержки изменяется в пределах от -6°C до 114°C [3]. Это свойство можно использовать для оптимизации стабильности в заданном интервале температур.

Влияние температуры на частотные характеристики устройств

Зависимость резонансной частоты пьезоэлектрического резонатора от температуры может быть представлена первыми членами степенного ряда [4]:

$$\frac{f_h - f_{h0}}{f_{h0}} = \frac{\Delta f_h}{f_{h0}} = \sum_{n=1}^3 T f_h^{(n)} \cdot (\Theta - \Theta_0)^n, \quad (4)$$

где $T f_h^{(n)}$ – температурный коэффициент n -го порядка резонансной частоты, равный

$$T f_h^{(n)} = \frac{1}{n! f_{h0}} \left(\frac{\partial^n f_h}{\partial \Theta^n} \right)_{\Theta=\Theta_0}. \quad (5)$$

Здесь f_h – резонансная частота резонатора при температуре Θ , f_{h0} – резонансная частота при температуре Θ_0 , в окрестности которой аппроксимируется температурная зависимость резонансной частоты.

Срезы применяемых на практике резонаторов обычно ориентированы таким образом, чтобы существовала температура Θ_m , при которой $\left(\frac{\partial f_h}{\partial \Theta} \right)_{\Theta=\Theta_m} = 0$. Это означает, что в окрестности температуры Θ_m резонансная частота не меняется. Точку называют точкой поворота, потому как при данной температуре меняется знак наклона касательной к кривой, выражающей зависимость резонансной частоты. Если в определенном диапазоне частот член с коэффициентом $T f_h^{(2)}$ или $T f_h^{(3)}$ оказывает существенное влияние на температурную зависимость резонансной частоты, то последнюю можно представить кривой типа параболы второго или третьего порядка (рис. 2).

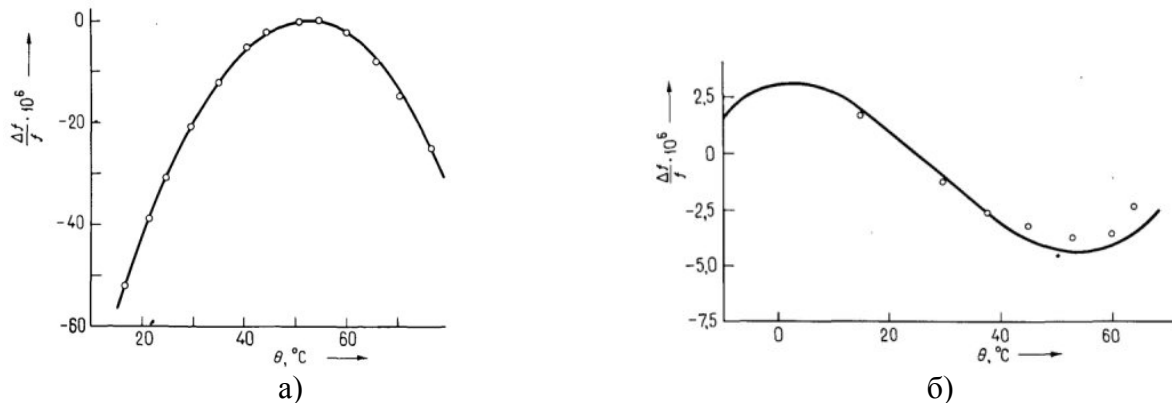


Рис. 2. Температурная зависимость резонансной частоты, имеющая вид параболы: а) второго порядка, для кварцевого продольно-колеблющегося стержня ($X\text{Ya}/-5^\circ$), б) третьего порядка, для кварцевого резонатора АТ-среза, совершающего колебания на третьей гармонике ($h = 3$)

Рассмотрим продольные колебания узкого тонкого стержня, используя выражения для определения температурной зависимости резонансной частоты из работы [4]. Резонансная частота такого стержня определяется соотношением:

$$f_h = \frac{h}{4l} \sqrt{\frac{1}{e s_{11}^D}}. \quad (6)$$

Выражения для температурного коэффициента первого, второго и третьего порядка резонансной частоты имеют следующий вид:

$$T f^{(1)} = -\frac{1}{2} \left(T s_{11}^{D(1)} + T e^{(1)} + 2 T l^{(1)} \right), \quad (7)$$

$$Tf^{(2)} = \frac{1}{2} \left\{ \left(Tf^{(1)} \right)^2 - Ts_{11}^{D(2)} - Te^{(2)} - 2Tl^{(2)} + \frac{1}{2} \left[\left(Ts_{11}^{D(1)} \right)^2 + \left(Te^{(1)} \right)^2 + \left(2Tl^{(1)} \right)^2 \right] \right\}, \quad (8)$$

$$Tf^{(3)} = Tf^{(2)}Tf^{(1)} - \frac{1}{3} \left(Tf^{(1)} \right)^3 - \frac{1}{2} \left[Ts_{11}^{D(2)} + Te^{(2)} + 2Tl^{(2)} - Ts_{11}^{D(2)}Ts_{11}^{D(1)} - \right. \\ \left. - Te^{(2)}Te^{(1)} - 2Tl^{(2)}Tl^{(1)} + \frac{1}{3} \left(Ts_{11}^{D(1)} \right)^3 + \frac{1}{3} \left(Te^{(1)} \right)^3 + \left(2Tl^{(1)} \right)^3 \right]. \quad (9)$$

В формулах (7)–(9) символом $Ts_{\lambda\mu}^{D(n)}$ обозначен температурный коэффициент n -го порядка коэффициента податливости $s_{\lambda\mu}^{D(n)}$, символом $Te^{(n)}$ – температурный коэффициент n -го порядка объемного расширения, $Tl^{(n)}$ – температурный коэффициент n -го порядка линейного расширения по длине стержня.

В настоящее время температурные коэффициенты n -го порядка коэффициентов жесткости и податливости определяют двумя способами.

Первый способ был предложен Бехманом и наиболее широко используется. Температурные коэффициенты определяются экспериментальным методом: путем измерения температурной зависимости резонансной частоты при использовании значений температурных коэффициентов линейного и объемного расширения и т.п. Полученные этим способом эффективные температурные коэффициенты коэффициентов жесткости и податливости соответствуют температурной зависимости промежуточных положений материальных точек пластины, и их использование возможно лишь при тех же условиях, при которых они были определены.

Второй способ основан на линейном описании колебаний пьезоэлектрического резонатора. Подробно он рассмотрен в работах [5] и [6]. Анализ базируется на предположении, что колебания совершаются в результате воздействия малого линейного динамического поля при наличии предварительного напряжения, вызванного температурой. Частотное уравнение рассматриваемого типа колебаний может быть выражено в виде частот отдельных материальных точек пластины, не зависящих от температуры. На эту возможность было указано в работе [7]. Частотное уравнение имеет один и тот же вид при использовании обоих указанных способов описания температурных коэффициентов. Используя относительные координаты, получим, что значения плотности и резонансной частоты, которые обуславливают размеры пластины, не зависят от температуры, причем температурные коэффициенты коэффициентов жесткости и податливости, определенные вторым способом, отличаются от эффективных температурных коэффициентов констант упругости, полученных первым способом.

Второй способ определения температурных коэффициентов, в отличие от первого, не нашел широкого распространения.

Аналогично температурной зависимости резонансной частоты, можно выразить температурную зависимость индуктивности резонатора. В этом случае требования к температуре существенно ниже, поэтому нет необходимости искать ориентацию среза резонатора, при которой температурный коэффициент индуктивности был бы равен нулю. Для применяемых типов пьезоэлектрических кварцевых резонаторов температурный коэффициент индуктивности находится в пределах $10^{-5} - 10^{-4} \text{ K}^{-1}$.

Со временем под влиянием различных физических изменений внутри и на поверхности резонатора происходит постоянное необратимое изменение резонансной частоты резонатора. Относительное изменение резонансной частоты за точно установленный интервал времени называют стабильностью (точнее, нестабильностью) резонансной частоты резонатора. В зависимости от величины интервала времени говорят о кратко- и долговременной стабильности. В первом случае имеются в виду интервалы времени порядка секунд или минут; во втором случае речь идет о сутках, неделях, месяцах или годах.

Долговременная стабильность резонансной частоты резонатора, которую часто называют «старением», зависит от свойств используемого пьезоэлектрического материала, а также от ориентации, размеров, типа колебаний, обработки и конечного исполнения резонатора.

У пьезоэлектрических кристаллических резонаторов, помещенных в вакуум или герметичный корпус, заполненный газом, не вызывающим коррозию электродов или поверхности пластины, в большинстве случаев со временем происходит повышение резонансной частоты. При длительном наблюдении за резонансной частотой можно выделить два периода. В течение первоначального периода, продолжающегося от нескольких недель до нескольких месяцев, временную зависимость резонансной частоты можно описать некоторой монотонно возрастающей функцией с монотонно убывающей до нуля производной. В течение второго периода более или менее регулярное значение резонансной частоты изменяется около постоянного или незначительно меняющегося среднего значения.

Старение указанной группы пьезоэлектрических резонаторов в начальный период может быть вызвано:

- отделением мельчайших частиц пластины или электродов при колебаниях, изменением упругих свойств электродов с течением времени,
- диффузионными процессами, протекающими в электродах и поверхностных слоях пластины, нарушенных при обработке и т.п.

Характер старения в существенной степени обусловлен причинами, его вызывающими, и тесно связан с типом колебаний, конструкцией резонатора и применяемой технологией обработки.

При понижении температуры, т.е. отключении термостата в течение нескольких часов или дней и последующем его включении, было зафиксировано при номинальной температуре некоторое уменьшение частоты, и затем опять частично повторялся начальный процесс старения. Такое же наблюдение сделал автор работы [8], когда измерял частотные изменения кварцевых резонаторов *AT*-среза с резонансной частотой 5 МГц, связанные с быстрым повышением температуры термостата от 50 до 51°C. Изменение температуры привело к уменьшению частоты на $\sim 10^{-7}$. В течение последующих нескольких минут частота увеличилась до значения, соответствующего новой температуре. Детальное изучение причин этого явления показало, что понижение частоты вызвано температурным ударом, т.е. возникновением упругого напряжения в электроде. Это упругое напряжение благодаря нелинейности коэффициентов жесткости вызвало кратковременное изменение резонансной частоты резонатора. Поэтому были рассмотрены такие ориентации срезов резонатора, при которых нелинейные коэффициенты жесткости минимальны и упругое напряжение в электроде или в поверхностном слое резонатора уже не приводит к зафиксированному ранее изменению резонансной частоты. Такой особенностью обладают, например, кварцевые резонаторы типа *SC*.

У стандартных пьезоэлектрических резонаторов величина старения обычно выше и составляет $\sim (1-2) \times 10^{-6}$ за год.

Более подробно процесс старения таких резонаторов рассмотрен в работе [4]. На рис. 3 показаны результаты измерения старения пьезокерамических резонаторов в форме дисков диаметром 12,5 мм и толщиной 1 мм, выполненных из материала $Pb(Zr_{0,59}Ti_{0,39}Mn_{0,02})O_3$ с поляризацией в направлении толщины. На одной и той же пластине измерялось в зависимости от времени, во-первых, изменение резонансной частоты (~ 200 кГц) для колебаний радиального типа, во-вторых, резонансной частоты ($\sim 2,5$ МГц) для продольных колебаний по толщине.

После 25 дней старения резонаторы были подвергнуты температурному удару (90°C в течение 8 ч), что вызвало уменьшение частоты. В дальнейшем, через несколько дней, резонансная частота вернулась на основную кривую старения. Под влиянием

температурного удара произошло изменение динамической индуктивности в эквивалентной электрической схеме и добротности Q резонатора (рис. 4).

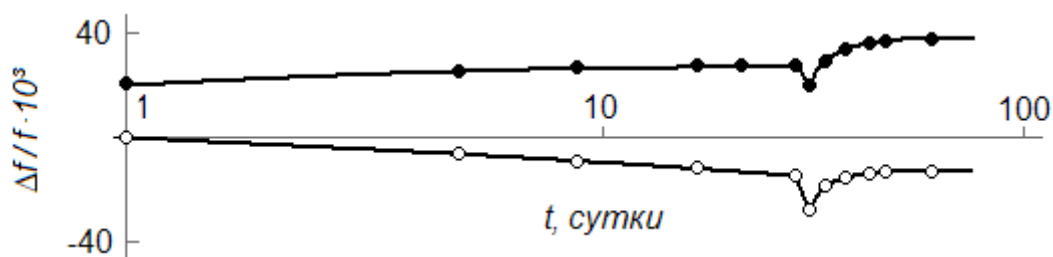


Рис. 3. Временная зависимость резонансной частоты продольных колебаний по толщине (темные точки) и радиальных колебаний (светлые точки) пьезоэлектрического резонатора, имеющего форму диска; в третьей декаде резонатор подвергнулся действию температурного удара

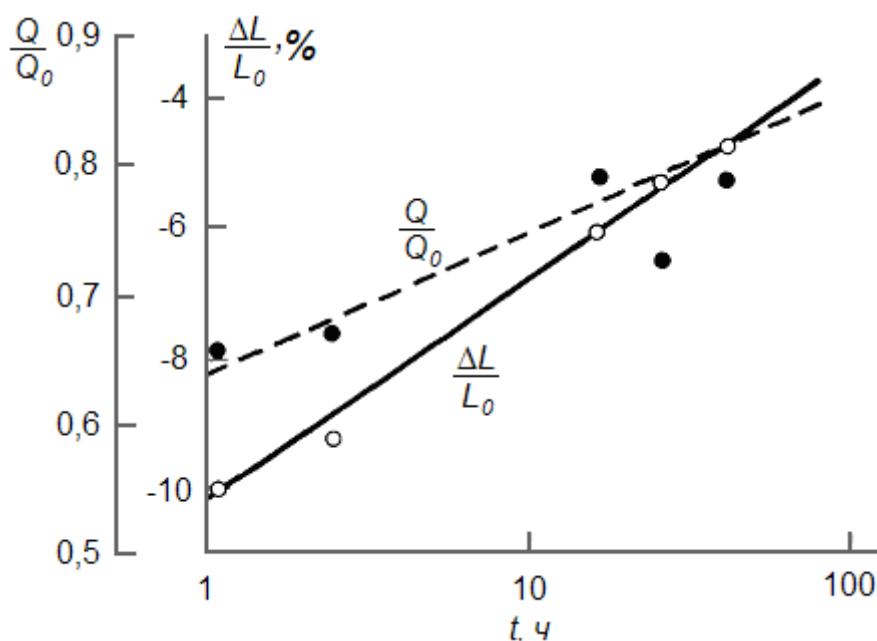


Рис. 4. Временные зависимости относительного изменения динамической индуктивности L_1 в эквивалентной электрической схеме и добротности Q_1 пьезоэлектрического резонатора в форме круглой пластины с радиальными колебаниями после температурного удара

Температурный удар у пьезокерамических материалов с составом, близким к морфотропному фазовому переходу, вызывает, по-видимому, кратковременное изменение соотношения частей, имеющих ромбоэдрическую и тетрагональную фазы, что приводит к отклонениям в формулах, описывающих упругие, пьезо- и диэлектрические свойства.

Заключение

В работе было рассмотрено влияние температуры окружающей среды на микроакустические устройства на поверхностных и объемных акустических волнах. В частности, было рассмотрено влияние температуры на коэффициент задержки и частотные характеристики. Приведены решения, которые позволяют компенсировать или учитывать это влияние. Заключительная часть статьи посвящена таким явлениям, как температурный удар и старение материала микроакустических устройств, рассмотрены причины этих явлений.

Литература

1. Морган Д. Устройства обработки сигналов на поверхностных акустических волнах: Пер. с англ. – М.: Радио и связь, 1990. – 416 с.
2. A.J. Slobodnik. Materials and their influence on performance. // in A.A. Oliner “Acoustic surface waves”, Springer, 1978. – P. 225–303.
3. J.F. Dias, H.E. Karrer, J.A. Kusters, J.H. Matsinger, M.B. Schulz. The temperature coefficient of delay-time for X-propagating acoustic surface waves on rotated Y-cuts of alpha quartz. // IEEE Trans. – SU-22. – 1975. – P. 46–50.
4. Зеленка И. Пьезоэлектрические резонаторы на объемных и поверхностных акустических волнах. – М.: Мир, 1990. – 584 с.
5. R. Holland. Temperature coefficients of stiffness in quartz. // IEEE Trans. Son. Ultrason. – SU-23. – 1976.
6. B.K. Sinha, H.F. Tiersten. Temperature derivatives of the fundamental elastic constants of quartz. // Proc. 32nd Ann. Symp. Freq. Control, Atlantic City, 1978.
7. R.N. Thurston. Waves in solids. / Handbuch der Physik, Vol. VI, a/4. – Springer, Berlin, Heidelberg, New York, 1974.
8. A.W. Warner. Use of parallel-field excitation in the design of quartz crystal units. // Proc. 17th Ann. Freq. Control Symp., 1963. – P. 248–266.

МЕТОДИКА ИНФОРМАЦИОННОЙ ОЦЕНКИ ВОСПРИЯТИЯ ИЗОБРАЖЕНИЙ

П.П. Коваленко

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В данной работе предлагается методика информационной оценки различных типов представления изображений. В работе используются достижения теории информации. В частности, для количественной информационной оценки изображений предлагается использовать формулы Хартли и Шеннона для количества информации в сообщении и информационной энтропии сообщения. С помощью данной методики можно оценивать восприятие человеком с точки зрения количества битовой информации.

В основе теории информации лежит открытие, что информация допускает количественную оценку. Эта идея была выдвинута еще в 1928 г. Хартли, но законченный и общий вид придал ей Шеннон.

Для системного анализа теория информации имеет большое значение. Ее методы позволяют провести ряд количественных исследований информационных потоков в изучаемой или проектируемой системе. Очень важными являются эвристические значения основных понятий теории информации – неопределенности, энтропии, количества информации, избыточности и пропускной способности.

Процесс получения информации о системе можно интерпретировать как изменение неопределенности состояния системы в результате приема сообщения. Исследуем сообщения, состоящие из последовательности элементов, взятых из некоторого набора. Количество информации, передаваемое этой последовательностью, совпадает с ее оригинальностью, или неожиданностью; следует подчеркнуть, что она не является прямой функцией длины последовательности [1]. Итак, информация, или оригинальность, является функцией «невероятности» получаемого сообщения. Рассматривая «невероятность», или «обратную вероятность», как воспринимаемое адресатом физическое возбуждение, можно связать количество информации H с обратной вероятностью («невероятностью») I . Восприятие, т.е. в данном случае количество информации, или оригинальность, пропорционально логарифму возбуждения, т. е. обратной вероятности I :

$$H = K \log_a I .$$

Невероятность I сообщения есть величина, обратная вероятности его появления ω , следовательно,

$$H = -K \log_a \omega .$$

Если мы используем логарифмы при основании 2, мы получим ровно одну единицу информации («бит» в терминологии общей теории связи). Таким образом, для количества информации в сообщении получаем выражение

$$H = -\log_2 \omega .$$

Введем следующее определение: количество информации, передаваемое сообщением, есть двоичный логарифм числа последовательных дилемм, однозначно определяющих сообщение. Такое определение количества информации имеет то преимущество, что оно является весьма общим и поэтому перекрывает все другие, имеющие более конкретную форму. Это в особенности относится к определениям, в основе которых лежит понятие комбинации элементов набора; они выражаются формулой, указанной Винером и приведенной Шенноном к более удобному для использования виду:

$$H = -t \sum_{i=1}^n p_i \log_2 p_i .$$

Здесь p_i – вероятности появления символов, выбираемых из набора n символов и упорядоченных в некоторую последовательность [1].

Посчитаем в соответствии с приведенными выше формулами и зависимостями количество информации для плоского изображения. В качестве исходного возьмем изображение картины Н. Рериха «Матерь Мира». Посчитаем количество информации для трех произвольно взятых областей изображения. Обозначим выбранные области как a, b, c (рис. 1).

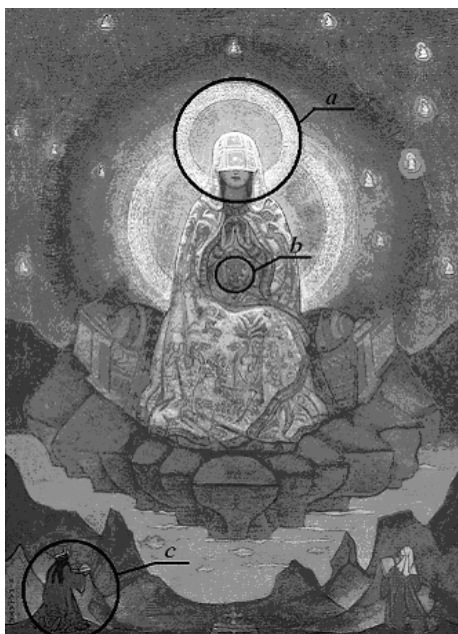


Рис. 1. Области для расчета количества информации

В данном расчете нами будет рассмотрено несколько случаев для различной освещенности. Для этого использованы лампы разной мощности (25, 45, 60 и 70 Вт.) (рис. 2). Положение источника света и наблюдателя относительно плоскости изображения во всех случаях одинаковое.

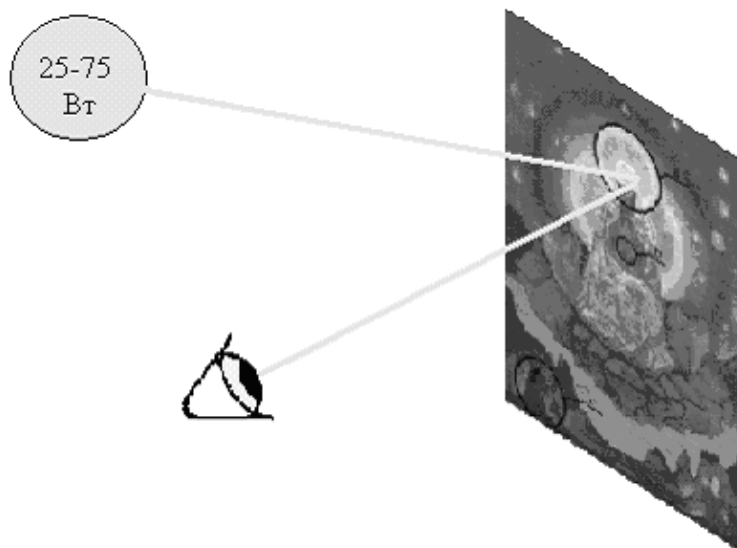


Рис. 2. Схема эксперимента

Для каждого варианта освещения нами будет посчитана суммарное количество информации для областей a, b и c. Расчет будем проводить по формуле Хартли для количества информации в сообщении:

$$J = -\sum_{l=1}^n \log_2 p_l,$$

здесь p_i – вероятности появления этих элементов; J – количество информации.

Чтобы воспользоваться данной формулой, посчитаем количество элементов изображения в выделенных областях а, б и с, которые мы можем различить невооруженным глазом при данном освещении. Приняв максимальное количество элементов n_{\max} , которое удалось разглядеть в данной области, за 100%, вычислим вероятности появления элементов n данной области для данного освещения:

$$p_i = \frac{n}{n_{\max}}.$$

Результаты вычислений сведены в табл. 1 и 2.

	N		
W	a	b	c
25	11	7	12
40	14	10	15
60	23	14	18
75	24	16	20

Таблица 1. Количество элементов изображения

	p_i		
	a	b	c
25	0.458	0.438	0.6
40	0.583	0.625	0.75
60	0.958	0.875	0.9
75	1	1	1

Таблица 2. Вероятности появления элементов изображения

По результатам расчета построим график зависимости количества информации в изображении от мощности источника света (рис. 3).

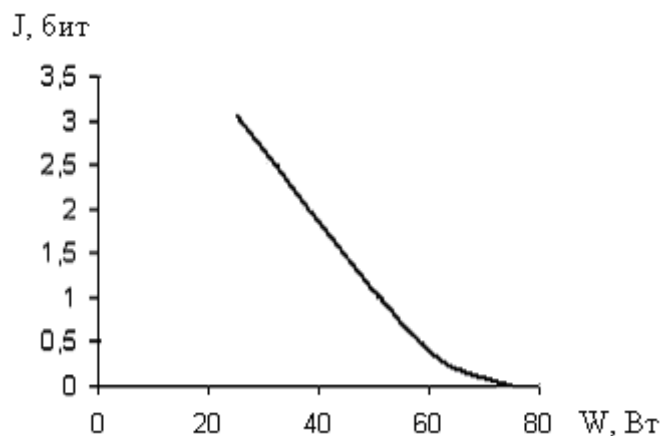


Рис. 3. Зависимость количества воспринимаемой при просмотре изображения информации от мощности источника света

Нанесем теперь исходное изображение на поверхность цилиндра и заставим его вращаться по этой поверхности с различной скоростью ω (рис. 4). Подробнее о том, как это было реализовано, можно посмотреть в [2].



Рис. 4. Вращение изображения по поверхности цилиндра

Подсчет количества элементов и вероятностей их появления произведем так же, как и в предыдущем расчете количества информации в плоском изображении. Результаты сведем в табл. 3. Для подсчета количества информации также воспользуемся формулой Хартли:

$$J = -\sum_{i=1}^n \log_2 p_i .$$

W, Вт	p_i			
	ω_1	ω_2	ω_3	ω_4
25	0,521	0,391	0,26	0,217
75	0,565	0,608	0,434	0,391

Таблица 3. Вероятности появления элементов изображения

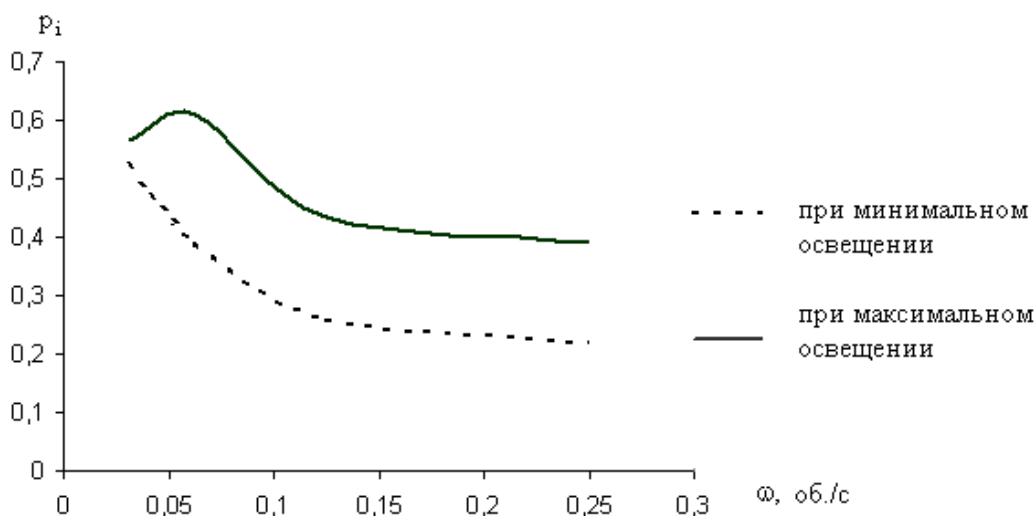


Рис. 5. Зависимость вероятности появления элементов от скорости вращения изображения по поверхности цилиндра

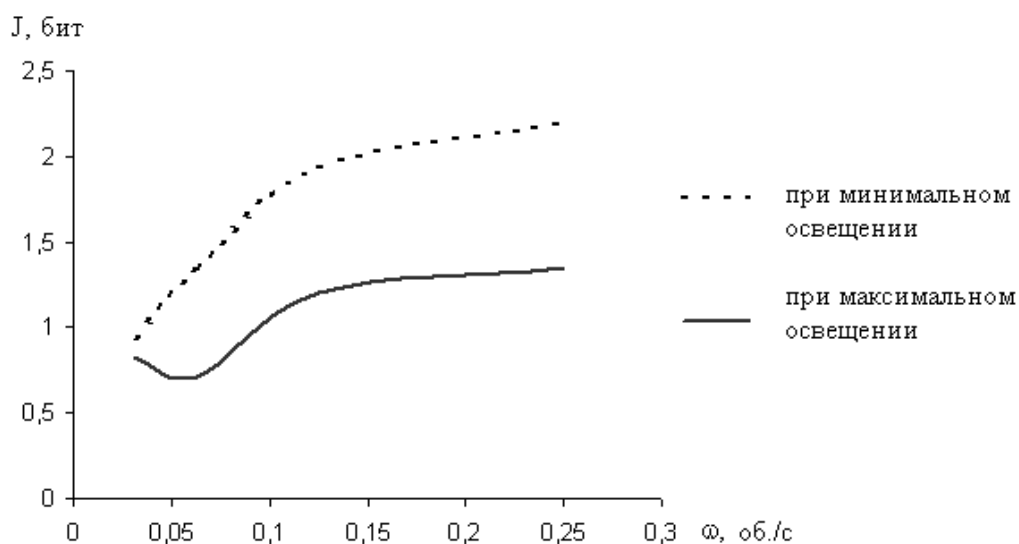


Рис. 6. Зависимость количества информации от скорости вращения изображения по поверхности цилиндра

Таким образом, в работе была предложена методика информационной оценки различных типов представления изображения. С помощью данной методики можно оценивать восприятие человека с точки зрения количества битовой информации.

Литература

1. Моль А. Теория информации и эстетическое восприятие. – М.: Мир, 1966. – 351 с.
2. Коваленко П.П. Визуализация изображений на цилиндре и торе // Научно-технический вестник СПбГУ ИТМО. – 2007. – Выпуск 37. Современная физика. Труды молодых ученых.– С. 318–321.

СРАВНЕНИЕ ДЕФОРМАЦИЙ АЛЮМИНИЕВЫХ И КРЕМНИЕВЫХ БАЛОК И ПЛАСТИН В ТЕПЛОВЫХ ПОЛЯХ

И.А. Абдурахманов

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В статье рассмотрен вопрос действия тепловых полей на балки и пластины из алюминия и кремния. Смоделированы нагрузки, возникающие при воздействии теплового поля, и получены деформации образцов, что и дает возможность оценивать воздействие теплового поля.

Термоупругость занимается вопросами равновесия тела как термодинамической системы, взаимодействие которой с окружающей средой заключается лишь в механической работе внешних сил и теплообмене. Тело, как и в классической теории упругости, рассматривается в виде материального континуума, обладающего свойствами идеальной упругости, однородности и изотропии. Состояние термодинамической системы определяется конечным числом независимых переменных – макроскопических величин, называемых термодинамическими параметрами.

Одним из независимых макроскопических параметров термодинамической системы, отличающим ее от механической, является температура как мера интенсивности теплового движения.

Для изучения реального процесса термоупругого деформирования тела, подвергающегося действию внешних сил, используется термодинамика необратимых процессов, основная идея которой заключается в понятиях локального равновесия и медленных процессов. Значение термодинамических параметров определяется для внезапно выделенной макроскопически малой части тела, которую можно рассматривать в состоянии локального равновесия. Такое обобщение классической термодинамики основано на предположении, что равновесное состояние устанавливается прежде всего в макроскопически малых частях системы, когда она в целом еще находится в равновесии.

Термодинамическая теория необратимых процессов предполагает, что основные уравнения термодинамики обратимых процессов сохраняются справедливыми и для локально равновесных макроскопически малых частях системы. На основании этого утверждения будут строиться дальнейшие выводы и проводиться опыты.

В статье будут рассмотрены балка и пластина из двух материалов: алюминия и кремния, физические свойства которых приведены в табл. 1.

	Алюминий	Кремний
Модуль Юнга $\frac{H}{мм^2}$	69000	10890
Плотность $\frac{ρ}{мм^3}$	0,0027	0,00233
Коэффициент теплового расширения $\frac{1}{°C}$	$23 \cdot 10^{-6}$	$2,33 \cdot 10^{-6}$
Теплопроводность $\frac{Вт}{мК}$	200	126

Таблица 1. Физические свойства материалов

Опыт 1. Алюминиевая балка (рис. 1) жестко закреплена на одном из своих концов, на балку действует тепловое поле с распределенной температурой $\Delta T = 5^{\circ}C$. Найдите силу, которая действует на балку в данном тепловом поле.

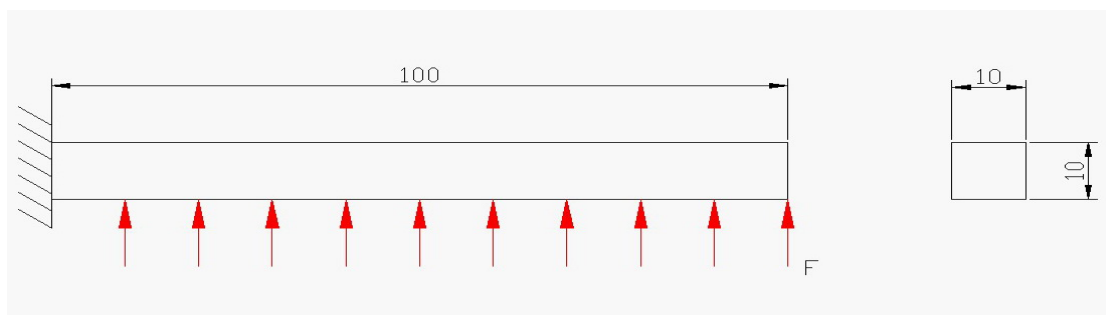


Рис. 1. Балка

Для этого воспользуемся формулой (1):

$$F = whE\alpha\Delta T, \tag{1}$$

где w – ширина балки, h – высота балки (размеры приведены на рис.1 соответственно), E – модуль Юнга, α – коэффициент теплового расширения (приведены в табл. 1), $\Delta T = 5^\circ\text{C}$. Из формулы (1) получаем, что сила F , действующая на балку равна: $F = 793,5H$.

Для исследования деформаций, вызванных тепловым полем, т.е. распределенной нагрузкой $F = 793,5H$, воспользуемся прикладным пакетом среды трехмерного моделирования Solid Works – Cosmos. Модель балки приведена на рис. 2.



Рис. 2. Модель балки

К балке приложена распределенная нагрузка, как показано на рис. 1, затем проведен анализ в прикладном пакете Cosmos. После проведения анализа были получены следующие результаты (табл. 2, рис. 3).

Имя	Тип	Мин (мм)	Место (мм)	Макс (мм)	Место (мм)
Результаты	Деформации (Перемещения)	0	-14,0239, 1,46187, 100	1,727	-14,0239, 1,46187, 0

Таблица 2. Результаты анализа

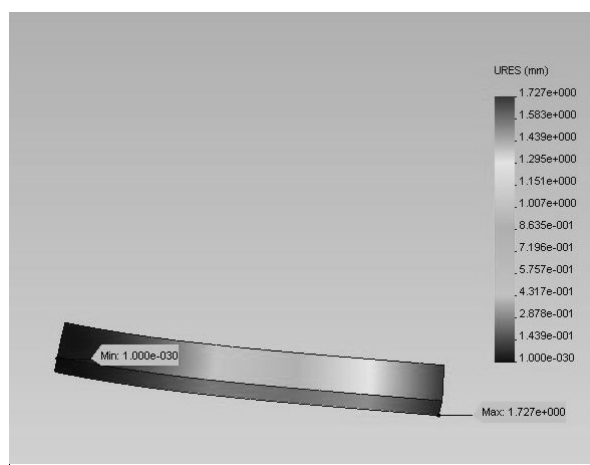


Рис. 3. Деформации и напряжения

Опыт 2. Алюминиевая пластина (рис. 4) жестко закреплена в центре, на пластину действует тепловое поле с распределенной температурой $\Delta T = 5^\circ\text{C}$. Найдем силу, которая действует на пластину в данном тепловом поле. Для этого воспользуемся формулой (2)

$$F = lwhE\alpha\Delta T, \quad (2)$$

где h – высота пластины, l – длина пластины, w – ширина пластины (размеры приведены на рис. 4 соответственно), E – модуль Юнга, α – коэффициент теплового расширения (приведены в табл. 1), $\Delta T = 5^\circ\text{C}$.

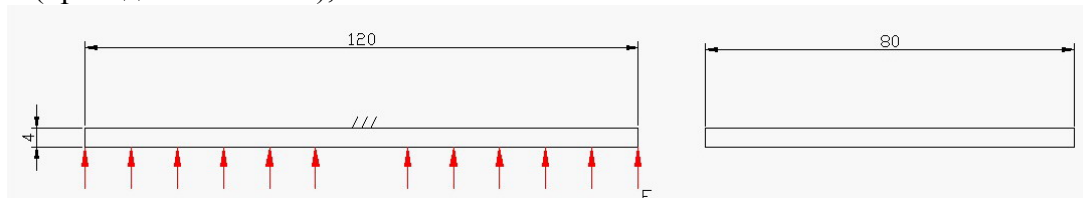


Рис. 4. Пластина

Из формулы (2) получаем, что сила F , действующая на балку, равна $F = 4416\text{H}$. Для исследования деформаций, вызванных тепловым полем, т.е. распределенной нагрузкой $F = 4416\text{H}$, воспользуемся прикладным пакетом среды трехмерного моделирования Solid Works – Cosmos. Модель пластины приведена на рис. 5.

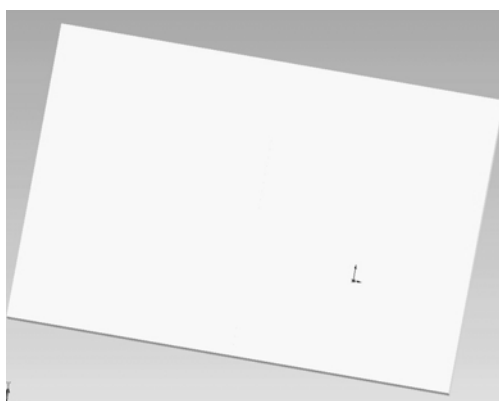


Рис. 5. Модель пластины

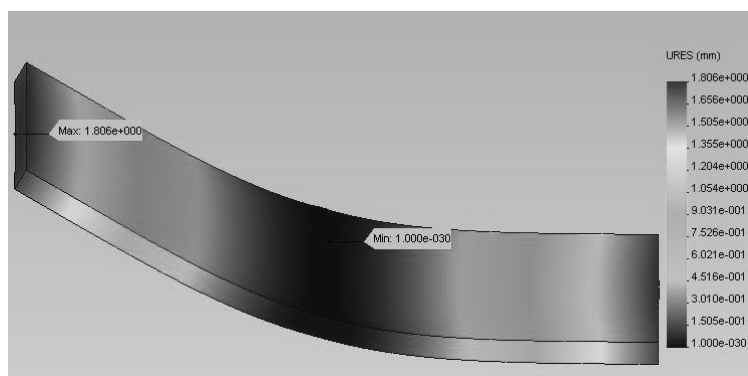


Рис. 6. Деформации и напряжения

К пластине приложена распределенная нагрузка, как показано на рис. 4, затем проведен анализ в прикладном пакете Cosmos. После проведения анализа были получены следующие результаты (табл. 3, рис. 6).

Имя	Тип	Мин (мм)	Место (мм)	Макс (мм)	Место (мм)
Результаты	Деформации (Перемещения)	0	-27,9981, 54,073, 4	1,80624	-88,9981, 14,0731, 0

Таблица 3. Результаты анализа

Опыт 3. Кремниевая балка (рис. 7) жестко закреплена на одном из своих концов, на балку действует тепловое поле с распределенной температурой $\Delta T = 5^\circ C$. Найдем силу, которая действует на балку в данном тепловом поле.

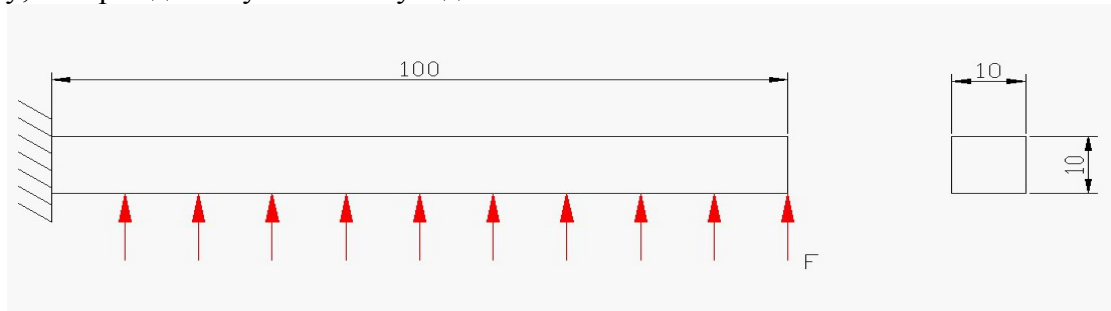


Рис. 7. Балка

Для этого воспользуемся формулой (1). Из формулы (1) получаем, что сила F , действующая на балку, равна $F = 12,6H$. Для исследования деформаций, вызванных тепловым полем, т.е. распределенной нагрузкой $F = 12,6H$, воспользуемся прикладным пакетом среды трехмерного моделирования Solid Works – Cosmos. Модель балки приведена на рис. 8.

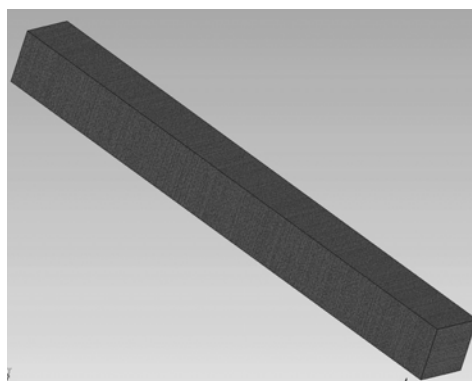


Рис. 8. Модель балки

К балке была приложена распределенная нагрузка, как показано на рис. 1, затем проведен анализ в прикладном пакете Cosmos. После проведения анализа были получены следующие результаты (табл. 4, рис. 9).

Имя	Тип	Мин (мм)	Место (мм)	Макс (мм)	Место (мм)
Результаты	Деформации (Перемещения)	0	-14,0239, 1,46187, 100	0,174757	-14,0239, 1,46187, 0

Таблица 4. Результаты анализа

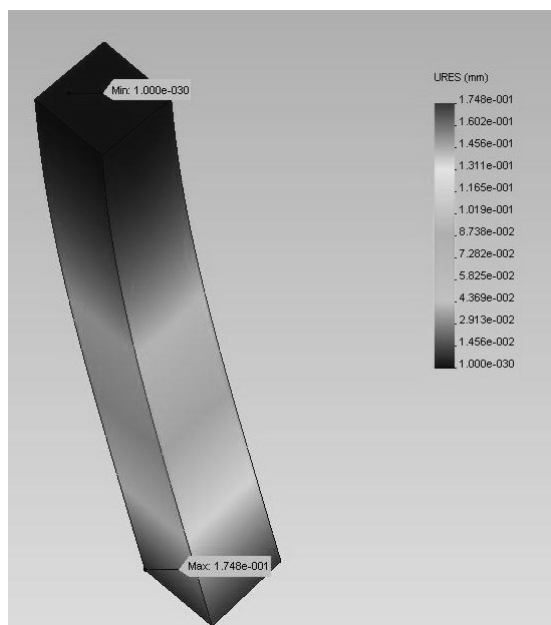


Рис. 9. Деформации и напряжения

Опыт 4. Кремниевая пластина (рис. 10) жестко закреплена в центре, на пластину действует тепловое поле с распределенной температурой $\Delta T = 5^\circ\text{C}$. Найдем силу, которая действует на пластину в данном тепловом поле. Для этого воспользуемся формулой (2). Из формулы (2) получаем, что сила F , действующая на балку, равна $F = 4872\text{H}$.

Для исследования деформаций, вызванных тепловым полем, т.е. распределенной нагрузкой $F = 4872\text{H}$, воспользуемся прикладным пакетом среды трехмерного моделирования Solid Works – Cosmos. Модель пластины приведена на рис. 11.

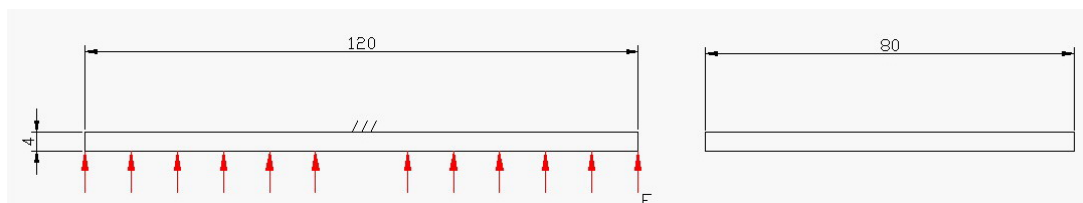


Рис. 10. Пластина

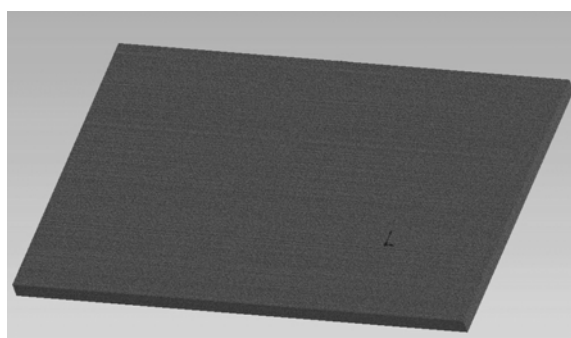


Рис. 11. Модель пластины

К пластине приложена распределенная нагрузка, как показано на рис. 7, затем проведен анализ в прикладном пакете Cosmos. После проведения анализа были получены следующие результаты (табл. 5, рис. 12).

Имя	Тип	Мин (мм)	Место (мм)	Макс (мм)	Место (мм)
Результаты	Деформации (Перемещения)	0	-27,9981, 54,0731, 4	13,1999	-88,9981, 14,0731, 0

Таблица 5. Результаты анализа

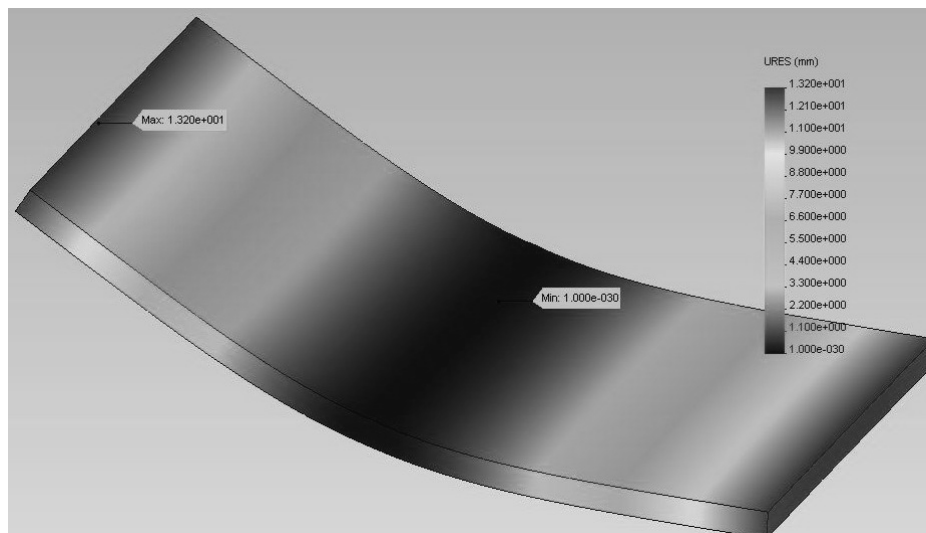


Рис. 12. Деформации и напряжения

В работе представлены четыре опыта, в каждом из которых была рассчитана нагрузка, которая действует на тот или иной образец под действием теплового поля. Нагрузка была смоделирована, получены результирующие деформации и напряжения образцов. Из полученных данных можно сделать вывод, что тепловое поле действует на алюминиевую балку и кремниевую балку практически одинаково: минимальные и максимальные деформации наблюдаются в одних и тех же местах образцов. Разница лишь в распределенной нагрузке, а значит и максимальной деформации, она больше у кремния, что и не удивительно, ибо модуль Юнга у кремния меньше, чем у алюминия. Такая же закономерность наблюдается и у пластин из алюминия и кремния.

Максимальная деформация алюминиевой балки составляет 1,727 мм, алюминиевой пластины 1,81 мм. Максимальная деформация кремниевой балки составляет 0,175 мм, кремниевой пластины 13,199 мм.

Также опыты показали, что прикладная программа Cosmos среды трехмерного моделирования Solid Works, дает достаточно наглядную картину распределения деформаций и напряжений в образцах (моделях) под действием теплового поля.

Литература

1. Коваленко Д.А. Введение в термоупругость. – Киев: Наукова думка, 1965. – 200 с.
2. Коваленко Д.А. Основы термоупругости. – Киев: Наукова думка, 1970. – 301 с.
3. Лысенко И.Е., Полищук Е.В., Хайрулина В.А. Сборник практических работ по курсу «Микроэлектромеханика». – Таганрог: Изд-во технологического института ЮФУ, 2007.
4. Свойства химических элементов. – Режим доступа: <http://www.bizinfo.otrok.ru/chem/elem.php?n=14>

ВОССТАНОВЛЕНИЕ ДЕФОКУСИРОВАННЫХ ИЗОБРАЖЕНИЙ МЕТОДОМ ДВУМЕРНОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ И РЕГУЛЯРИЗАЦИИ ТИХОНОВА

В.В. Шемплинер

Научный руководитель – д.т.н., профессор В.С. Сизиков

Рассматривается задача реконструкции дефокусированных изображений. Задача сводится к решению двумерного интегрального уравнения Фредгольма I рода типа свертки. Для решения прямой задачи (моделирования дефокусированного изображения) используется двумерное преобразование Фурье. Для решения обратной задачи (реконструкции дефокусированного изображения) используется метод преобразования Фурье и метод регуляризации Тихонова. Приведены численные результаты с иллюстрациями.

Введение

В данной работе рассматривается задача реконструкции (восстановления, реставрации) дефокусированных изображений. Под *изображением* подразумевается фотоснимок или оптико-электронное воспроизведение объекта природы, текста, человека, здания, самолета, автомобиля, космического объекта, наземного объекта из космоса и т.д.

Реконструкция дефокусированных изображений является одной из актуальных задач цифровой обработки изображений [1–5]. Данная задача обычно описывается двумерным интегральным уравнением Фредгольма I рода типа свертки [6–13]. Одним из распространенных способов решения этой некорректной задачи (как прямой, так и обратной) является использование преобразования Фурье (ПФ) с фильтрацией Винера, Тихонова и т.д. (для устойчивости решения). При этом края дефокусированного изображения обычно формируются с использованием так называемых *граничных условий* [5, с. 108], [14, 15]. Их введение обусловлено тем, что на практике для регистрации изображений применяются системы с матрицами, размер которых совпадает с размером области фокусировки. В данной работе отмечается искусственность граничных условий при формировании краев дефокусированного изображения, а также неадекватность описания физической задачи дефокусировки с помощью ПФ. Разработан ряд программ на MATLAB'е для решения прямой и обратной задач. Производится сравнение разработанных алгоритмов и программ с классическими методами (параметрической фильтрации Винера и др.) и соответствующими программами реконструкции (`deconvwnr.m`, `deconvlucy.m`, `deconvreg.m`), а также дана качественная и количественная оценка погрешностей полученных результатов.

Математическое описание задачи получения дефокусированного изображения

Изображение можно описать функцией $f(\xi, \eta)$ двух пространственных координат ξ, η , представляющей интенсивность или яркость изображения в каждой точке (ξ, η) плоскости. Математически задача восстановления изображений, искаженных линейной пространственно-инвариантной системой (однородной системой), описывается *интегральным уравнением Фредгольма I рода типа свертки* [4–13, 16, 17]:

$$Af \equiv \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} k(x - \xi, y - \eta) f(\xi, \eta) d\xi d\eta = g(x, y), \quad -\infty < x, y < +\infty, \quad (1)$$

где $g(x, y)$ – искаженное изображение на выходе системы; $f(\xi, \eta)$ – исходное изображение на входе системы; $k(x - \xi, y - \eta)$ – импульсная характеристика системы, или функция рассеяния точки (ФРТ), являющаяся трансляционно-инвариантной (разностной) [4, с. 380].

Рассмотрим физический принцип получения расфокусированного изображения. Считаем, что фотографируемый объект (для простоты полагаемый плоским) и фотопленка фотоаппарата расположены параллельно апертуре линзы фотоаппарата по разные стороны от нее на расстояниях соответственно, f_1 и $f_2 + \delta$ от линзы (δ – погрешность фокусировки изображения), причем

$$1/f_1 + 1/f_2 = 1/f, \quad (2)$$

где f – фокусное расстояние линзы, $f_1 \geq f_2$. В результате на фотопленке возникнет перевернутое изображение (см. рис. 1).

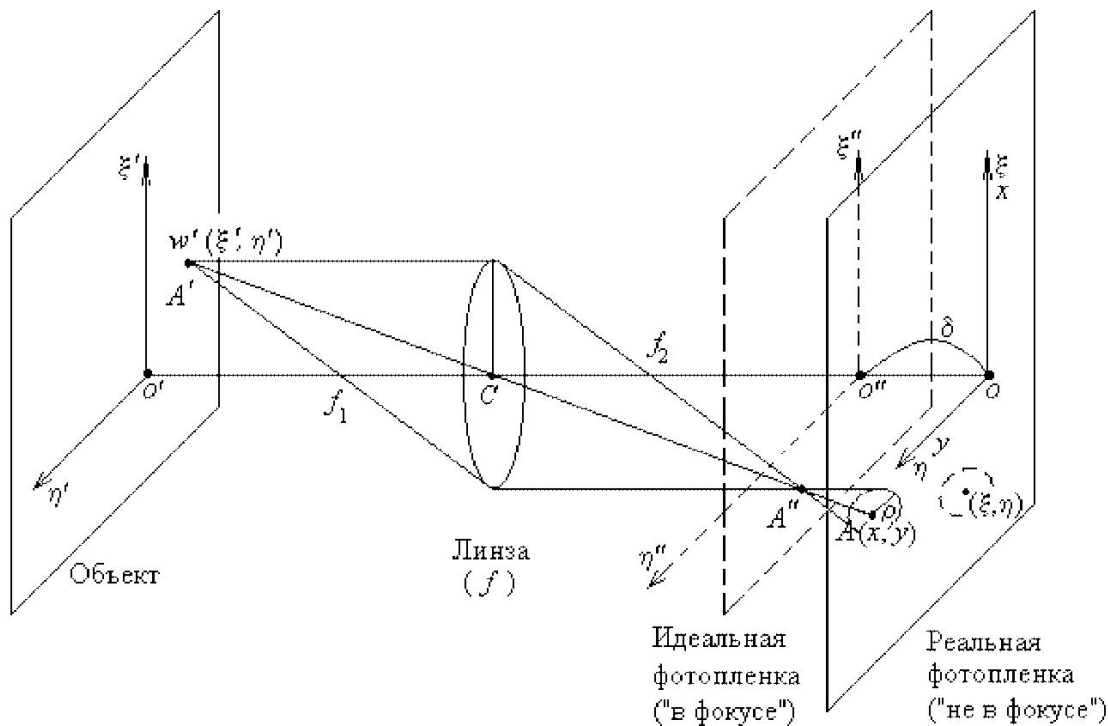


Рис. 1. Модель расфокусировки изображения

Введем в плоскости объекта прямоугольную систему координат $\xi'O'\eta'$, на «идеальной» фотопленке, расположенной «в фокусе» ($\delta = 0$) – систему координат $\xi''O''\eta''$, а на реальной фотопленке, расположенной «вне фокуса» ($\delta \neq 0$) – систему координат $\xi O\eta$, а также совпадающую с ней xOy . Обозначим через $w'(\xi', \eta')$ интенсивность, исходящую из некоторой точки $A'(\xi', \eta')$ объекта. Точка A' отобразится на «идеальной» фотопленке также в точку A'' с интенсивностью $w''(\xi'', \eta'') = w'(\xi', \eta')$ и с координатами $\xi'' = -\xi'/q$, $\eta'' = -\eta'/q$, где $q = f_1 / f_2$. На реальной же фотопленке A' отобразится в круг радиуса

$$\rho = \frac{a\delta}{f_2} \quad (3)$$

с центром в точке $A(x, y)$, где a – радиус апертуры линзы.

Опишем математически задачу дефокусировки. Рассмотрим, помимо круга с центром в точке $A(x, y)$, некоторый другой круг с центром в точке (ξ, η) (см. рис. 1). Радиусы этих (и других) кругов одинаковы и равны ρ , а их площади равны $S = \pi\rho^2$. В результате некоторая интенсивность $w(\xi, \eta)$, соответствующая точке (ξ, η) , будет «размазана» по кругу радиуса ρ и площади $S = \pi\rho^2$ с плотностью интенсивности $w(\xi, \eta)/\pi\rho^2$ (постоянной, в первом приближении, в пределах круга). Интенсивность в точке $A(x, y)$ будет результатом суммирования (интегрирования) по всем тем кругам, которые накрывают точку $A(x, y)$. Условие накрытия точки $A(x, y)$ кругом с центром в точке (ξ, η) и радиуса ρ есть

$$\sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho, \quad (4)$$

в результате интенсивность в точке $A(x, y)$ будет равна

$$g(x, y) = \iint_{\sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho} \frac{w(\xi, \eta)}{\pi\rho^2} d\xi d\eta. \quad (5)$$

Запишем (5) в виде

$$\iint_{\sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho} \frac{w(\xi, \eta)}{\pi\rho^2} d\xi d\eta = g(x, y). \quad (6)$$

Соотношение (6) есть *двумерное интегральное уравнение I рода относительно $w(\xi, \eta)$* [8–10,15]. Однако оно записано не в стандартной форме. Приведем его к стандартной форме. Запишем (6) в виде

$$\int_{-\infty - \infty}^{\infty} \int_{-\infty - \infty}^{\infty} k(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta = g(x, y), \quad -\infty < x, y < \infty, \quad (7)$$

где

$$k(x - \xi, y - \eta) = \begin{cases} 1/\pi\rho^2, & \sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho, \\ 0, & \text{иначе} \end{cases} \quad (8)$$

или

$$k(x, y) = \begin{cases} 1/\pi\rho^2, & \sqrt{x^2 + y^2} \leq \rho, \\ 0, & \text{иначе.} \end{cases} \quad (9)$$

Соотношение (7) есть *двумерное интегральное уравнение Фредгольма I рода типа свертки* [6–13]. В нем $g(x, y)$ – интенсивность в плоскости реальной (расположенной «не в фокусе») фотопленки, $k(x, y)$ – ядро интегрального уравнения, причем ρ определяется согласно (3), где a и f_2 известны, а δ (или ρ) может быть определено путем подбора. Ядро интегрального уравнения $k(x, y)$ называется *функцией рассеяния точки* (ФРТ, PSF). Наконец, $w(\xi, \eta)$ есть искомая интенсивность (которая была бы на снимке при $\delta = 0$).

На рис. 2 представлен результат применения к исходному изображению (томограмме-фантому) искажающего оператора с ядром $k(x, y)$ (9).

В данном случае предполагается, что информация на краях снимка недоступна, а изображение усечено по краям. Это связано с тем, что в большинстве случаев регистрирующая система (матрица фотоаппарата) не позволяет получить снимок завышенных размеров (т.е. с учетом размытых краев).

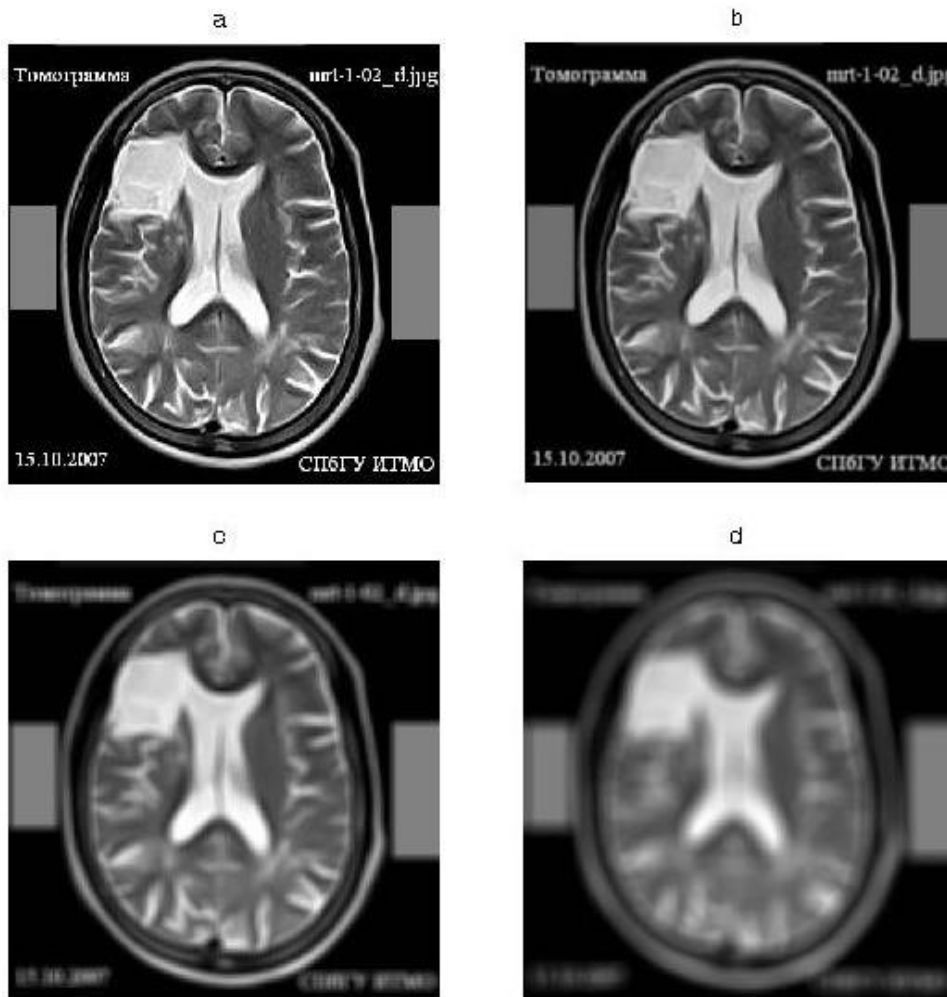


Рис. 2. Исходное изображение (а). Результат дефокусировки исходного изображения с использованием искажающей функции в виде диска радиуса 2 (b), 5 (c) и 10 (d) пикселей, соответственно

Для моделирования дефокусированного изображения больше подходит методика Гонсалеса и др. [4, 5], когда обрезаются размытые края и решается сложная краевая задача. Если же наблюдательное устройство позволяет зарегистрировать дефокусированный снимок с размытыми краями (большого размера, чем исходное изображение), то более эффективным представляется алгоритм обработки с учетом размытых краев. При этом не нужно решать сложную краевую задачу, так как информация на краю снимка не потеряна.

Реконструкция дефокусированного изображения методом преобразования Фурье и регуляризации Тихонова

Ниже рассмотрена фильтрация *методом регуляризации Тихонова*, которая в некоторых источниках [4, с. 395] называется *методом минимизации сглаживающего функционала со связью*, или *методом наименьших квадратов со связью* [5, с. 187]. В этом методе регуляризация задачи достигается заменой исходной задачи на задачу нахождения экстре-

мум (минимума) некоторого сглаживающего функционала. В качестве такого функционала $C[f]$ можно использовать квадрат нормы лапласиана [4, с. 397]

$$C[f] = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (\nabla^2 f(x, y))^2 \quad (10)$$

с дополнительным ограничением (связью) вида

$$\|g - H \tilde{f}\|^2 = \|\eta\|^2. \quad (11)$$

Норма (10) соответствует норме в пространстве Соболева W_2^2 [16, с. 501].

В отечественной литературе [11, с. 192] чаще используют норму в пространстве L_2 . В методе регуляризации Тихонова ставят два условия: условие минимизации невязки типа

$$\|g - H f\|^2 = \min_f \quad (12)$$

и условие минимизации нормы решения типа

$$\|f\|^2 = \min_f. \quad (13)$$

Рассмотрим одномерное интегральное уравнение Фредгольма I рода типа свертки:

$$Hw \equiv \int_{-\infty}^{\infty} k(x - \xi) w(\xi) d\xi = f(x), \quad -\infty < x < \infty. \quad (14)$$

Применительно к нему в методе регуляризации Тихонова решение находится из условия минимума сглаживающего функционала (10):

$$\int_{-\infty}^{\infty} [Hw - f(x)]^2 dx + \alpha \int_{-\infty}^{\infty} M(\omega) |Y(\omega)|^2 d\omega = \min_w, \quad (15)$$

где

$$M(\omega) = |\omega|^{2q} \quad (16)$$

– регуляризатор q -го порядка, причем $q \geq 0$ – порядок регуляризации, обычно $q = 1$.

Из условия (15) получается следующее выражение для регуляризованного решения

$$w_\alpha(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{\lambda(-\omega) F(\omega)}{L(\omega) + \alpha M(\omega)} e^{-i\omega\xi} d\omega, \quad (17)$$

где

$$L(\omega) = |\lambda(\omega)|^2 = \lambda(\omega)\lambda(-\omega) = \text{Re}^2 \lambda(\omega) + \text{Im}^2 \lambda(\omega). \quad (18)$$

Уравнение (1) как двумерное интегральное уравнение Фредгольма I рода типа свертки может быть решено методом двумерного ПФ (инверсная фильтрация). Решение записывается в виде ОПФ

$$w(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(\omega_1, \omega_2) e^{-i(\omega_1\xi + \omega_2\eta)} d\omega_1 d\omega_2, \quad (19)$$

где ПФ (спектр) решения

$$W(\omega_1, \omega_2) = \frac{G(\omega_1, \omega_2)}{K(\omega_1, \omega_2)}, \quad (20)$$

где $G(\omega_1, \omega_2)$ и $K(\omega_1, \omega_2)$ – преобразования Фурье (спектры) правой части $g(x, y)$ и ядра интегрального уравнения (2.21), равные

$$G(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy, \quad (21)$$

$$K(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} k(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy. \quad (22)$$

Ядро $k(x, y)$ выражается аналитической формулой (9), поэтому $K(\omega_1, \omega_2)$ может быть найдено аналитически согласно (16). А $G(\omega_1, \omega_2)$ (а также $K(\omega_1, \omega_2)$) должно быть найдено численно по стандартной программе двумерного ДПФ (обычно в виде БПФ).

Однако задача решения уравнения (1) является некорректной [1–3, 8, 13–16]. Это связано с тем, что функция $g(x, y)$ измеряется с погрешностью, и это ведет к сколь угодно большим погрешностям решения $w(\xi, \eta)$. Поэтому формулы (14)–(16) не годятся для устойчивого решения уравнения (1). Устойчивое решение уравнения (1) методами двумерного ПФ и регуляризации Тихонова имеет вид

$$w_\alpha(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_\alpha(\omega_1, \omega_2) e^{-i(\omega_1 \xi + \omega_2 \eta)} d\omega_1 d\omega_2, \quad (23)$$

где

$$W_\alpha(\omega_1, \omega_2) = \frac{K(-\omega_1, -\omega_2)G(\omega_1, \omega_2)}{L(\omega_1, \omega_2) + \alpha M(\omega_1, \omega_2)}, \quad (24)$$

$$L(\omega_1, \omega_2) = |K(\omega_1, \omega_2)|^2 = K(\omega_1, \omega_2)K(-\omega_1, -\omega_2),$$

$$M(\omega_1, \omega_2) = (\omega_1^2 + \omega_2^2)^2 \text{ – регуляризатор, } \alpha > 0 \text{ – параметр регуляризации.}$$

Решение (17)–(18) при правильно выбранных значениях α и δ (или ρ) обладает устойчивостью и достаточной разрешающей способностью.

Выбор параметра регуляризации

Важным является вопрос о выборе значения параметра регуляризации α . Разработан ряд способов выбора α в методе регуляризации Тихонова [16, 19]. Выбор α можно осуществлять, например, способом невязки или обобщенным принципом невязки [8–9, 16, 19]. Разработаны также следующие способы выбора параметра регуляризации α : способ квазиоптимального (квазинаилучшего), способ отношения, способ независимых реализаций, способ перекрестной значимости, способ моделирования и др. [16, 20]. Однако для задачи реконструкции изображений, как показала практика, более эффективен *способ подбора* [10–13, 18]). В данной работе для задачи реконструкции дефокусированных изображений выбор α осуществлялся *способом подбора*. Согласно ему, для ряда значений α вычисляются решения $w_\alpha(\xi, \eta)$ по вышеизложенным формулам, они выводятся на дисплей в графической форме и выбирается значение α , дающее наилучшее восстановление изображения с точки зрения визуальных, физиологических (но не математических) критериев восприятия. Этот способ аналогичен способу настройки контраста телеизображения (в этом случае α обратно пропорционален контрасту). Способ подбора можно назвать также *визуальным критерием*, или критерием *качественной оценки*. Этот способ эффективен при реконструкции реальных дефокусированных изображений, когда истинное изображение w неизвестно. Если же обрабатывается смоделированное изображение, когда w известно (задается), то наряду с качественной

оценкой следует использовать также *количественную оценку* среднеквадратического отклонения (СКО) регуляризованного решения $w_\alpha(\xi, y)$ от точного w .

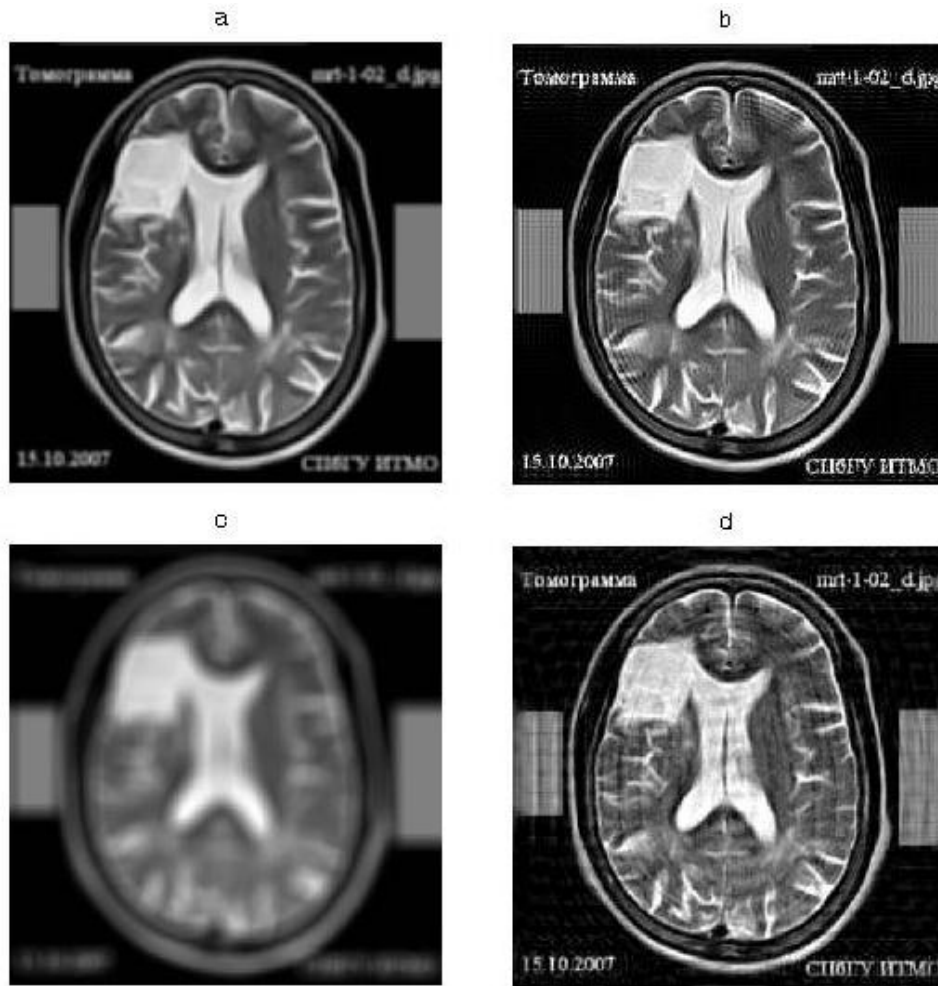


Рис. 3. Результат реконструкции дефокусированных изображений при $\rho = 3$ (a) и $\rho = 10$ (c) методом регуляризации Тихонова с $\alpha = 3,5 \cdot 10^{-5}$ (b, d)

Для количественной оценки погрешности метода реконструкции изображений использовалось *относительное среднеквадратическое отклонение* (СКО) восстановленного распределения плотности от точного распределения плотности [21]:

$$\sigma_{\text{отн}} = \frac{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (f_{ij} - f_{ij}^0)^2}}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N f_{ij}^0{}^2}}, \quad (25)$$

где f_{ij} – значения восстановленной плотности интенсивности; f_{ij}^0 – значения точной плотности; M, N – соответственно, количество строк и столбцов матрицы плотности. При компьютерном моделировании выражения (25) при помощи средств системы программирования MATLAB была составлена соответствующая программа RMSD0.m. На рис. 3

представлен результат реконструкции дефокусированных изображений (фантомов томограмм, искаженных оператором $k(x, y)$ (9) с $\rho = 3$ и $\rho = 10$ пикселей) методом преобразования Фурье и регуляризации Тихонова с параметром регуляризации $\alpha = 3,5 \cdot 10^{-5}$.

Сравнение методов реконструкции

В этой части дается сравнение известных методов реконструкции дефокусированных изображений методом преобразования Фурье и регуляризации Тихонова (далее ПФиРТ).

Оценка эффективности того или иного метода производится как субъективно (визуально), так и с помощью расчета относительного СКО. На рис. 4 представлен результат восстановления дефокусированного изображения фотографа (с $\rho = 10$ пс) методом фильтрации Винера и методом ПФиРТ. На рис. 5 представлен результат восстановления дефокусированного изображения девушки ($\rho = 10$ пс) методом фильтрации Люси–Ричардсона и методом ПФиРТ. На рис. 6 представлен результат восстановления дефокусированного изображения фотографа ($\rho = 10$ пс) методом «слепой» деконволюции и методом ПФиРТ.



Рис. 4. Результат реконструкции дефокусированного изображения:
 а) исходное изображение; б) дефокусированное изображение, $\rho = 10$;
 с) метод Винера ($\sigma_{\text{отн}} = 0,25$); д) метод ПФ и РТ, $\alpha = 10^{-3}$ ($\sigma_{\text{отн}} = 0,22$)

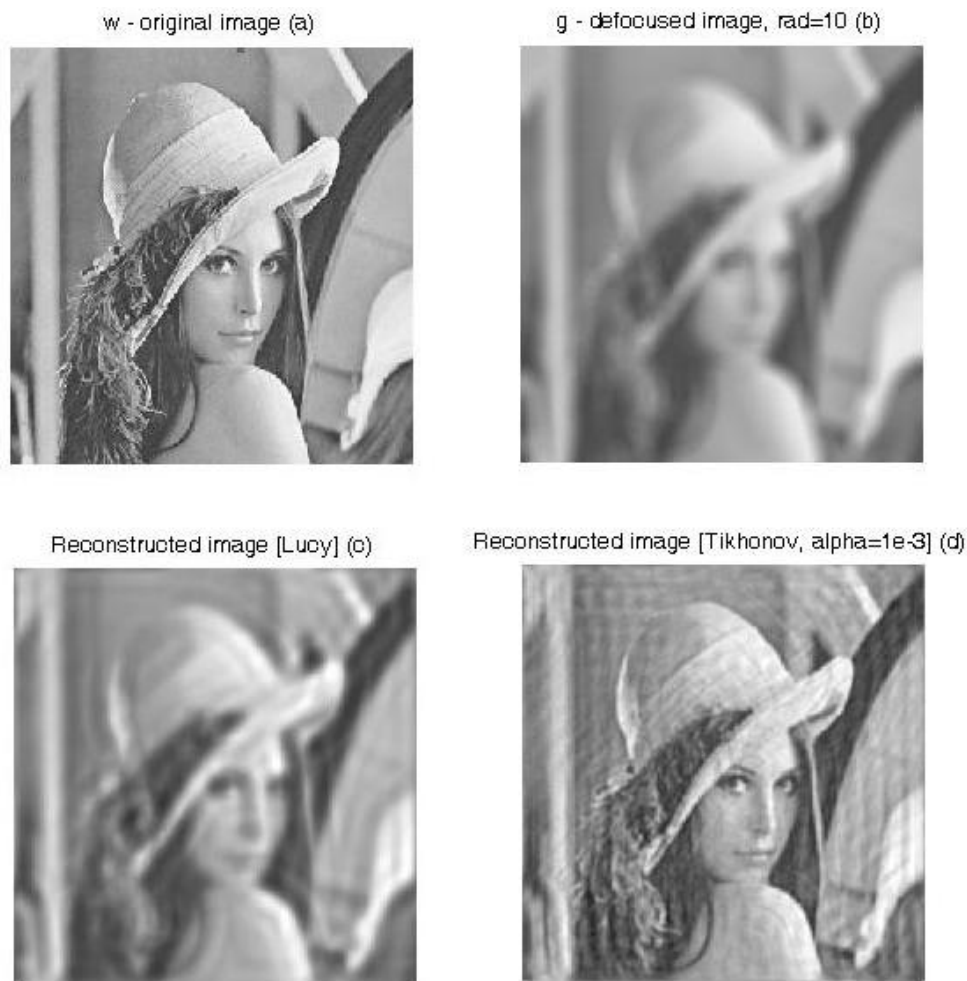


Рис. 5. Результат реконструкции дефокусированного изображения:
 а) исходное изображение; б) дефокусированное изображение, $\rho = 10$;
 с) метод Люси-Ричардсона ($\sigma_{\text{отн}} = 0,14$); д) метод ПФРТ, $\alpha = 10^{-3}$ ($\sigma_{\text{отн}} = 0,15$)

Заключение

В результате выполнения данной работы были получены следующие *результаты*.

1. В работах [4–5, 14–15] при моделировании прямой задачи для расчета интенсивностей вблизи краев изображения используется такой прием, как *граничные условия* (boundary conditions). Однако в случаях, когда этот прием создает резкие края у дефокусированного изображения, возникают помехи (эффект Гиббса и т.д.) на реконструированном изображении. Для устранения таких помех предложено искусственно моделировать размытые края у дефокусированного изображения, но без использования граничного условия типа 'replicate', 'symmetric', 'circular' [5, с. 108], что повышает точность реконструкции.

2. В работах [4, 5] используются также такие методы реконструкции дефокусированных изображений, как метод фильтрации Винера, метод регуляризации Тихонова и др. При этом как прямая, так и обратная задачи в них решаются с использованием преобразования Фурье. Однако аппарат ПФ неадекватен физической сути задачи дефокусировки, в

которой сама природа использует лишь операцию накопления (суммирования) в пределах ФРТ. Делается вывод, что наилучшие результаты должны давать методы, в которых как прямая, так и обратная задачи решаются с использованием лишь операций суммирования. Это – методы квадратур, кубатур, итераций и т.п.

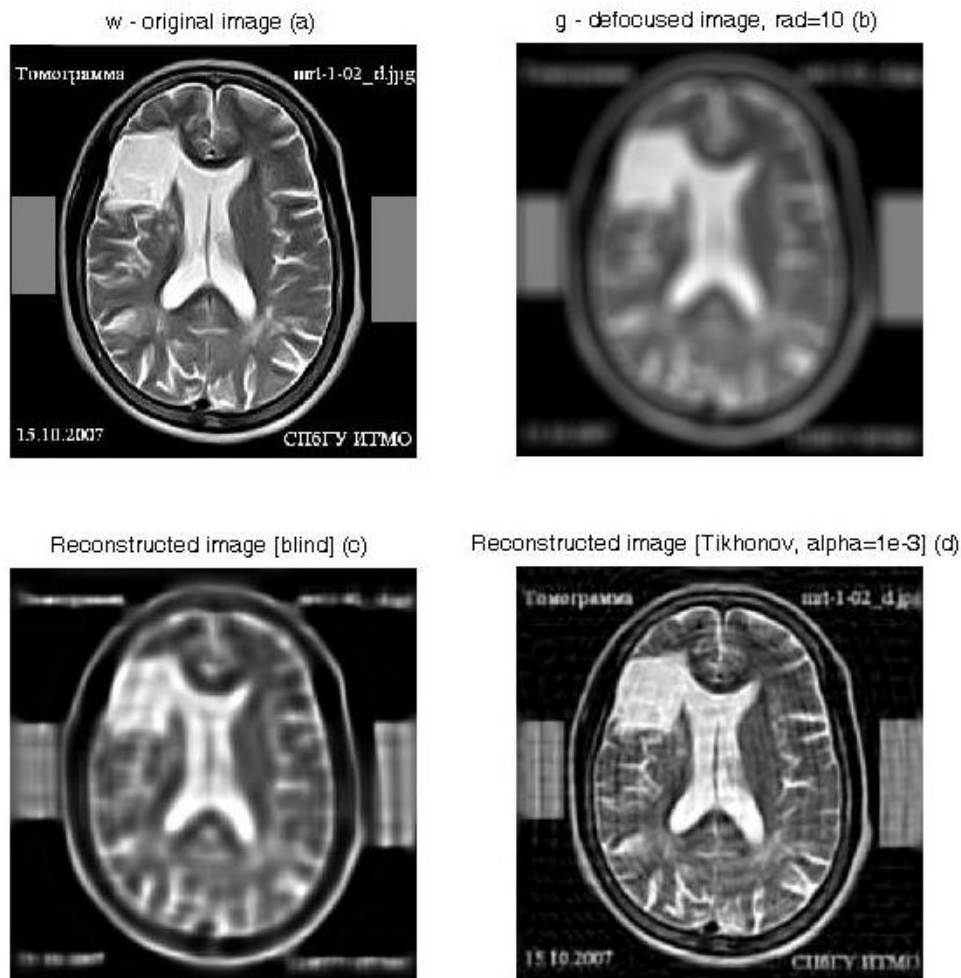


Рис. 6. Результат реконструкции дефокусированного изображения:
 а) исходное изображение; б) дефокусированное изображение, $\rho = 10$;

с) метод «слепой» деконволюции ($\sigma_{\text{отн}} = 0,37$); д) метод ПФИРТ, $\alpha = 10^{-3}$ ($\sigma_{\text{отн}} = 0,39$)

3. Рассмотрено два варианта алгоритма решения прямой задачи: с использованием операции накопления (суммирования) в пределах ФРТ и с использованием аппарата преобразования Фурье.

4. Построен устойчивый алгоритм решения обратной задачи (восстановление истинного изображения по дефокусированному изображению и функции рассеяния точки), использующий метод преобразования Фурье и метод регуляризации Тихонова.

5. Разработаны программы в виде собственных m-функций в системе MATLAB.

Литература

1. Бейтс Р., Мак-Доннелл М. Восстановление и реконструкция изображений. – М.: Мир, 1989. – 336 с.

2. Прэтт У. Цифровая обработка изображений. – М.: Мир, 1982. – Т. 2. – 792 с.
3. Грузман И.С., Киричук В.С., Косых В.П. и др. Цифровая обработка изображений в информационных системах. – Новосибирск: Изд-во НГТУ, 2000. – 168 с.
4. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2006. – 1072 с.
5. Гонсалес Р., Вудс Р., Эддинс С. Цифровая обработка изображений в среде MATLAB. – М.: Техносфера, 2006. – 616 с.
6. Арефьева М.В., Сысоев А.Ф. Быстрые регуляризирующие алгоритмы цифрового восстановления изображений // Вычислительные методы и программирование. – 1983. – Вып. 39. – С. 40–55.
7. Василенко Г.И., Тараторин А.М. Восстановление изображений. – М.: Радио и связь, 1986. – 304 с.
8. Тихонов А.Н., Гончарский А.В., Степанов В.В. Обратные задачи обработки фотоизображений // Некорректные задачи естествознания / Под ред. А.Н. Тихонова, А.В. Гончарского. – М.: Изд-во МГУ, 1987. – С. 185–195.
9. Бакушинский А.Б., Гончарский А.В. Некорректные задачи. Численные методы и приложения. – М.: Изд-во МГУ. – 1989. – 199 с.
10. Сизиков В.С., Белов И.А. Реконструкция смазанных и дефокусированных изображений методом регуляризации // Оптический журнал. – 2000. – Т. 67, № 4. – С. 60–63.
11. Сизиков В.С. Математические методы обработки результатов измерений. – СПб.: Политехника, 2001. – 240 с.
12. Петров Ю.П., Сизиков В.С. Корректные, некорректные и промежуточные задачи с приложениями: Учебное пособие для вузов. – СПб.: Политехника, 2003. – 261 с.
13. Petrov Yu.P., Sizikov V.S. Well-Posed, Ill-Posed, and Intermediate Problems with Applications. – Leiden–Boston: VSP, 2005. – 234 p.
14. Lee K.P., Nagy J.G., Perrone L. Iterative methods for image restoration: a Matlab object oriented approach, 2002. <http://www.matcs.emory.edu>
15. Donatelli M. et al. Improved image deblurring with anti-reflective boundary conditions and re-blurring // Inverse Problems. – 2006. – Vol. 22. – P. 2035–2053.
16. Верлань А.Ф., Сизиков В.С. Интегральные уравнения: методы, алгоритмы, программы. – Киев: Наук. думка, 1986. – 544 с.
17. Воскобойников Ю.Е., Литасов В.А. Устойчивый алгоритм восстановления изображения при неточно заданной аппаратной функции // Автометрия. – 2006. – Т. 42, № 6. – С. 3–15.
18. Римских М.В., Евсеев В.О., Сизиков В.С. Реконструкция смазанных изображений различными методами // Оптический журнал. – 2007. – Т. 74, № 11. – С. 53–57.
19. Тихонов А.Н., Гончарский А.В., Степанов В.В., Ягола А.Г. Численные методы решения некорректных задач. – М.: Наука, 1990. – 232 с.
20. Воскобойников Ю.Е., Преображенский Н.Г., Седельников А.И. Математическая обработка эксперимента в молекулярной газодинамике. – Новосибирск: Наука, 1984. – 240 с.
21. Пикалов В.В., Непомнящий А.В. Итерационный алгоритм с вэйвлет-фильтрацией в задаче двумерной томографии // Вычислительные методы и программирование. – 2003. – Т. 4. – С. 244–253.

ОПТИЧЕСКИЕ ХАРАКТЕРИСТИКИ МИКРОЛИНЗ, ПОЛУЧЕННЫХ МЕТОДОМ ЛАЗЕРНО-ИНДУЦИРОВАННОЙ АМОРФИЗАЦИИ СТЕКЛОКЕРАМИК

Б.Ю. Новиков

Научный руководитель – д.т.н., профессор В.П. Вейко

Продемонстрирована перспективность изготовления оптических микролинз методом лазерно-индуцированной аморфизации стеклокерамического материала. Показано влияние параметров лазерного излучения на оптические характеристики и особенности формируемых микролинз, описаны способы определения фокусного расстояния таких микролинз.

Введение

Тенденция к миниатюризации и усложнению технических устройств требует производства различных микроэлементов. В частности, в системах обработки и передачи информации, в интегральных оптических схемах, в волоконно-оптических устройствах требуются микроэлементы с определенными оптическими и геометрическими параметрами.

Традиционный механический способ не позволяет изготовить оптические элементы размером менее 1 мм. Существующие методы изготовления микрооптики можно разделить на следующие группы:

1) избирательное удаление отдельных участков оптического материала (лазерная абляция [1]; травление пучком частиц [2, 3] и др.) – профиль оптической поверхности задается оператором за счет интенсивности и направления обработки;

2) отпечатывание профиля элемента в восприимчивом оптическом материале [4] (в основном для массивов элементов) – профиль оптической поверхности повторяет профиль штампа (изготовление которого становится отдельной проблемой);

3) локальная модификация материала за счет деформации (например, под влиянием сил поверхностного натяжения [5–7]) или смены фазово-структурного состояния (молекулярная диффузия [8]; фазово-структурная перестройка за счет активации фотохимических реакций [9, 10], при термическом воздействии лазерного излучения [11, 12] и др.) – профиль оптической поверхности формируется под действием физических эффектов, может требоваться измерение параметров сформированного элемента.

К методам третьей группы относится лазерно-индуцированная локальная фазово-структурная модификация стеклокерамических материалов, которая является перспективной технологией формирования микроэлементов. Стеклокерамики обладают двумя фазовыми состояниями, радикально различными по оптическим, механическим, химическим свойствам. Использование лазерного излучения имеет ряд преимуществ перед другими инструментами обработки, а именно локальность и бесконтактность воздействия, высокоэнергетический характер излучения, возможность задания энергетических и геометрических параметров воздействия. Благодаря этим особенностям лазерного излучения модификация стеклокерамик может быть осуществлена локально, за короткий период (1–100 с), за один этап.

Такое направление данной технологии, как локальная фазово-структурная модификация ситалла СТ-50-1, возможная только под действием лазерного излучения, имеет множество практических приложений в области изготовления микрооптики. Локальная аморфизация ситалла СТ-50-1 позволяет получить оптические микроэлементы разных типов, в том числе различные микролинзы. Варьирование характеристик лазерного излучения дает широкий диапазон параметров формируемых элементов, задание определенных условий обработки обеспечивает воспроизводимость изделий. В работе проанализированы оптические характеристики микролинз, полученных данным методом.

В частности, оценено влияние параметров лазерного излучения, описаны методики определения их фокусного расстояния.

Методика формирования микролинз

Ситалл СТ-50-1 представляет собой промышленно выпускаемый материал, характеризующийся большим количеством хаотично расположенных по отношению друг к другу микрокристаллов размером 1–2 мкм и отсутствием пористости. В составе ситалла преобладает SiO_2 – распространенный оптический материал. Однако ситалл непрозрачен в видимом и ближнем ИК диапазоне за счет сильного рассеяния на границах микрокристаллов, размер которых эквивалентен этим длинам волн [13].

Под действием высокоэнергетического лазерного излучения локальная область ситалла может быть нагрета до температуры плавления (1473 К), тогда все упорядоченные связи между атомами исчезают, и ситалл просветляется. При этом модифицированная область занимает весьма малый участок по сравнению с объемом всей заготовки, что позволяет сохранить аморфное состояние материала за счет особенностей остывания ванны расплава. Теплопроводность в твердом теле является значительно более эффективным механизмом охлаждения, чем естественное воздушное охлаждение или водяное (жидкостное) охлаждение. После прекращения лазерного воздействия за счет теплоотвода в окружающие участки исходного материала модифицированная область ситалла охлаждается до температуры стеклования (1033 К) настолько быстро (50–100 К/с), что атомы не успевают перегруппироваться в упорядоченную структуру и высокотемпературное аморфное состояние «замораживается» [13–18].

Первоначальная кристаллическая структура обладает более высокой удельной плотностью по сравнению с аморфной структурой, поэтому объем материала в лазерномодифицированной зоне увеличивается. За счет влияния сил поверхностного натяжения образуется линзообразная форма поверхности аморфизированной области [14, 17, 18].

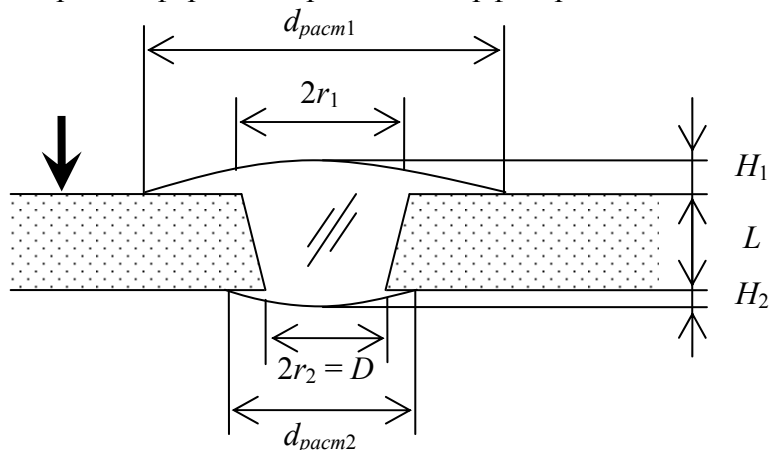


Рис. 1. Схематическое изображение поперечного среза локальной лазерно-аморфизированной области на пластине ситалла СТ-50-1, где r – радиус аморфизированной области (r_1 может быть больше r_2), $d_{\text{расм}}$ – размер области растекания расплава по поверхности пластины, H – стрелка прогиба, L – толщина пластины. Стрелкой показана сторона, с которой производилось воздействие лазерного излучения

Таким образом, под действием лазерного излучения в непрозрачном исходном ситалле может быть сформирована оптически прозрачная область с линзообразной поверхностью. При сквозной аморфизации тонких плоскопараллельных пластин могут быть сформированы двояковыпуклые микролинзы [13, 14, 17, 18–20] (рис. 1), получение которых возможно только с использованием данного метода. Данный метод также

может быть использован для формирования геодезических микролинз [13, 14] и линз с различной формой зрачка (при использовании маски) [13, 14, 17]. Сканирование лазерного луча по поверхности ситалла позволяет получить просветленные каналы и цилиндрические линзы [13, 14, 17, 18]. Воздействие лазерного излучения на вращающуюся заготовку ситалла позволяет формировать аморфизированные области с геометрическими профилями поверхностей, соответствующих рассеивающим линзам, линзам, обладающим кольцевым фокусом, с асферическими выпуклостями (например, параболическими) [13].

Влияние параметров лазерного излучения

Форма оптических поверхностей микролинз определяет их фокусное расстояние, размер микролинзы (окруженной непрозрачным исходным ситаллом) – размер области пропускания. Эти геометрические параметры могут варьироваться при лазерно-индуцированном формировании микролинз.

На геометрические параметры аморфизованных областей ситалла СТ-50-1 оказывают влияние следующие параметры лазерного излучения:

- длина волны излучения выбранного лазера;
- плотность мощности излучения;
- размер и форма поперечного сечения пучка излучения в плоскости обработки;
- время воздействия.

В настоящее время локальная аморфизация ситалла СТ-50-1 исследована при использовании излучений CO_2 -лазера (10,6 мкм) [13–21] и YAG:Nd-лазера (1,06 мкм) [22, 23]. Разные механизмы взаимодействия излучений этих лазеров с различными фазами ситалла СТ-50-1 влияют на протекание процесса модификации и обуславливают получение аморфизированных областей с различными геометрическими параметрами. На рис. 2 приведена кинетика (на начальном этапе модификации) диаметра и глубины аморфизованных областей от времени при одинаковых условиях воздействия излучений CO_2 - и YAG:Nd-лазеров. Были приложены эквивалентные поглощенные плотности мощности излучений q , определенные из соотношения: $q_{\text{CO}_2} \cdot A_{\text{CO}_2} \cdot \delta_{\text{CO}_2} = q_{\text{YAG:Nd}} \cdot A_{\text{YAG:Nd}} \cdot \delta_{\text{YAG:Nd}}$, где A – коэффициент отражения, δ – глубина проникновения излучения. Тогда при $A_{\text{CO}_2} = 0,9$; $A_{\text{YAG:Nd}} = 0,5$; $\delta_{\text{CO}_2} = 1,25$ мкм; $\delta_{\text{YAG:Nd}} = 105$ мкм [13, 15, 17] имеем $q_{\text{CO}_2} = 10^7$ Вт/м² и $q_{\text{YAG:Nd}} = 18 \cdot 10^7$ Вт/м². Размер поперечных сечений пучков излучений в плоскости обработки составлял 1,5 мм, для предотвращения терморазрушений (из-за высоких температурных градиентов) производился дополнительный подогрев до температуры ≈ 850 К. Различия в кинетике роста аморфизированных областей под действием рассматриваемых излучений очевидны.

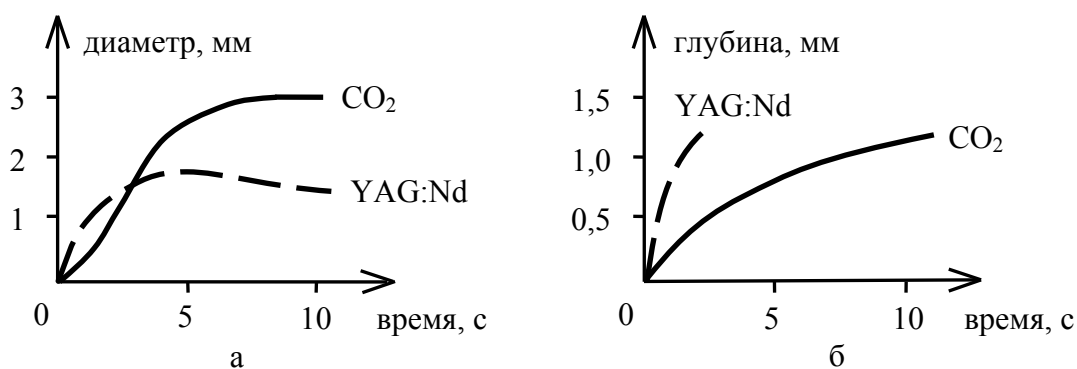


Рис. 2. Качественный ход зависимостей диаметра (а) и глубины (б) аморфизованной области (на начальном этапе модификации) от времени при одинаковых условиях воздействия излучений CO_2 - и YAG:Nd-лазеров

Излучение CO_2 -лазера поглощается в поверхностном слое ситалла СТ-50-1 (1,25 мкм [13]), поэтому расплавление материала в этом случае происходит исключительно за счет теплопроводности. Изотерма плавления движется (т.е. области аморфизации развиваются) одинаково по всем направлениям.

Под действием излучения YAG:Nd-лазера сообщение материалу энергии, необходимой для нагрева до температуры плавления, происходит за счет остаточного поглощения при многократном отражении излучения от границ микрокристаллов. Аморфная фаза (расплав) свободно пропускает излучение данной длины волны. Взаимодействие излучения YAG:Nd-лазера с ситаллом происходит на границе поликристаллического материала в основном по ходу луча. Такое смещение зоны поглощения (и изотермы плавления) преимущественно вглубь материала под действием излучения YAG:Nd-лазера позволяет формировать аморфизированные области без ограничений их геометрических параметров.

Стрелки прогиба могут уменьшаться при превышении некоторых значений плотности мощности излучения и времени воздействия за счет постепенного растекания расплава по поверхности пластины.

Размер аморфизированной области прямо пропорционально зависит от времени воздействия, размера пучка и плотности мощности излучения. Это объясняется тем, что данные параметры эквивалентны общему количеству сообщенной материалу энергии. По сравнению с теплопроводным характером взаимодействия излучения CO_2 -лазера с ситаллом СТ-50-1, при поэтапном просветлении ситалла под действием излучения YAG:Nd-лазера эти параметры оказывают менее интенсивное влияние. Поглощение в расплаве на длине волны излучения YAG:Nd-лазера мало, следовательно, сформированная ванна расплава под действием данного излучения перестает нагреваться, и теплоотвод в прилегающие боковые зоны ситалла будет недостаточен для достижения плавления.

Отметим, что прекращение нагрева и остывание ванны расплава под действием излучения YAG:Nd-лазера также может приводить к ухудшению оптического качества формируемых микроэлементов из-за разделения фаз, формирования вторичных кристаллических структур, возникновения градиента показателя преломления.

Оптические характеристики микролинз

Полученные под действием лазерного излучения аморфизированные области ситалла СТ-50-1 обладают пропусканием в видимом и ближнем инфракрасном диапазоне (рис. 3) [14–16, 20, 21].

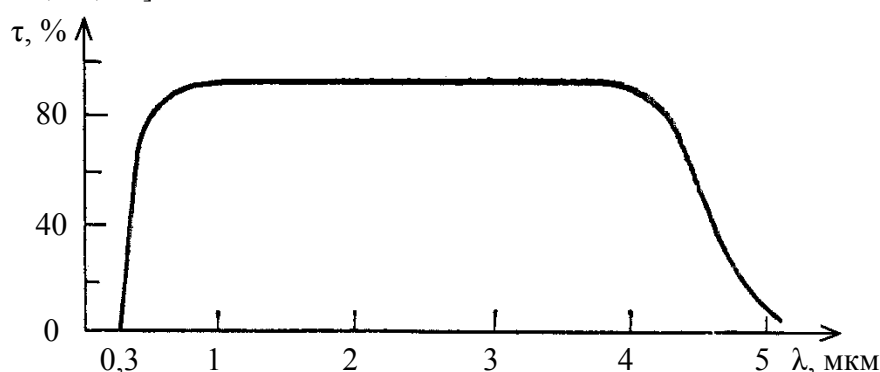


Рис. 3. Спектр пропускания в аморфизированной области ситалла СТ-50-1 толщиной 0,3 мм [20]

Наши исследования показали независимость величины пропускания аморфизированных областей данного ситалла от условий их лазерно-индуцированного формирова-

ния: длины волны (применялись излучения CO₂- и YAG:Nd-лазеров), плотности мощности, диаметра пучка лазерного излучения, времени воздействия. Можно сказать, что все остальные физико-химические свойства аморфизированных областей (т.е. свойства самой аморфной фазы) также будут одинаковы вне зависимости от характеристик используемого лазерного излучения.

Идентичность изображений, снятых при помощи растра двояковыпуклых микролинз (рис. 4), показывает высокие информационные характеристики линз и подтверждает воспроизводимость, обеспечиваемую описанным методом.

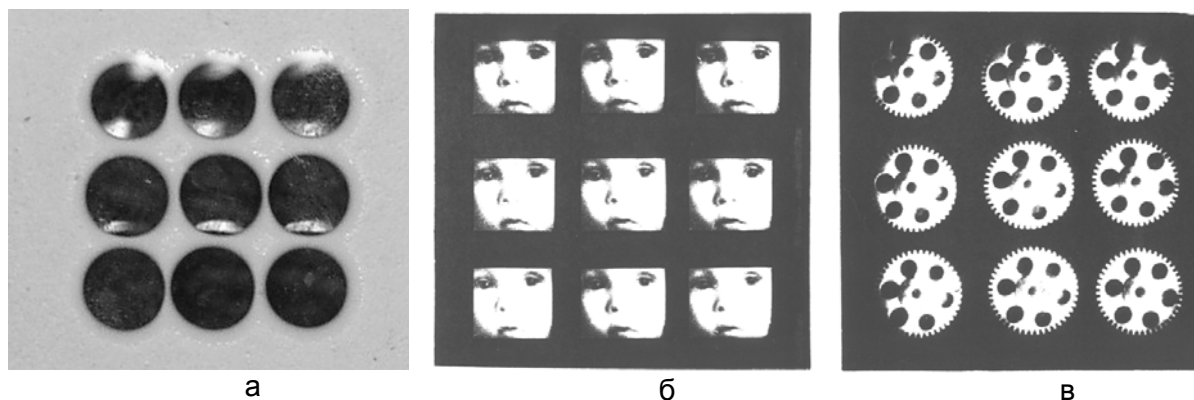


Рис. 4. Растр двояковыпуклых микролинз, полученный в пластине ситалла СТ-50-1 методом лазерно-индуцированной локальной аморфизации (а), и изображения объектов, снятых с его помощью (б, в). Фокусное расстояние микролинз ≈ 21 мм. Параметры режима обработки: плотность мощности излучения CO₂-лазера $\approx 2 \cdot 10^6$ Вт/м²; перед облучением используется дополнительный подогрев до температуры 1000 К. Толщина пластин ситалла 0,6 мм [14]

В наших экспериментах при определенных условиях освещения в аморфизованных областях ситалла СТ-50-1 наблюдался градиент показателя преломления, который может возникнуть под действием излучения YAG:Nd-лазера (рис. 5). Отметим, что в аморфизованных областях ситалла, сформированных под действием излучения CO₂-лазера, подобной оптической неоднородности аморфной фазы не обнаружено.

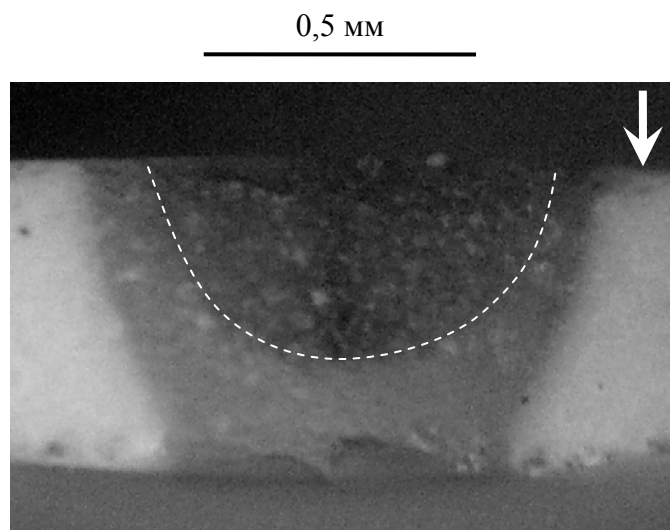


Рис. 5. Градиент показателя преломления (граница зон с разными показателями преломления отмечена пунктиром) аморфизованной области ситалла СТ-50-1, сформированной под действием излучения YAG:Nd-лазера. Фотография поперечного шлифа аморфизованной области на пластине. Стрелкой показана сторона, с которой производилось воздействие излучения YAG:Nd-лазера

Наличие градиента показателя преломления как в радиальном, так и осевом направлениях может быть объяснено, исходя из особенностей формирования аморфизованных областей ситалла СТ-50-1 под действием излучения YAG:Nd-лазера. Аморфизованная область формируется послойно, температура каждого слоя ненамного превышает температуру плавления. Вероятно, при сквозном просветлении пластины ситалла уменьшается нагрев той части ванны расплава, которая противоположна воздействию лазерного излучения. Разные участки аморфной области подвергаются различному отжигу. Эксперименты показывают, что максимальный градиент показателя преломления возникает в слое, граничащем с той поверхностью пластины, с которой производилось воздействие лазерного излучения при формировании микролинзы, а минимальный – в слое, граничащем с противоположной поверхностью пластины. Наличие градиента показателя преломления оказывает некоторое влияние на фокусное расстояние микролинз.

Определение фокусного расстояния микролинз

Определение фокусного расстояния микролинз очень важно для их практического применения. Как уже говорилось, оптическая сила микролинз определяется кривизной их оптических поверхностей, форма которых определяется поведением расплава. Поведение расплава, помимо действия сил поверхностного натяжения, обуславливается размером ванны расплава (который, в свою очередь, зависит от параметров лазерного излучения: диаметра пучка и плотности мощности излучения, времени облучения материала).

В [20] предложен метод определения фокусных расстояний двояковыпуклых микролинз, исходя из размера аморфизированной области и отношения плотностей исходного материала ρ_0 и аморфной фазы ρ . Для простоты рассуждений принимались следующие допущения: (1) форма свободной поверхности расплава в поле силы тяжести мало отличается от сферической; (2) аморфизированная область имеет цилиндрическую форму радиуса r ; (3) вследствие быстрого затвердевания расплава $\rho = \rho_p$, где ρ_p – плотность расплава. Тогда из условия сохранения энергии при изменении величины площади свободной поверхности расплава под действием силы тяжести фокусное расстояние линзы приближенно может быть определено из соотношения:

$$F = \frac{[r^2 + (h - h')^2][r^2 + (h + h')^2]}{4(n_0 - 1)(r^2 + h^2)h},$$

где

$$h = \left(\beta r^2 d + r^2 \sqrt{r^2 + \beta^2 d^2} \right)^{1/3} + \left(\beta r^2 d - r^2 \sqrt{r^2 + \beta^2 d^2} \right)^{1/3};$$

$$h' = \frac{\rho_0 g}{2\sigma\alpha} \left[(r^2 + h^2) \left(\frac{d}{2} + h_0 \right) + \gamma h (3r^2 + h^2) \right]; \quad h_0 = \frac{h(2r^2 + h^2)}{2(3r^2 + h^2)};$$

$$\gamma = \frac{2r^2 + 3h^2}{12(3r^2 + h^2)}; \quad \alpha = \frac{\rho_0}{\rho}; \quad \beta = \frac{3}{2}(\alpha - 1);$$

n_0 – показатель преломления аморфной фазы ситалла СТ-50-1 ($n_0 = 1,585$ [20]), d – толщина пластины; g – гравитационная постоянная; σ – коэффициент поверхностного натяжения расплава.

Аналогичный способ определения фокусных расстояний микролинз, основанный на законах сохранения, предложен в [13, 14]. Радиусы кривизны поверхностей R_1 и R_2 микролинзы вычисляются, исходя из уравнения баланса сил, действующих на облученную зону во время лазерного воздействия,

$$\frac{\sigma S_1}{R_1} + \frac{mg}{2} = \frac{\sigma S_2}{R_2}$$

и уравнения сохранения масс

$$m_{\text{аморф}} = m_{\text{крист}} = m;$$

где m – масса расплава, равная массе аморфной фазы $m_{\text{аморф}}$, полученной при модификации, и массе исходной кристаллической фазы $m_{\text{крист}}$, подвергшейся модификации; S_1 и S_2 – площади верхней и нижней расплавленных поверхностей. Алгоритм нахождения R_1 и R_2 из приведенных соотношений подробно разобран в [13].

Автором также были проведены исследования по определению фокусного расстояния двояковыпуклых микролинз, которые были экспериментально сформированы на пластине ситалла СТ-50-1 под действием лазерного излучения. Как и в других методиках, было допущено, что геометрический профиль поверхности сформированной линзы близок к сферическому. При помощи методов микроскопии было проведено измерение таких параметров микролинзы, как стрелка прогиба H и размер области растекания расплава по поверхности пластины $d_{\text{расч}}$ ($d_{\text{расч}}$ может быть больше $2r$, см. рис. 1). Значения R_1 и R_2 определялись по формуле, полученной на основе геометрических соотношений:

$$R = \frac{d_{\text{расч}}^2 + 4H^2}{8H}.$$

Для расчета фокусного расстояния микролинз $f'_{\text{расч}}$ по известным R_1 и R_2 можно использовать известное выражение (в [4] данное выражение упрощено):

$$\frac{1}{f'_{\text{расч}}} = (n_0 - n_1) \left(\frac{1}{R_1} - \frac{1}{R_2} \right) + \frac{(n_0 - n_1)^2}{n_0 R_1 R_2} Z,$$

где n_1 – показатель преломления окружающей среды (воздух, $n_1 = 1$), Z – толщина микролинзы. В нашем случае $Z = L + H_1 + H_2$, где $L = 0,6$ мм – толщина пластины. Способ определения фокусных расстояний плосковыпуклых микролинз методом расчета на основании геометрических параметров также был предложен в [24].

Также автором были выполнены измерения фокальных отрезков тех же микролинз по методике, описанной в [25], установка для проведения измерений показана на рис. 6. Для удобства визуального наблюдения измерительный пучок излучения He-Ne лазера (1) пропускается через ослабитель (2). Пройдя через микролинзу (4), измерительный пучок фокусируется на подвижный просвечивающий экран (5), который может перемещаться вдоль оптической оси. Расстояние между задней поверхностью микролинзы (4) и экраном (5), при котором диаметр поперечного сечения пятна излучения будет минимален, будет являться фокальным отрезком. Скрепленный с экраном (5) микрометр (6) позволяет определить величину перемещения экрана. Недостатком данного метода является визуальное определение момента фокусировки, но это облегчается тем, что фотокамера (7) делает снимки на каждом шаге экрана. Этот способ измерения фокальных отрезков микролинз также позволяет одновременно определить диаметр $d_{\text{изм}}$ поперечного сечения пучка излучения, прошедшего через микролинзу (4), на любом расстоянии от нее до экрана (5).

Измерения фокальных отрезков проводились при следующих условиях:

а) измерительный пучок сфокусирован посредством вспомогательной линзы (3) в пятно размером 0,35 мм и проходит через микролинзу (4) без потерь (S_{F0});

б) несфокусированный измерительный пучок диафрагмировался на исходной непрозрачной фазе ситалла СТ-50-1 при вводе с разных сторон микролинзы (S_{F1} и S_{F2}).

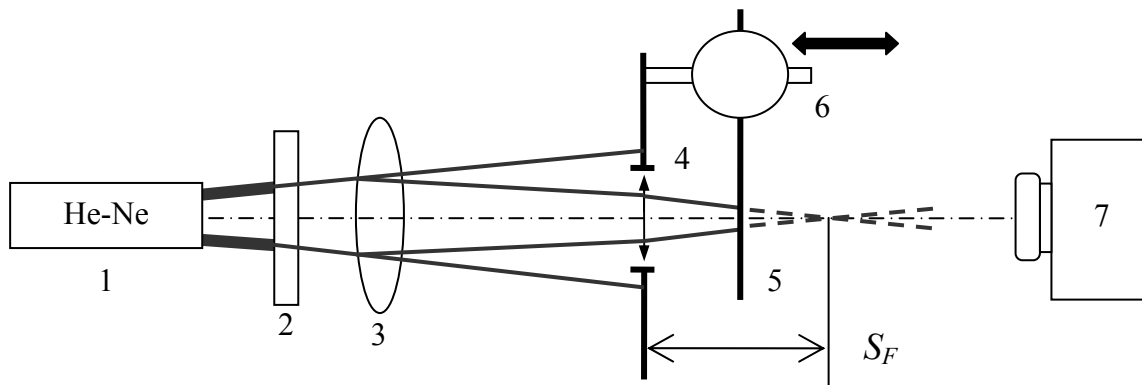


Рис. 6. Схема установки для измерения фокальных отрезков двояковыпуклых микролинз

Расчетные и измеренные параметры микролинз приведены в табл. 1 и 2. Для полноты картины здесь также приведены диаметры поперечного сечения пучка излучения в фокальной плоскости $d_{изм0}$, $d_{изм1}$ и $d_{изм2}$, которые соответствуют измеренным S_{F0} , S_{F1} и S_{F2} , и размер микролинзы на просвет (т.е. наименьший диаметр аморфной области в осевом направлении) D ($D = 2r_2$, см. рис. 1).

№ П/П	$f'_{расч} \pm$ $\pm \Delta f'_{расч}$, ММ	$S_{F0} \pm$ $\pm 0,5$, ММ	$S_{F1} \pm$ $\pm 0,5$, ММ	$S_{F2} \pm$ $\pm 0,5$, ММ	$D \pm$ $\pm 0,05$, ММ	$d_{изм0} \pm$ $\pm 0,17$, ММ	$d_{изм1} \pm$ $\pm 0,17$, ММ	$d_{изм2} \pm$ $\pm 0,17$, ММ
1	$4,8 \pm 0,3$	4	5	5	0,56	0,27	0,25	0,29
2	$5,4 \pm 0,5$	4	7	10	0,7	0,31	0,28	0,31
3	$6,6 \pm 0,6$	5	7	6	0,7	0,33	0,26	0,27
4	$7,9 \pm 0,8$	6	8	6	0,61	0,34	0,29	0,35
5	$8,5 \pm 0,8$	7	9	7	0,85	0,33	0,34	0,43
6	$10,6 \pm 1,1$	6	11	11	0,85	0,3	0,28	0,28

Таблица 1. Параметры микролинз, сформированных под действием излучения YAG:Nd-лазера

№ П/П	$f'_{расч} \pm$ $\pm \Delta f'_{расч}$, ММ	$S_{F1} \pm$ $\pm 0,25$, ММ	$S_{F2} \pm$ $\pm 0,25$, ММ	$f'_{изм1} \pm$ $\pm \Delta f'_{изм1}$, ММ	$f'_{изм2} \pm$ $\pm \Delta f'_{изм2}$, ММ	$D \pm$ $\pm 0,05$, ММ	$d_{изм1} \pm$ $\pm 0,17$, ММ	$d_{изм2} \pm$ $\pm 0,17$, ММ
1	$4 \pm 0,1$	3,5	3,5	$3,7 \pm 0,1$	$3,7 \pm 0,1$	0,94	0,31	0,22
2	$4,3 \pm 0,2$	3,5	3,5	$3,6 \pm 0,1$	$3,6 \pm 0,1$	1,13	0,29	0,17
3	$14,7 \pm 1,1$	14,5	14	$14,6 \pm 0,8$	$14,1 \pm 0,8$	0,66	0,24	0,22

Таблица 2. Параметры микролинз, сформированных под действием излучения CO₂-лазера

Для микролинз, сформированных под действием излучения YAG:Nd-лазера, существенное расхождение S_{F0} , S_{F1} и S_{F2} может быть объяснено наличием градиента показателя преломления. В этом случае анизотропность аморфной области заранее обуславливает неточность расчета $f'_{расч}$ и не допускает возможности на основе измерения S_F корректно определить фокусные расстояния $f'_{изм}$ микролинз.

Для микролинз, сформированных под действием излучения CO₂-лазера, определение $f'_{изм}$ производилось на основе известного соотношения:

$$S'_{F'} = f' \left(1 - \frac{(n_0 - 1)Z}{nR_1} \right).$$

Отдельные незначительные расхождения S_{F1} и S_{F2} могут объясняться конусообразной формой аморфизированной области, обусловленной механизмом взаимодействия излучений данного лазера с ситаллом. Некоторые незначительные расхождения $f'_{расч}$ и $f'_{изм}$, вероятно, связаны с погрешностями использованных методов измерения. В [26] говорится о незначительном микрорельефе, возникающем на границах области лазерно-индуцированной модификации силикатного материала. Наличие такого микрорельефа может оказывать влияние на поведение расплава, тогда для наиболее точного определения профиля поверхности микролинз лучше применять не методы микроскопии, а получать профилограммы.

Заключение

Лазерно-индуцированная локальная аморфизация ситалла СТ-50-1 является перспективным методом для формирования разных типов оптических микроэлементов. Такие микроэлементы обладают пропусканием в оптическом диапазоне, их геометрические параметры (некоторые из которых определяют оптическую силу микролинз) могут варьироваться за счет задания характеристик лазерного излучения. Данный метод показывает высокую воспроизводимость.

Может быть предложено несколько способов определения фокусного расстояния микролинз, сформированных данным методом. Показано, что существенный вклад в фокусное расстояние микролинз вносит наличие градиента показателя преломления, который может возникать при использовании излучения YAG:Nd-лазера.

Задание определенных условий обработки позволит формировать оптические микроэлементы с заданными параметрами.

Литература

1. K. Naessens, H. Ottevaere, R. Baets, P. Van Daele, H. Thienpont. Direct writing of microlenses in polycarbonate with excimer laser ablation // *Appl. Opt.* – 2003. – Vol. 42, № 31. – P. 6349–6359.
2. M.E. Motamedi, W.H. Southwell, R.J. Anderson, L.G. Hale, W.J. Gunning. High speed binary optic microlens array in GaAs // *Proc. SPIE.* – 1991. – Vol. 1544. – P. 33–44.
3. T. Werner, J. A. Cox, S. Swanson. Microlens array for staring infrared imager // *Proc. SPIE.* – 1991. – Vol. 1544. – P. 46–52.
4. U. Teubner. Micro-optical components and systems in polymers for optical networks and sensors // *Proc. SPIE.* – 2004. – Vol. 5454. – P. 31–40.
5. T. Yamabe, H. Murotani, K. Nakamoto, M. Wakaki. Composition change of glass materials in microlens formed by CO₂ laser // *J. Laser Micro/Nanoeng.* – 2007. – Vol. 2, № 1. – P. 40–43.
6. Вейко В.П., Вознесенский Н.Б., Вознесенская Н.Н., Пашин В.Ф., Петров А.А. Гибридные оптические схемы: исследование проблемы оптических межсоединений на основе классического подхода // *Изв. вузов. Приборостр-е.* – 2004. – Т. 47, № 10. – С. 5–14.
7. Вейко В.П., Кieu К., Rho S. Лазерное изготовление оптических микролинз и микросфер // *Изв. вузов. Приборостр-е.* – 2004. – Т. 47, № 10. – С. 30–35.
8. B. Volckaerts, P. Vynck, M. Vervaeke, al. Basic aspects of deep lithography with particles for the fabrication of micro-optical and micro-mechanical structures // *Proc. SPIE.* – 2004. – Vol. 5454. – P. 52–63.

9. Y. Cheng, K. Sugioka, K. Midorikawa. Freestanding optical fibers fabricated in a glass chip using femtosecond laser micromachining for lab-on-a-chip application // *Opt. Exp.* – 2005. – V. 13, № 18. – P. 7225–7232.
10. Yongkang Guo, Qinjun Peng, Jianhua Zhu, Yangsu Zeng, Shijie Liu. Micro optical elements made of silver-halide sensitized gelatin etched by enzyme with real-time mask // *Proc. SPIE.* – 2002. – Vol. 4924. – P. 69–75.
11. O. Chibulka, V. Chab. Phase transformations in CDZnTe crystals induced by nanosecond laser irradiation // *Proc. SPIE.* – 2006. – Vol. 6161. – P. 61610B-1–61610B-8.
12. Петров Д.В., Яковлев Е.Б., Костюк Г.К., Вейко В.П., Антропова Т.В. Исследование условий формирования областей с измененным показателем преломления на поверхности пластин пористого стекла излучением CO₂-лазера // *Изв. вузов. Приборостр-е.* – 2004. – Т. 47, № 10. – С. 26–29.
13. Скиба П.А. Лазерная модификация стекловидных материалов. – Минск: БГУ, 1999. – 131 с.
14. Вейко В.П., Киеу К.К. Лазерная аморфизация стеклокерамик: основные закономерности и новые возможности изготовления микрооптических элементов // *Квант. эл-ка.* – 2007. – Т. 37, № 1. – С. 92–98.
15. Вейко В.П., Никоноров Н.В., Скиба П.А. Фазово-структурная модификация стеклокерамики, индуцированная лазерным излучением // *Опт. журнал.* – 2006. – Т. 73, № 6. – С. 64–70.
16. Veiko V.P., Kieu Q.K., Nikonorov N.V., Skiba P.A. On the reversibility of laser-induced phase-structure modification of glass-ceramics // *J. Laser Micro/Nanoeng.* – 2006. – V. 1, № 2. – P. 149–154.
17. Veiko V.P., Kieu Q.K., Nikonorov N.V. Laser modification of glass-ceramics structure and properties: a new view to traditional materials // *Proc. SPIE.* – 2004. – V. 5662. – P. 119–128.
18. Skiba P.A., Volkov V.P., Predko K.G., Veiko V.P. Laser-stimulated local change of glass-ceramic optical properties // *Opt. Eng.* – 1994. – Vol. 33, № 11. – P. 3572–3577.
19. Скиба П.А., Каранчук Д.Я. Оптимизация процессов лазерного формирования оптических элементов на основе стеклокерамики // *Изв. вузов. Приборостр-е.* – 2004. – Т. 47, № 10. – С. 14–20.
20. Скиба А.П., Волков В.П., Сечко А.Г., Непокойчицкий А.Г., Емельянов А.В., Бондаренко А.В. Оптические элементы, полученные методом аморфизации ситаллов // *Опт.-мех. пром-ть.* – 1991. – № 9. – С. 59–62.
21. Q.K. Kieu, V.Yu. Novikov, E.A. Shakhno, E.B. Yakovlev. Physical mechanisms of fast structure modification of glass-ceramics under CO₂-laser action // *Proc. SPIE.* – 2007. – V. 6458. – Опубликовано online на сайте «SPIE.org». Режим доступа: <http://spie.org/x10.xml>, платный.
22. Вейко В.П., Новиков Б.Ю., Шахно Е.А., Яковлев Е.Б. Нелинейное просветление ситалла излучением YAG:Nd-лазера // *Изв. вузов. Приборостр-е.* – 2007. – Т. 50, № 7. – С. 14–19.
23. Новиков Б.Ю. Формирование микролинз методом аморфизации стеклокерамики при использовании YAG:Nd-лазера // *Изв. РАН. Серия физ.* – 2006. – Т. 70, № 9. – С. 1323–1326.
24. Ph. Nussbaumu, R. Völkel, H.P. Herzig, M. Eisner, S. Haselbeck. Design, fabrication and testing of microlens arrays for sensors and microsystems // *Pure Appl. Opt.* – 1997. – № 6. – P. 617–636.
25. Гвоздева Н.П., Коркина К.И. Прикладная оптика и оптические измерения (учебник для техникумов). – М.: Машиностроение, 1976. – 383 с.
26. Макин В.С., Пестов Ю.И. Особенности рельефа, формируемого на поверхности кварцевого стекла под действием движущего луча непрерывного CO₂-лазера // *Опт. журнал.* – 2005. – Т. 72, № 11. – С. 78–81.

РАСЧЕТ УСИЛИЯ В КЛЕММНЫХ СОЕДИНЕНИЯХ

А.В. Амвросьева

Научный руководитель – д.т.н., профессор В.М. Мусалимов

До второй половины прошлого века в электромонтаже доминировали винтовые клеммные соединители. Их конструкция постоянно совершенствовалась, однако от некоторых недостатков не удалось избавиться до сих пор. Главный из них – зависимость качества монтажа от квалификации персонала: усилие, с которым следует зажать в клемме проводник, монтажник определяет буквально на ощупь. С появлением пружинных клемм качество монтажа перестало напрямую зависеть от умения обращаться с отверткой. Целью работы автора является исследование пружинного элемента как объекта для проектного расчета.

Математическая модель клемм

Для достижения цели нужно выбрать необходимый тип зажимного элемента, создать для него математическую модель для дальнейшего расчета его характеристик. В работе рассматривается плоскопружинный зажим в клемме для строительного монтажа, на который оказывается воздействие извне, изгибающее его. Плоскопружинный зажим и расчетная схема представлены на рис. 1, 2.

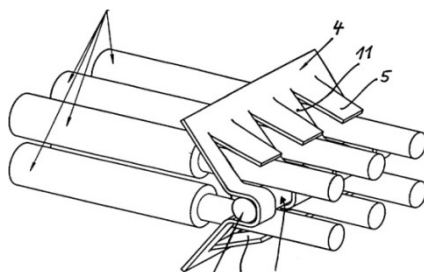


Рис. 1. Упругий элемент клеммы для строительного монтажа



Рис. 2. Расчетная схема

В клемме для строительного монтажа упругий элемент расположен, как показано на рис. 3: точка Р – первый контакт провода и пружины, точка Q – конечный контакт провода и пружины.

Рассмотрим деформацию пружины при плоском изгибе. Ось пружины под действием нагрузки искривляется в плоскости действия сил (плоскость xOy), при этом поперечные сечения поворачиваются и смещаются на некоторую величину (рис. 4).

Деформацию пружины при изгибе будем описывать двумя параметрами:

1) прогиб (y) – смещение центра тяжести сечения пружины по направлению, перпендикулярному к ее оси. Наибольший прогиб пружины называется стрелой прогиба ($f=y_{max}$);
2) угол поворота сечения (φ) – угол, на который сечение поворачивается относительно своего первоначального положения (или угол между касательной к упругой линии и первоначальной осью пружины).

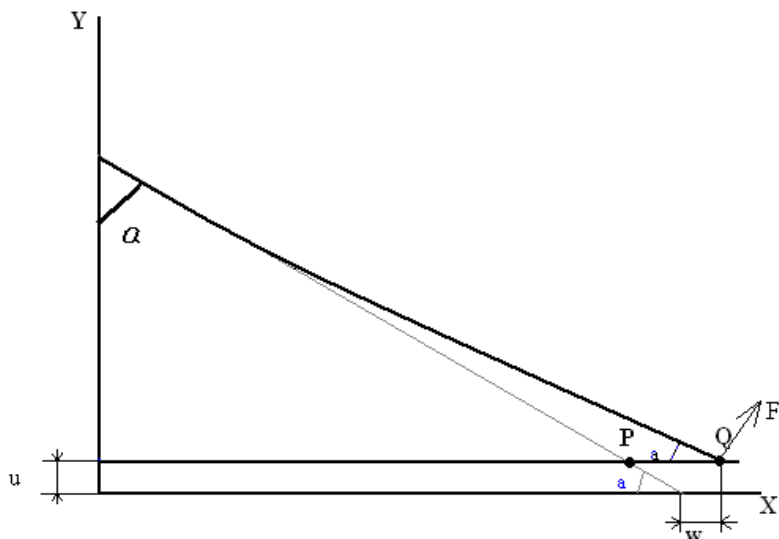


Рис. 3. Геометрия расположения упругого элемента в клемме

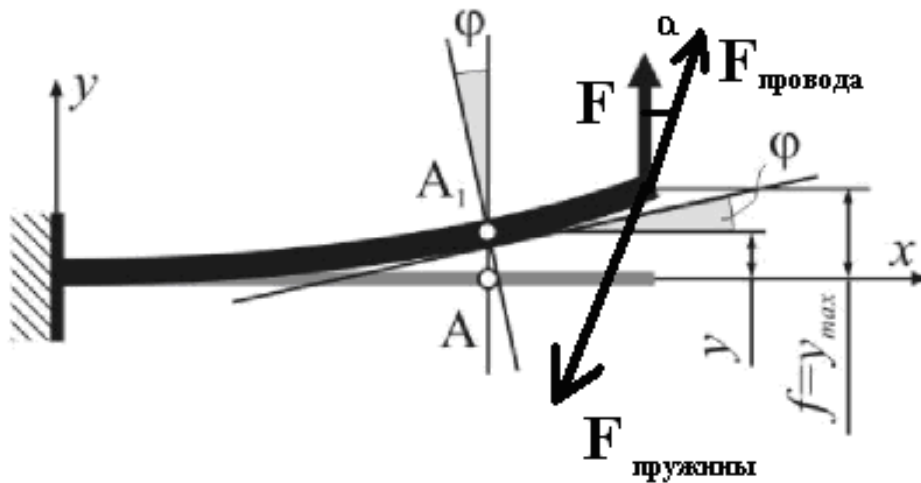


Рис. 4. Распределение сил

В общем случае величина прогиба пружины в данной точке является функцией координаты x и может быть записана в виде уравнения

$$y = y(x). \quad (1)$$

Тогда угол между касательной к изогнутой оси пружины и осью x будет определяться из выражения

$$\tan \varphi = \frac{\partial y}{\partial x}. \quad (2)$$

Ввиду малости углов и перемещений можем считать, что

$$\varphi = \frac{\partial y}{\partial x}, \quad (3)$$

т.е. угол поворота сечения есть первая производная от прогиба пружины по абсциссе сечения. Кривизна пружины равна (рис. 5)

$$\frac{1}{\rho} = \frac{M}{E \cdot J}, \quad (4)$$

где M – момент инерции пружины ($\text{кг} \cdot \text{мм}^2$); E – модуль Юнга ($\text{Н}/\text{мм}^2$); J – момент инерции поперечных сечений (мм^4).

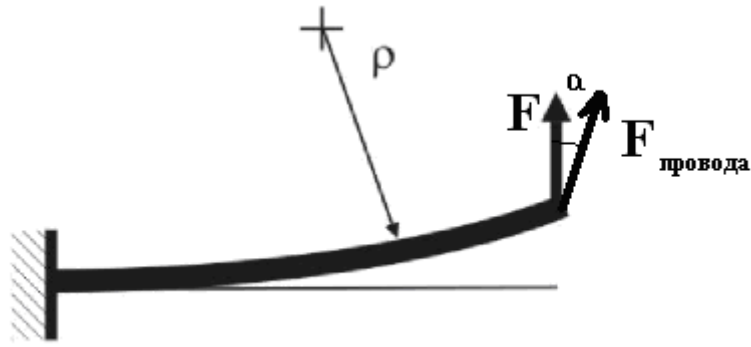


Рис. 5. Радиус кривизны пружины

С другой стороны, из курса высшей математики известно, что уравнение кривизны кривой выглядит следующим образом:

$$\frac{1}{\rho} = \frac{\pm \frac{\partial^2 y}{\partial x^2}}{\left[1 + \left(\frac{\partial y}{\partial x}\right)^2\right]^{\frac{3}{2}}}. \quad (5)$$

Приравняв правые части данных выражений, получим дифференциальное уравнение изогнутой оси балки, которое называется точным уравнением изогнутой оси бруса:

$$\frac{\pm \frac{\partial^2 y}{\partial x^2}}{\left[1 + \left(\frac{\partial y}{\partial x}\right)^2\right]^{\frac{3}{2}}} = \frac{M}{E \cdot J}. \quad (6)$$

В координатной системе прогибов xOy , когда ось y направлена вверх, знак момента определяет знак второй производной от y по x .

Интегрирование данного уравнения, очевидно, представляет некоторые трудности. Поэтому его, как правило, записывают в упрощенной форме, пренебрегая величиной в скобках по сравнению с единицей. Тогда дифференциальное уравнение упругой линии пружины есть

$$\frac{\partial^2 y}{\partial x^2} = \frac{M}{E \cdot J}. \quad (7)$$

Решение дифференциального уравнения

Проинтегрируем уравнение (7) по переменной x :

$$\varphi = \frac{\partial y}{\partial x} = \int \frac{M(x)}{E \cdot J} \cdot dx + C_1, \quad (8)$$

$$y = \iint \frac{M(x)}{E \cdot J} \cdot dx + C_1 \cdot x + D_1. \quad (9)$$

Постоянные интегрирования C_1, D_1 находят из граничных условий – условий закрепления пружины, при этом для каждого участка пружины будут определяться свои постоянные.

Рассчитаем силы, действующие на провод.

1) Определение реакции в заделке (рис. 6):

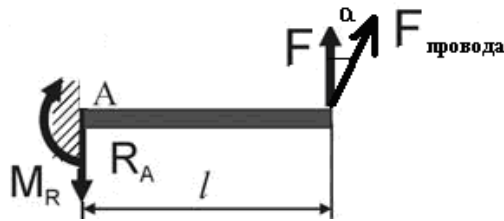


Рис. 6. Реакции в заделке

$$\sum F_y = 0 \rightarrow R_A = \frac{-F}{\cos \alpha}; \quad (10)$$

$$\sum M_A = 0 \rightarrow M_R = \frac{-F \cdot l}{\cos \alpha}; \quad (11)$$

$$F_{\text{пров}} = F \cdot \cos \alpha. \quad (12)$$

2) Определение внутреннего изгибающего момента (рис. 7):

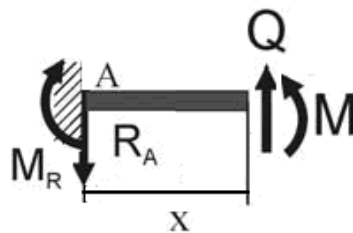


Рис. 7. Внутренний изгибающий момент

$$M(x) = M_R - R_A \cdot x = \frac{F \cdot l}{\cos \alpha} - \frac{F \cdot x}{\cos \alpha}. \quad (13)$$

3) Определение приложенной силы:

$$y(x) = \int \varphi \cdot dx + C_2 = \int \left(\frac{F \cdot l \cdot x}{E \cdot J \cdot \cos \alpha} - \frac{F \cdot x^2}{2 \cdot E \cdot J \cdot \cos \alpha} \right) \cdot dx + D_1; \quad (14)$$

$$y(x) = \frac{F \cdot l \cdot x^2}{2 \cdot E \cdot J \cdot \cos \alpha} - \frac{F \cdot x^3}{6 \cdot E \cdot J \cdot \cos \alpha}. \quad (15)$$

Постоянная D_1 определяется из условий закрепления, а именно: в жесткой заделке прогиб равен нулю, тогда

$$y(x=0) = 0 \rightarrow D_1 = 0. \quad (16)$$

Сила равна

$$F_{\text{пров}}(x=1) = \frac{3 \cdot y \cdot E \cdot J}{l^3 \cdot \cos \alpha}. \quad (17)$$

4) Определение угла поворота сечений пружины:

$$\varphi(x) = \int \frac{M(x)}{E \cdot J} \cdot dx + C_1 = \int \frac{F_{\text{пров}} \cdot l - F \cdot x}{E \cdot J} \cdot dx + C_1; \quad (18)$$

$$\varphi(x) = \frac{F_{\text{пров}} \cdot l \cdot x}{E \cdot J} - \frac{F_{\text{пров}} \cdot x^2}{2 \cdot E \cdot J} + C_1. \quad (19)$$

Постоянная C_1 определяется из условий заделки, а именно: в жесткой заделке угол поворота равен нулю, тогда

$$y(x=0) = 0 \rightarrow C_1 = 0. \quad (20)$$

Угол поворота свободного конца пружины равен (рис. 8)

$$\varphi(x=1) = \frac{F_{\text{провода}} \cdot l}{2 \cdot E \cdot J}. \quad (21)$$

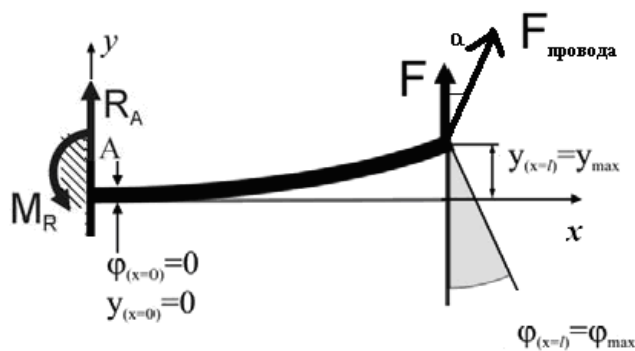


Рис. 8. Угол поворота сечений пружины

Силу F и угол поворота рассчитаем в пакете прикладных программ Maple 10. Сила, приложенная к проводу, равна (рис. 9)

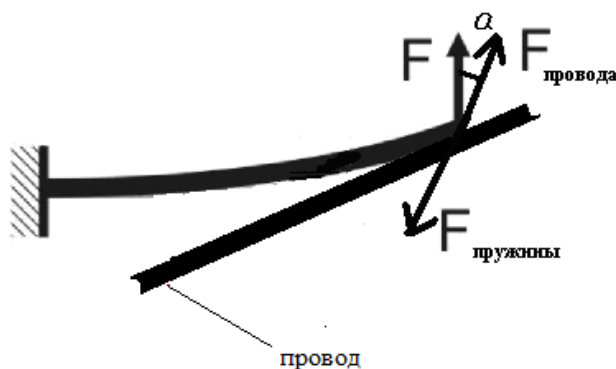


Рис. 9. Распределение сил

$$F_{\text{пруж}} = F_{\text{пров}}(x=1) = \frac{3 \cdot y \cdot E \cdot J}{l^3 \cdot \cos \alpha}, \quad (22)$$

где $y = u$ – компонент перемещений точек срединной поверхности в направлении оси x (мм); E – модуль Юнга, зависящий только от свойств материала (Н/мм^2); J – момент инерции поперечных сечений (мм^4); l – длина пружины (мм), α – угол между F и $F_{\text{провода}}$ ($\alpha = 15^\circ$). В качестве материала для зажимной пружины применяется высококачественная, тщательно проверенная аустенитная хромоникелевая сталь (CrNi), модуль Юнга для стали равен $20 \cdot 10^4 \text{ Н/мм}^2$.

Момент инерции поперечных сечений равен

$$J = \frac{l \cdot h \cdot b^3}{12}, \quad (23)$$

где b – ширина пружины (мм); h – высота пружины (мм); l – длина пружины (мм).

Расчет силы, приложенной к пружине, производится три раза – при разных площадях поперечного сечения проводов, установленных в клемме. Все результаты приведены в табл. 1.

S – площадь поперечного провода (мм^2)	1,5	2,5	4
J – момент инерции поперечных сечений (мм^4)	0,000843	0,000843	0,000843
F – сила (Н)	5,771571	7,451067	9,424937

Таблица 1. Результаты расчета силы $F_{\text{провода}}$

Расчет угла поворота сечений пружины производится по формуле

$$\varphi(x=1) = \frac{F_{\text{пров}} \cdot l^2}{2 \cdot E \cdot J} \quad (24)$$

Результаты приведены на рис. 10.

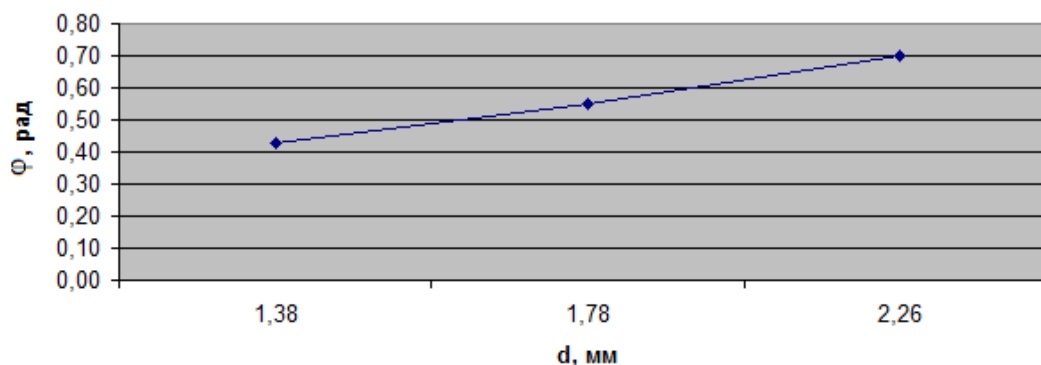


Рис. 10. Зависимость угла поворота сечений от диаметра вставляемого провода

Заключение

Таким образом, в работе произведены расчеты силы, зажимающей провод в клемме, и угла поворота сечений пружины. Результаты расчетов показали, что изменения силы и угла с увеличением диаметра провода происходят линейно.

Литература

1. Алфутов И.А. Основы расчета на устойчивость упругих систем. – М.: Машиностроение, 1978. – 178 с.
2. Тимошенко С.П. Теория упругости. – М.: Наука, 1975. – 256 с.
3. Дьяконов В. Maple 9.5/10 в математике, физике и образовании. – СПб: Солон, 2006. – 28 с.

МЕТОДЫ И СРЕДСТВА ОЦЕНКИ ДИНАМИЧЕСКОЙ ОСТРОТЫ ЗРЕНИЯ

А.П. Саенко

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В работе рассматриваются современные методы и средства оценки динамической остроты зрения, приводится их сравнительный анализ, выявляются преимущества и недостатки каждого.

Введение

Острота зрения является важнейшим интегральным показателем состояния функции зрительного анализатора, и общепринятые методы ее определения основаны на оценке способности зрения различать детали статических визуальных объектов. При этом нормальный глаз с остротой зрения, равной 1, способен различать две точки, видимые под углом 1'. Стандартными средствами определения остроты зрения являются таблицы опто типов. В настоящее время существуют методы исследования остроты зрения с помощью опто типов Ландольта, Снеллена и др. Одной из важных характеристик человеческого зрения является способность восприятия движущихся объектов, которая получила название динамической остроты зрения (ДОЗ) [1]. Для каждого человека существует предельная скорость движения объектов, при превышении которой нарушается восприятие их деталей. Величина ДОЗ определяется предельной угловой скоростью объекта и минимальным временем, при котором испытуемый способен различать его детали.

Существующие методы оценки ДОЗ

За последнее время появилось и активно используется несколько принципиально отличающихся друг от друга методов определения ДОЗ.

Один из них, довольно простой способ, впервые был испытан в 1993 г. [2]. Bhansali и соавт. использовали стандартную таблицу Снеллена (рис. 1). Тестируемые должны были качать головой в горизонтальной плоскости с частотой около 1 Гц и читать вслух буквы на оптометрической таблице. Считалось ненормальным, если наименьшая читаемая линия во время движений головы была более чем на 3 линии больше наименьшей читаемой линии в состоянии покоя. Недостатком данного метода является слишком низкая частота колебаний головы.

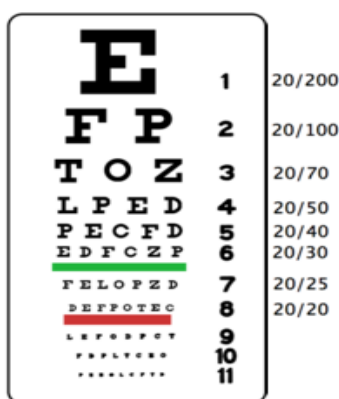


Рис. 1. Таблица Снеллена

С учетом этого в 1997 г. Lee и соавт. предложили измерять ДОЗ при частоте колебаний равной, по крайней мере, 2 Гц. При соответствующей скорости движения головы

на возможность вестибулоокулярного рефлекса удерживать стабилизацию взгляда большое влияние оказывают оптокинетические и системы гладкого приближения.

Однократное или ограниченное использование таблиц Снеллена может быть эффективно при определении состояния зрения, но многократное их задействование может негативно сказаться на результатах экспериментов, так как тестируемый может запомнить некоторые буквы из таблицы.

Следующий метод, который впервые был использован Herdman и соавт. (1998), заключался в измерении ДОЗ с помощью компьютеризированной системы. На лбах тестируемых размещался датчик частоты, после чего они перемещали свою голову в горизонтальной плоскости (со скоростью 120–180°/с) и идентифицировали визуальные объекты. В качестве опто типа использовалась буква «Е», которая отображалась на экране компьютера. Целью тестируемых было определить направление горизонтальных линий буквы «Е». Эксперимент прекращался, когда человек неправильно определял направление.

В 1999 г. Hillman и соавт. использовали ходьбу на беговой дорожке для определения ДОЗ. Аргументом в пользу проведения измерений в подобных условиях был тот факт, что ходьба – это один из видов повседневной деятельности, следовательно, такой метод позволял определить ДОЗ приближенно к жизненным условиям, так как ходьба и бег вызывают перемещения головы в вертикальной плоскости при каждом шаге, учитывался только этот тип колебаний. В этом было принципиальное отличие и преимущество перед другими методиками.

В 2005 г. А.И. Кубарко и соавт. применили новый способ определения ДОЗ [1]. В полностью затемненном помещении после десятиминутной адаптации зрения испытуемых к темноте им предлагалось распознавать направление дефекта кольца Ландольта во время его движения по темному экрану монитора. Для исследования скорости обработки зрительных сигналов и инициирования глазодвигательной реакции у испытуемых оценивался минимальный временной интервал между предъявлением двух визуальных стимулов, достаточный для различения последовательности их презентации на экране. Для этого тестируемый усаживался перед экраном монитора в 17 дюймов так, чтобы его глаза находились на расстоянии 30 см от экрана монитора. Голова испытуемого фиксировалась посредством помещения подбородка на неподвижной подставке.

Оценка свойств зрения с использованием цилиндра

В работе [3] был разработан новый способ оценки свойств зрения: авторы предложили использовать таблицу симметричных опто типов (рис. 2).

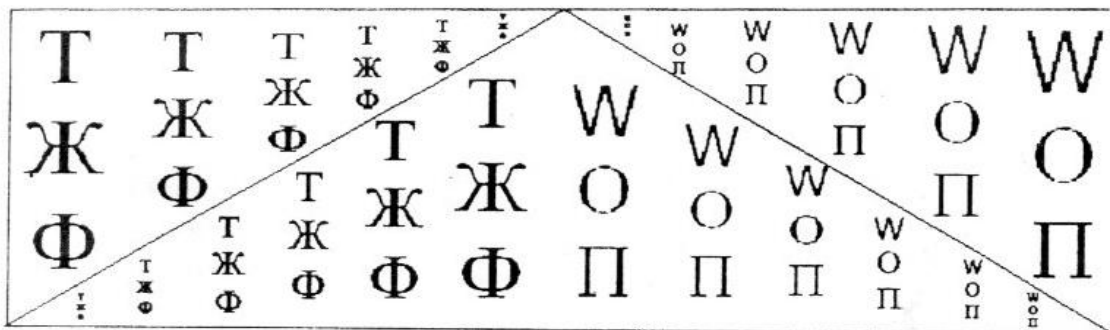


Рис. 2. Таблица симметричных опто типов

При наложении таблицы в качестве текстуры на цилиндр возможны два способа изучения характеристик зрения. Первый заключается в наблюдении боковой поверхно-

сти цилиндра с внешней стороны. Во втором варианте испытуемого помещают в центр цилиндра. При этом в обоих вариантах предполагается вращение цилиндра вокруг своей оси. Такие методы исследования позволяют работать с учетом любых линий горизонта.

Выводы

Зрение играет важную роль в жизни каждого человека, от него не только зависит, насколько комфортно будет чувствовать себя человек в той или иной ситуации, оно, зачастую, определяет способность выполнять ту или иную работу. Так, например, существует ряд профессий, которые предъявляют высокие требования к зрению вообще и к динамическому, в частности. Представителям таких специальностей важно воспринимать и различать движущиеся объекты, поэтому при квалификации водителя, летчика и представителей других профессий необходимо определять ДОЗ.

Однако большинство существующих ныне методов и способов определения ДОЗ обладают рядом существенных недостатков. Некоторые из них требуют довольно дорогого оборудования (например, беговые дорожки), почти все требуют движения головы тестируемого. Эти недостатки во многих случаях исключают применение этих методов в повседневной практике.

Литература

1. Кубарко А.И., Лукашевич И.В. Анализ механизмов динамической остроты зрения // Минск, Медицинский журнал. – 2007. – №1 (19).
2. Erika L. Johnson. Computerized Dynamic Visual Acuity with Volitional Head Movement in Patients with Vestibular Dysfunction // Tampa (Florida, USA), Professional Research Project submitted to the Faculty of the University of South Florida in partial fulfillment of the requirements for the degree of Doctor of Audiology. – 2002.
3. Мусалимов В.М., Воронцов Е.А. Прямые и обратные перспективы на цилиндре как средство оценки зрительной восприимчивости // СПб, «Фундаментальные проблемы оптики-2006».

БЕСКОНТАКТНЫЙ СПОСОБ ИЗМЕРЕНИЯ ГЕОМЕТРИЧЕСКИХ РАЗМЕРОВ ТЕЛ

**Ю.А. Ашмарин, А.А. Виноградова, П.П. Коваленко, Г.А. Недоцука,
М. Urbanski (Технический университет Ильменау, Германия)
Научный руководитель – д.т.н., профессор В.М. Мусалимов**

В работе приводится описание бесконтактного теневого метода измерения размеров изделий с использованием проходящего и отраженного света. Также уделяется внимание оборудованию для реализации данного метода измерений. В указанном оборудовании применяются компьютерные и оптические технологии, и также используются достижения в области технического зрения. При проведении измерений большое внимание уделяется вопросам точности. В работе показывается методика измерений и калибровки камеры с использованием программы для измерительной обработки изображений.

Для измерения размеров объектов могут быть использованы различные методы. Под методом измерений понимают прием или совокупность приемов сравнения измеряемой физической величины с ее единицей в соответствии с реализованным принципом измерений. Обычно метод измерений обусловлен устройством средства измерений. Различают дифференциальный, нулевой, контактный и бесконтактный методы измерений, а также методы сравнения с мерой и метод непосредственной оценки [1].

Контактный метод измерений – метод измерений, основанный на том, что чувствительный элемент прибора приводится в контакт с объектом измерения.

Бесконтактный метод измерений – метод измерений, основанный на том, что чувствительный элемент прибора не приводится в контакт с объектом измерения [2].

Во многих случаях предпочтительнее использовать бесконтактный способ измерений, так как он гарантирует отсутствие повреждений измеряемого объекта, которые могут появиться в результате контакта объекта с чувствительным элементом прибора при контактном методе измерений. В частности, при измерении оптических деталей необходимо пользоваться бесконтактными методами, так как контакт рабочей поверхности оптической детали с чувствительным элементом может привести к ухудшению оптических свойств.

Существуют различные бесконтактные методы измерений. Одним из них является теневой метод измерения. При использовании данного метода измеряемый объект помещается между источником света и приемной видеокамерой. Размеры объекта определяются как разница между границами теневых контуров объекта. Метод является относительно простым в реализации и надежным в эксплуатации [3].

Основными компонентами системы обработки изображений являются камера, объектив и освещение (рис. 1). Также необходим блок для обработки результатов измерения (компьютер и программное обеспечение).

Цифровая камера представляет собой устройство, основным компонентом которой является ПЗС-матрица. Камера производит видеосъемку, преобразование аналогового видеосигнала в цифровой, сжатие цифрового видеосигнала и передачу видеоизображения. Объектив – это линзовая система, предназначенная для проецирования изображения объекта на светочувствительный элемент камеры.

Освещение является важнейшей составной частью системы обработки изображений. От выбора определенного способа освещения зависят точности измерения и позиционирования. Существуют два основных вида освещения: в падающем и проходящем свете, каждый из которых может быть рассеянным и направленным (рис. 2).

Точность данного метода измерений зависит от каждого компонента системы, условий измерения, от самого измеряемого объекта и от его положения в пространстве.

Наличие следующих факторов может снизить точность системы.

1. Наличие в детали слишком темных или слишком светлых участков поверхности, а также резких переходов от темного к светлому.

2. Работа с прозрачными, зеркальными или бликующими поверхностями.
3. Работа с мелкими деталями, острыми углами, маленькими отверстиями.
4. Работа с поверхностью, расположенной под значительным углом по отношению к камере.
5. Нарушение или изначально плохая настройка внутренних параметров системы, нарушение процесса калибровки – основополагающей операции, после которой камера превращается в измерительный инструмент.



Рис. 1. Измерительная установка

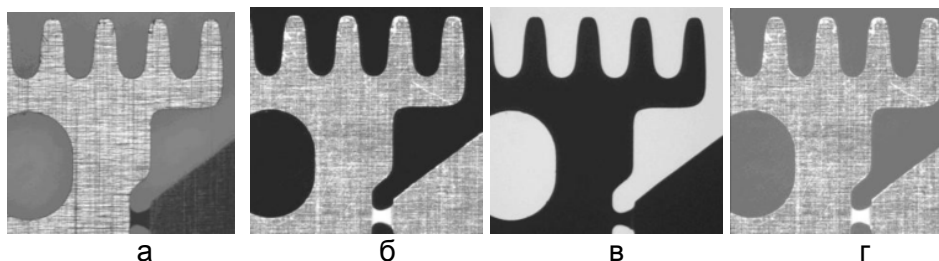


Рис. 2. Примеры использования различного освещения: а) падающий свет, светлое поле; б) падающий свет, темное поле; в) проходящий свет; г) сочетание проходящего и падающего света, светлое поле

При калибровке определяются положения камеры в пространстве, а также оцениваются искажения вносимые объективами в изображения. Неточности калибровки приводят к систематическим ошибкам в измерениях; трудно заметить невооруженным глазом изменение масштаба на 0,1%, однако это вносит существенную систематическую ошибку и может вызвать проблемы при последующей обработке данных.

Кроме систематических ошибок, возможно появление случайных погрешностей. Это в основном связано с появлением шумов и искажениям на изображениях, получаемых с камеры. Поэтому в процессе обработки изображения возможно использование различных фильтров для удаления шумов.

Для измерительной обработки получаемого с камеры изображения объекта используется программа «Osprey». Работа данной программы основана на обнаружении и анализе перепадов яркости изображения. Предварительно стоит указать программе область, которую необходимо проанализировать (рис. 3). Программа, проанализировав заданную область изображения, найдет контур, разделяющий светлое и темное поля. Этот контур будет автоматически измерен, полученные данные будут записаны в банк данных.



Рис. 3. Нахождение границы темного и светлого полей



Рис. 4. Калибровочный шаблон

№ окружности	Радиус, мкм
1	61,256
2	126,332
3	248,806
4	501,371
5	998,894
6	2001,541
7	3998,916
8	8000,690

Таблица 1. Радиусы окружностей калибровочного шаблона



Рис. 5. Калибровочный шаблон в проходящем свете

При проведении измерений особое внимание стоит уделять вопросу точности. Чтобы результаты измерений были верными, камеру предварительно необходимо откалибровать. Процесс калибровки производится с использованием специального калибровочного шаблона, представляющего собой набор концентрических прозрачных и непрозрачных круговых полос (рис. 4). Радиусы границ полос измерены с высокой точностью (табл. 1).

При использовании данного шаблона рекомендуется производить калибровку в проходящем свете (рис. 5). Поместив шаблон в поле зрения объектива камеры, и включив лампу проходящего света, необходимо добиться контрастного изображения шаблона, получаемого с камеры, путем соответствующей регулировки настроек объектива камеры. Изображение просматривается с помощью программы «Osprey».

Получив контрастное изображение шаблона, надо измерить средствами программы радиус одной из окружностей шаблона. Запишем полученное значение:

$$M = 4,1814 \text{ мм} = 4181,4 \text{ мкм.}$$

Из таблицы 1 возьмем значение радиуса соответствующей окружности:

$$S = 8000,69 \text{ мкм.}$$

Далее необходимо обратиться к файлу настроек программы «WOZJ.dat», находящемуся в папке программы «Osprey». В данном файле в соответствующей строке содержится текущее значение масштабного коэффициента, определяющего соотношение между действительным размером измеряемого объекта и размером его изображения (рис. 6). Запишем это значение:

$$P = 43,88500.$$

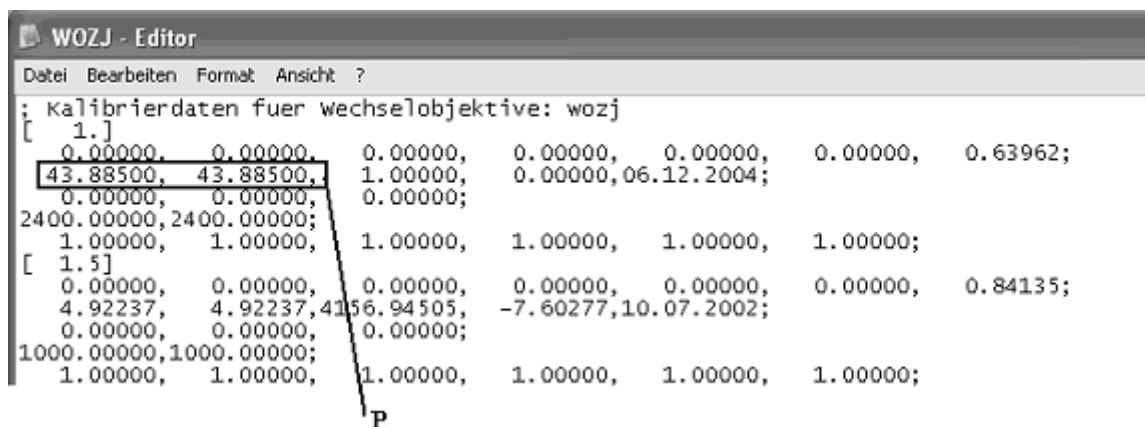


Рис. 6. Значение коэффициента P в файле «WOZJ.dat»

Далее необходимо посчитать уточненное значение этого коэффициента по формуле

$$x = \frac{S}{M} \cdot P,$$

$$x = \frac{8000,69}{4181,4} \cdot 43,885 = 83,96955.$$

Полученное значение необходимо записать вместо старого, полностью соблюдая форматирование данных. Изменения в файле необходимо сохранить, а программу перезапустить, чтобы новые данные были прочитаны из файла и занесены в память компьютера.

Проверим правильность калибровки, вновь измерив ту же окружность:

$$M = 8,0007 \text{ мм} = 8000,7 \text{ мкм};$$

$$S = 8000,69 \text{ мкм}.$$

Сравнив полученное значение M с действительным S из табл. 1, убеждаемся, что калибровка камеры произведена, камера готова к использованию, можно приступить к измерениям. После любого перемещения камеры калибровку необходимо провести вновь.

Таким образом, в данной работе приведено описание бесконтактного теневого метода измерения размеров изделий с использованием проходящего и отраженного света. При проведении измерений большое внимание было уделено вопросам точности. В данной работе была показана методика измерений и калибровки камеры с использованием программы для измерительной обработки изображений.

Литература

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1072 с.
2. Ernst H. Einfuhrung in die digitale Bildverarbeitung; Franzis Verlag, Munchen, 1991.
3. Soille P. Morphologische Bildverarbeitung; Springer Verlag, Berlin u.a., 1998.

РАБОТА С ПАКЕТОМ ПРОГРАММ OPEN COMPUTER VISION

М.С. Ларин

Научный руководитель – д.т.н., профессор В.М. Мусалимов

Статья посвящена описанию возможностей и сфер применений библиотеки компьютерного зрения [1]. Успехи в этой области связаны с появлением все более мощных компьютеров, когда появилась возможность обрабатывать информацию больших размеров. Здесь используются специализированные программы по обработке (2D и 3D) изображений [2–8].

Представление изображения

Редакторы изображений (обработчики графической информации) существуют давно – где в основном было необходимо присутствие человека для оценки изображения и внесения каких-то соответствующих цели действий (корректировок).

Графическая информация (изображение) довольно громоздка или даже избыточна по сравнению, к примеру с координатами объектов или текстом (команды). Но координаты или команды нужно откуда-то брать, в отличие от изображения, которое может существовать и без человека, так как для ввода команд или координат нужен человек. Именно из изображения посредством программной обработки можно получить необходимую информацию – текст или координаты – не прибегая к вмешательству человека. Подобное извлечение информации присуще живым организмам с некоторым интеллектом, достаточным для реакции на изменение окружения. С появлением подобной реакции (хоть и по заданной программе) у компьютерных систем будет появляться эмуляция интеллекта, что значительно упростит взаимодействие человека с системой или системы с окружением. Поэтому избыточная графическая информация, поступающая на обработку, необходима в данном случае.

Распознавание

Изображения обычно представляются в виде массива пикселей, где каждый пиксель состоит из трех (или более) скалярных значений – красный, зеленый, синий, альфа – RGBA. При различном варианте смешивания их можно получить довольно большой диапазон цветов. Но конечная информация, извлеченная из изображения, может довольно сильно отличаться. Например, при распознавании объектов по силуэту можно определить зоны перепадов цвета и записать как 1 (true), а сплошь залитые цветом зоны записать как 0 (false). Так получится массив единиц и нулей, которые можно сравнивать с заранее известными силуэтами и, соответственно, с некоторой погрешностью определить, что за объект на изображении. Но только силуэтов может быть недостаточно для определения объекта, так как с разных сторон объект может иметь разные силуэты. Поэтому в базе силуэтов лучше иметь объемные силуэты или модели объектов, что увеличит распознаваемость, но усложнит алгоритм распознавания. Модели также можно получать из изображения. Абстрактно системе можно, как ребенку, показать картинку с объектом и сказать, каким термином он называется и какие свойства имеет. Для построения модели необходима информация об удаленности каждого фрагмента. На плоских изображениях теряется эта информация, но появляются искажения.

Перенос в виртуальное пространство

В [2] упомянута компенсация отсутствия третьей координаты (глубины), ведь при выводе трехмерного виртуального пространства на монитор ощущение объемности пространства все равно терялось. Был предложен метод, основанный на особенностях человеческого восприятия (зрения) – анаглиф [9]. За основу взято цветное перекрыва-

ние изображений для правого и левого глаза с помощью специальных очков, где одно стекло имеет красный светофильтр, а второе – синий, и изображения для разных глаз полностью окрашены в красный и синий цвета. Изображения должны быть сняты так, как если бы они были видны с позиции правого и левого глаза. Но тут появляется недостаток – теряется обычная цветопередача и довольно сложно сказать, какого цвета объект. В чем-то схожий алгоритм применяется и для реконструкции объекта по изображениям, но, естественно, лишенный недостатка цветопередачи.

Потерянную информацию можно восстановить, имея несколько снимков одного объекта с разных сторон (обычно с двух сторон с небольшой угловой разницей и расстоянием между точками наблюдения); такое восстановление называется стереорекопструкцией. К примеру, Photosynch, разрабатываемый корпорацией Microsoft Live Labs, использует схожую технологию сопоставления фотографий с координатами в пространстве, что дает возможность по фотографиям виртуально осмотреть достопримечательности и получить представление о пространстве вокруг. Зная координаты на обоих изображениях одной и той же точки, можно восстановить третью координату; здесь главная проблема – найти соответствующие точки. Обычно в качестве таких точек берутся ярко выраженные углы, вершины граней, перепады цвета.

Описание библиотеки

Для подобных целей была разработана специальная библиотека, названная OpenCV[1] – сокращение от Open Source Computer Vision Library. Она является библиотекой алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения. Библиотека написана на языке C++ и компилируется под MSVC 6. Библиотека OpenCV состоит, главным образом, из нацеленных на обработку компьютерного зрения в реальном времени алгоритмов: идентификация объектов, сегментация и распознавание, распознавание человеческих лиц, интеллектуальное распознавание, отслеживание движения, получение структур из движения.

Перечислим основные части библиотеки.

- Функции работы с изображением: создание, выделение, удаление изображений, быстрый доступ к пикселям через макросы.
- Структуры данных: статические и динамические типы структур.
- Обработка контуров: нахождение, вывод на экран, манипуляция и упрощение контуров изображений.
- Геометрия: линии и вписывание в эллипс, выпуклые контуры, анализ контуров.
- Особенности: первые и вторые производные изображения; линии, углы нахождения, отслеживание.
- Статистика изображений: в интересующем регионе – количество, минимум, максимум, нормировка, моменты.
- Пирамиды изображений: степени 2, цветовая/текстурная сегментация.
- Разность заднего плана: накапливает изображения. Подсчет средних чисел.
- Дистанционная трансформация.
- Пороговая обработка: бинарный, обратный, нулевой, нулевой обратный.
- Калибровка камеры: внутренняя и внешняя, избавление от искажений, калибровка по шахматной доске.
- Движение: накладывающиеся силуэты, история движения изображения, градиент и весовое глобальное движение.
- Активные контуры: змейки.
- Оптический поток: HS, L-K, B-M и L-K в пирамиде.
- Обработка изображения: фильтр Kalman.

- Построение 3d модели по оценке 2d изображения.
- Гистограммы (распознавание): манипуляции, сравнение, обратная проекция, EMD.
- Интеллектуальное распознавание (основанное на стерео изображениях): определение руки, маска, гистограммы изображений, охватывающий квадрат.
- Матрицы: работа с матрицами, SVD, инвертирование, векторное произведение, перспективная проекция.
- Отображение примитивов: линии, прямоугольники, круги, эллипс, полигон, текст на изображениях.
- Системные функции: загрузка оптимизированного кода, информация о процессоре.
- Утилиты: разность модулей, соответствие, конвертация масштаба, би-линейная интерполяция, тригонометрические функции.

Особенности библиотеки OpenCV

Доступность библиотеки привлекает большее число разработчиков – доступность не только в финансовом плане, но и в плане открытости исходного кода, что позволяет изменять код непосредственно под свои запросы. Открытость исходного кода также способствует его улучшению и оптимизации. Разрабатываемая при поддержке компании Intel библиотека компьютерного зрения имеет выигрыш в производительности, используя специально оптимизированный код для процессоров Intel. Для получения большей производительности можно подключить библиотеку Intel Performance Library – IPP, что повысит быстродействие в пять раз.

Библиотека OpenCV состоит из следующих модулей.

Ядро `score`

- Базовые операции над многомерными числовыми массивами.
- Матричная алгебра, математические функции, генераторы случайных чисел.
- DFT, DCT.
- Запись/восстановление структур данных в/из XML/YAML.
- Базовые функции 2D графики.
- Поддержка более сложных структур данных: разреженные массивы, динамически растущие последовательности, графы.

CV – Модуль обработки изображений и компьютерного зрения

- Базовые операции над изображениями (фильтрация, геометрические преобразования, преобразование цветовых пространств и т.д.).
- Анализ изображений (выбор отличительных признаков, морфология, поиск контуров, гистограммы).
- Структурный анализ, описание форм.
- Анализ движения, слежение за объектами.
- Обнаружение объектов, в частности, лиц.
- Калибровка камер, элементы восстановления пространственной структуры.

Highgui – Модуль для ввода/вывода изображений и видео, создания пользовательского интерфейса

- Захват видео с камер и из видеофайлов, чтение/запись статических изображений.
- Функции для организации простого UI (сейчас все демо приложения используют HighGUI)

Svaux – Экспериментальные и устаревшие функции

- Пространственное зрение: настройка параметров камеры на основе калибров.
- Поиск стерео-соответствия, клики в графах.
- Нахождение и описание черт лица.
- Сравнение форм, построение структур.

- Скрытые марковские цепи.
- Описание текстур.

CvCam – захват видео

- Позволяет осуществлять захват видео (только в Windows), имеет несколько недоделок, поддержка прекращена.

С учетом роста вычислительной производительности видеокарт по сравнению с процессорами была разработана библиотека GpuCV, основанная на коде OpenCV, где для обработки изображений на GPU (видеокартах), т.е. непосредственно в памяти видеокарты, и расчёты производятся уже на процессоре видеокарты. Эта библиотека – неплохая замена библиотеки OpenCV от Intel, так как развитие процессоров для видеокарт происходит стремительнее, чем развитие главных процессоров CPU. GpuCV предлагает полную заменяемость и поддержку команд OpenCV, используя видеокарту.

Тестирование библиотеки

При тестировании систем компьютерного зрения основной задачей являлось пересечение пустыни Мохаве беспилотными машинами, оборудованными разными системами ориентирования. Один из внедорожников был оснащен технологией стереоскопического зрения (библиотека OpenCV) при участии сотрудников научно-исследовательских центров Intel в Сарове и Нижнем Новгороде. Развитие подобных беспилотных автомобилей оказалось довольно прогрессивным. Если в первом Grand Challenge к финишу не приехал ни один из автомобилей, то год спустя к финишу пришли 8, три из которых спонсировались корпорацией Intel. Важно заметить, что эти три внедорожника проехали по всем контрольным точкам, не допустив ни одной ошибки.

Следующую гонку, организованную Управлением перспективного планирования оборонных научно-исследовательских работ США (Defense Advanced Research Projects Agency, DARPA), проводили уже не в пустыне, а в городе, который был воссоздан на заброшенной базе ВВС США George A.F.B. Гонку назвали Urban Challenge Race. В гонке также участвовал автомобиль от Intel, оборудованный шестью камерами GPS. Мозгом автомобиля являлись два компьютера, каждый из которых оснащен четырехядерным процессором Intel® Core™ 2 Quad Q6600 с тактовой частотой 2,4 GHz и платой Intel® D975XBX2 с 2 GB оперативной памяти. Один компьютер обрабатывал информацию, а второй контролировал системы управления. Передача данных от датчиков для обработки на компьютерах осуществлялась с помощью технологии Gigabit Ethernet. Для хранения данных использовались устройства флэш-памяти, а блок питания только аккумулятор автомобиля.

Заключение

В определенной степени система позволяет повысить безопасность не только движения на дорогах города, но и во многих других сферах, требующих интеллектуальной оценки. Здесь за аварийными ситуациями будет следить компьютерная система, предупреждающая от непоправимых ошибок.

Литература

1. Компьютерное зрение / под ред. С.М. Соколова – М.: Бином, 2006.
2. Ларин М.С. Среда виртуального пространства // Сборник докладов 8 сессии международной школы. Фундаментальные и прикладные проблемы надёжности и диагно-

стики машин и механизмов, ФГУП НТЦ “ИНФОРМРЕГИСТР”, РС №11991 от 20 ноября 2007г. (№ гос. рег. 0320702575), СПбГУ ИТМО 2007, С 146–149.

3. <http://www.kv.by/index2001513402.htm>
4. <https://picoforge.int-evry.fr/cgi-bin/twiki/view/Gpucv/Web>
5. <http://labs.live.com/photosynth/default.html>
6. <http://www.intel.com/technology/computing/opencv/index.htm>
7. http://www.atlant.ru/comar/stati_16403.htm
8. http://www.toms-hardware.ru/technews/20071126_150303.html

ФРАКТАЛЬНЫЙ ГЕНЕРАТОР КОХА ДЛЯ КОВРА СЕРПИНСКОГО

А.А. Виноградова, Д.Н. Калитеевский

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В работе приведено описание фракталов. Показан расчет фрактальной размерности для кривой Коха и ковра Серпинского. Применен фрактальный генератор Коха к ковра Серпинского.

Введение

Строгого и полного определения фракталов пока не существует. Дело в том, что определение Мандельброта ограничено. Оно исключает многие фракталы, встречающиеся в физике. Второе определение содержит существенный признак: фрактал выглядит одинаково, в каком бы масштабе его ни наблюдать.

Структуры, похожие на фракталы, можно обнаружить в окружающей нас природе: границы облаков, границы морских побережий, турбулентные потоки в жидкостях, трещины в некоторых породах, зимние узоры на стекле, изображения структуры некоторых веществ, полученные с помощью электронного микроскопа, кровеносная система сердечной мышцы и т.д. Хорошим примером может служить кривая Коха [2]. В 1904 г. математик Кох дал пример кривой, которая нигде не имеет касательной. Это кривая, состоящая из частей, каждая из которых – бесконечной длины. Также в 1915 г. польский математик Вацлав Серпинский представил множество, получаемое путем удаления из равностороннего треугольника подобных ему треугольников (ковер Серпинского) [2]. Важно, что в нем самоподобие сочетается с симметрией поворота.

Понятие «фрактал» уже доказало свою пользу в ряде прикладных областей. Фракталы используются при анализе и классификации сигналов сложной формы, возникающих в разных областях, например при анализе колебаний курса валют в экономике. Они применяются в физике твердого тела, в динамике активных сред и т.д.

В настоящее время фракталы используются для сжатия изображений. Идея фрактального сжатия состоит в нахождении в изображении подобных областей и сохранении в файле только коэффициентов преобразований подобия. Например, в качестве таких областей можно выбрать квадратные области. Набор преобразований подобия – это сдвиг, отражение, поворот и изменение яркости с контрастностью. Сжатие происходит в том случае, если коэффициенты займут места меньше, чем исходное изображение.

Фрактальная размерность

Пусть d – обычная евклидова размерность пространства, в котором находится наш фрактальный объект ($d=1$ – линия, $d=2$ – плоскость, $d=3$ – обычное трехмерное пространство). Теперь покроем этот объект целиком d -мерными шарами радиуса l . Предположим, что нам для этого потребовалось не менее, чем $N(l)$ шаров. Тогда, если при достаточно малых l величина $N(l)$ меняется по степенному закону

$$N(l) \cong \frac{1}{l^D}, \quad (1)$$

то D называется хаусдорфовой или фрактальной размерностью этого объекта. Эта формула эквивалентна соотношению $N \approx (R/l)^D$, использованному для определения береговой линии [3].

Формулу (1) можно переписать также в виде

$$D = \lim_{l \rightarrow 0} \frac{\ln N(l)}{\ln l}. \quad (2)$$

Это служит общим определением фрактальной размерности D . В соответствии с ним величина D является локальной характеристикой данного объекта.

Разберем некоторые классические примеры регулярных фракталов. Пусть на некотором этапе покрытия фрактала пришлось использовать, как минимум, $N(l)$ таких элементов характерного размера l , а на другом $N(l')$ элементов размера l' . Тогда величина фрактальной размерности может быть вычислена по формуле

$$D = -\lim_{l \rightarrow 0} \frac{\ln\left(\frac{N(l)}{N(l')}\right)}{\ln\left(\frac{l}{l'}\right)}. \quad (3)$$

Эту формулу можно переписать в виде:

$$\frac{N(l)}{N(l')} = \left(\frac{l'}{l}\right)^D, \quad (4)$$

что является следствием выражения (1).

Несложно рассчитать фрактальную размерность для кривой Коха. Построение кривой похоже на строение точек множества Кантора. Начинаем с отрезка-основы: удаляем его среднюю третью часть и заменяем ее сторонами равностороннего треугольника (рис. 1).

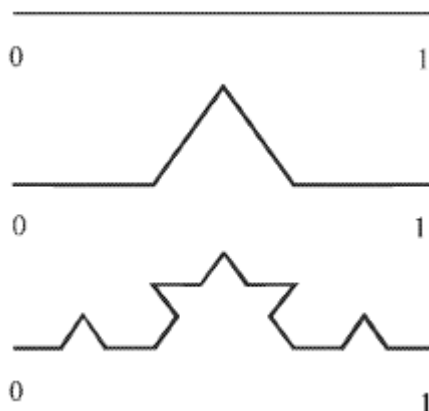


Рис. 1. Последовательные приближения кривой Коха

Мысленно можно представить кривую Коха как предел таких операций. Если основа имеет длину 1, то фрагмент будет состоять из четырех отрезков, каждый длины $1/3$ и, следовательно, общей длины $4/3$. На следующем шаге получаем ломаную, состоящую из 16 отрезков и имеющую общую длину $16/9$ или $(4/3)^2$ и т.д. Так как на каждом шаге

$$\frac{\ln 4}{\ln 3} = \frac{\ln 4^2}{\ln 3^2} = \dots, \quad (5)$$

то можно применить формулу (2). Так как предел $l \rightarrow 0$ соответствует $n \rightarrow \infty$, то фрактальная размерность равна

$$D = \frac{\ln 4}{\ln 3} = 1.26\dots \quad (6)$$

Ковер Серпинского – это регулярный фрактал, который получается последовательным вырезанием равносторонних треугольников (рис. 2). Здесь на нулевом шаге имеется один равносторонний треугольник с длиной стороны $l = 1$, а на следующем – три равносторонних треугольника со сторонами $l' = 1/2$. Поэтому $l = 1$, $N(l) = 1$, а $l' = 1/2$, $N(l') = 3$. Тогда фрактальная размерность

$$D = \frac{\ln 3}{\ln 2} = 1,5849. \quad (7)$$

Ковер имеет нулевую площадь, поскольку в процессе его построения была исключена площадь, в точности равная площади исходного треугольника. Об этом же говорит и значение фрактальной размерности $D < 2$, которая меньше размерности плоскости, на которой находится этот объект.



Рис. 2. Построение салфетки Серпинского

Применение фрактального генератора Коха для ковра Серпинского

При помощи программирования в М-файлах системы MATLAB была выведена кривая Коха (рис. 3).

Построение кривой Коха начинается с прямолинейного отрезка единичной длины. Исходный отрезок называется затравкой и может быть заменен каким-нибудь многоугольником. Затравка – это нулевое поколение кривой Коха. Затем каждое звено затравки мы заменяем образующим элементом, $n=1$. В результате этого получаем первое поколение – кривую из четырех прямолинейных звеньев. Следующее поколение получается при замене каждого прямолинейного звена уменьшенным образующим элементом. В результате получаем кривую второго поколения, $n=2$. В MATLAB мы вывели кривую пятого поколения (рис. 3).

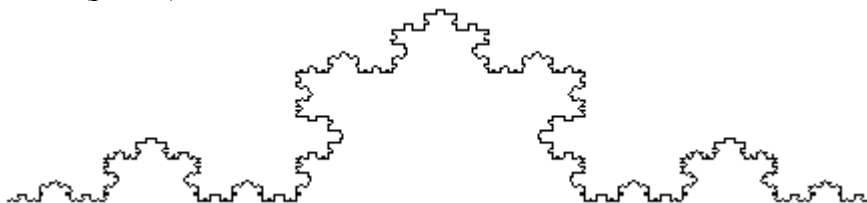


Рис. 3. Кривая Коха

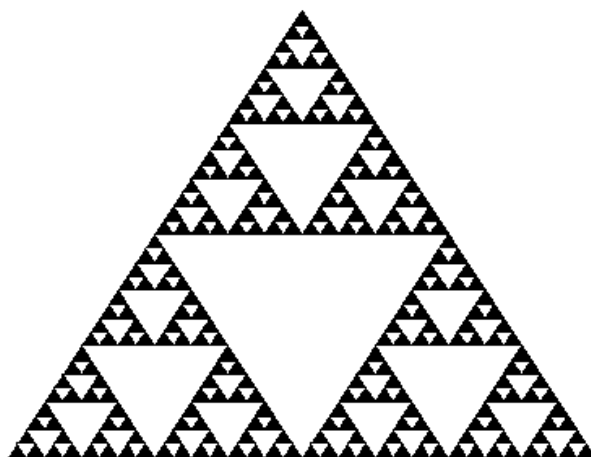


Рис. 4. Салфетка Серпинского

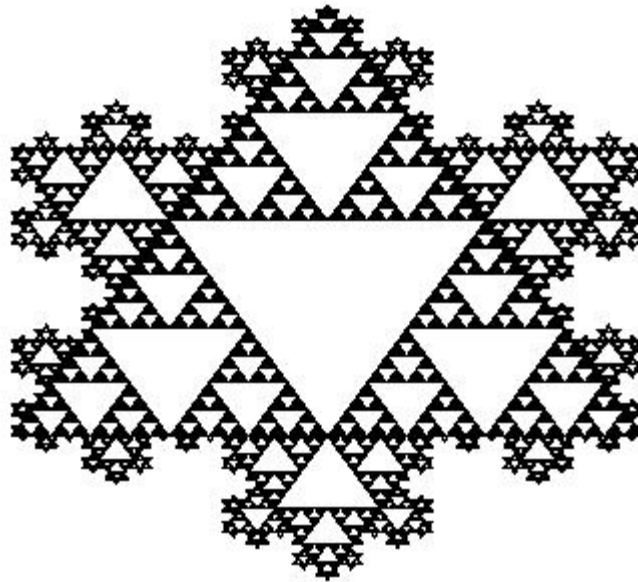


Рис. 5. Применение кривой Коха для ковра Серпинского

Также в системе MATLAB был выведен ковер Серпинского. Построение начинается с равностороннего треугольника. Из него удаляется перевернутый равносторонний треугольник со стороной, равной половине длины стороны исходного треугольника. У нас остаются три равносторонних треугольника со сторонами, вдвое меньшими стороны исходного треугольника. Повторив эту операцию 5 раз, мы получаем ковер Серпинского, показанный на рис. 4. Затем, совмещая две программы построения и совмещая координаты построения кривой Коха и ковра Серпинского, получаем изображение, приведенное на рис. 5. Ниже приведена программа для построения этого изображения в системе MATLAB.

```
function z=Koch(N);
    x1=0; y1=0; x2=0.5; y2=cos(pi/6);
    figure (1);
    axis ([-0.19 1.19 -0.29 1.09]);
    hold on;
    set(gca, 'xtick', [], 'ytick', []);
    set(gca, 'XColor', 'w', 'YColor', 'w');
    Coord (x1,y1,x2,y2,4);
    Coord (x2+x2,y1,x1,y1,4);
    Coord (x2,y2,x2+x2,y1,4);
    Serpinsky(x1,y1,x2,y2,x2+x2,y1,5);
    function z=Coord (x1,y1,x2,y2,n)
        if n>0
            dx=(x2-x1)/3; dy=(y2-y1)/3;
            x1n=x1+dx; y1n=y1+dy;
            x2n=x1+2*dx; y2n=y1+2*dy;
            xmid=dx/2-dy*sin(pi/3)+x1n;
            ymid=dy/2+dx*sin(pi/3)+y1n;
            Serpinsky(x1n,y1n,x2n,y2n,xmid,ymid,n);
            Coord(x1,y1,x1n,y1n,n-1);
            Coord(x1n,y1n,xmid,ymid,n-1);
            Coord(xmid,ymid,x2n,y2n,n-1);
            Coord(x2n,y2n,x2,y2,n-1);
        end
    end
end
```

```

        else
            hold on;
            r1=[x1 y1]; r2=[x2 y2]; R=cat(1,r1,r2);
            plot (R(:,1), R(:,2), 'Color','k');
        end;
        z=1;
    end;
end;
function z=Serpinsky(x1,y1,x2,y2,x3,y3,Lmax);
    h= figure (1);
    hold on;
    fill ([x1 x2 x3],[y1 y2 y3],'k');
    Simplex (x1,y1,x2,y2,x3,y3,0,Lmax);
    hold off;
    function z=Simplex(x1,y1,x2,y2,x3,y3,m,Lmax);
        if m<Lmax;
            x1m=(x1+x2)/2;
            y1m=(y1+y2)/2;
            x2m=(x2+x3)/2;
            y2m=(y2+y3)/2;
            x3m=(x3+x1)/2;
            y3m=(y3+y1)/2;
            fill([x1m x2m x3m],[y1m y2m y3m],'w');
            m=m+1;
            Simplex(x1,y1,x1m,y1m,x3m,y3m,m,Lmax);
            Simplex(x1m,y1m,x2,y2,x2m,y2m,m,Lmax);
            Simplex(x3m,y3m,x2m,y2m,x3,y3,m,Lmax);
        end;
    end;
end;
end;

```

Заключение

В работе были рассмотрены примеры фракталов (Коха и Серпинского). В ходе данной работы были приведены расчеты фрактальной размерности, в том числе для кривой Коха ($D = 1,26$) и для ковра Серпинского ($D = 1,5849$).

Получен совмещенный фрактал кривой Коха и ковра Серпинского. Что в дальнейшем может способствовать изучению многих физических процессов, выбирающих естественным образом те подмножества структур, на которых они происходят. Также они могут применяться в процессе изучения технического зрения.

Литература

1. Мандельброт Б.Б. Фрактальная геометрия природы. – М.: –Ижевск, Ин-т комп. ис- след., 2002. – 666 с.
2. Морозов А.Д. Введение в теорию фракталов – М.: –Ижевск, 2002. – 163 с.
3. Федер Е. Фракталы. – М.: Мир, 1991. – 260с.
4. Шредер М. Фракталы, хаос, степенные законы. – Ижевск: НИЦ «РХД», Ижевская республиканская типография, 2001. – 528 с.

МОДЕРНИЗАЦИЯ ТРИБОМЕТРИЧЕСКОЙ СИСТЕМЫ «ТРИБАЛ»

Д.А. Бузунов, Р.А. Бузунов, А.А. Сизова

Научный руководитель – д.т.н., профессор В.М. Мусалимов

В статье предложен вариант модернизации установки для исследования трибологических взаимодействий «Трибал». Подобраны новые узлы и элементы, позволяющие улучшить качественные и массогабаритные характеристики системы.

Введение

Установка «Трибал» относится к исследованию трибологических свойств материалов [1, 2]. На рис. 1 представлена установка, которую планируется модернизировать. Устройство состоит из основания и установленной на основании платформы на катках с расположенными на ней держателями, в которых размещаются образец и контрообразец. Также устройство содержит привод возвратно-поступательного перемещения, узел нагружения образцов, датчики перемещения, соединенные с держателями, компьютер, с которым связаны датчики.

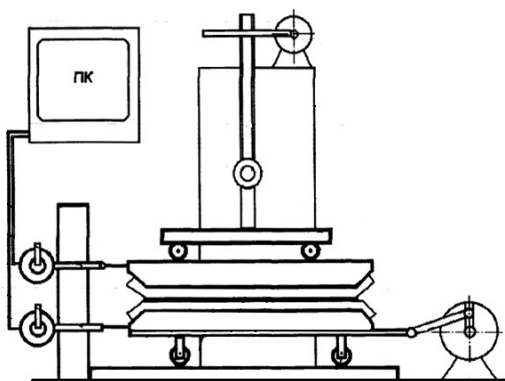


Рис. 1. Схема установки

Узел нагружения представляет собой винтовой домкрат, соединенный через пружину с верхней платформой, на которой закрепляется контрообразец, что обеспечивает плавную передачу усилия от домкрата к образцу. Датчик перемещения выполнен в виде комбинации индикаторной часовой головки и оптоэлектронного преобразователя, установленного на оси для указательной стрелки соответствующего индикатора.

Данная система хорошо зарекомендовала себя. Она проста и надежна, позволяет получать достаточно точные результаты. Была поставлена задача модернизации установки: оснащение ее современными узлами, агрегатами, датчиками, позволяющими улучшить ее ценные качества. Задачи сводятся к:

- выбору современного привода перемещения платформы с образцом;
- модернизации направляющих линейного перемещения образцов;
- выбору датчиков для регистраций усилий и перемещений, отвечающих всем необходимым точностным требованиям;
- модернизации узла нагружения.

Также было необходимо продумать способ крепления образцов, учесть при этом возможность взаимодействия образцов в условиях непрерывной смазки, что позволит увеличить область задач, решаемых данной установкой, и выведет ее на новый уровень исследования трибологических свойств материалов.

Разработанная модель

На рис. 2 представлен разработанный вариант конструкции установки.

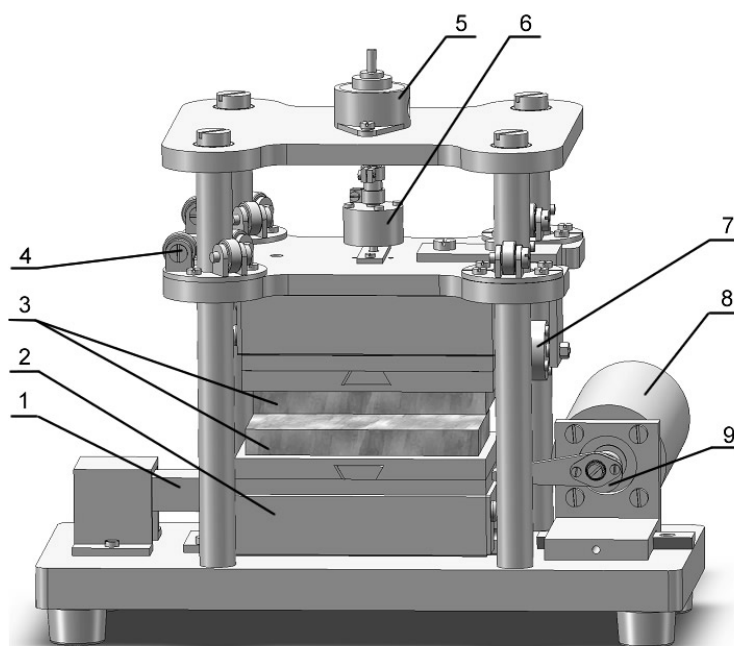


Рис. 2. Модель установки: 1 – растровый датчик перемещения, 2 – стол, 3 – образцы, 4 – ролики вертикальных направляющих, 5 – двигатель линейно перемещения, 6 – датчик силы нагружения, 7 – датчик силы взаимодействия, 8 – мотор-редуктор, 9 – кривошип с регулируемой длиной

Рассматривались два варианта привода возвратно-поступательного перемещения:

- привод линейного перемещения,
- более экономичный вариант – мотор-редуктор.

Привод линейного перемещения имеет множество положительных качеств. Он отвечает современным тенденциям развития техники, удобен в применении, имеет широкие возможности по регулированию скорости, перемещения, моделированию законов движения, однако и имеет серьезный недостаток – высокую цену.

Мотор-редукторы широко распространены, просты в конструкции, имеют относительно низкую стоимость, однако обладают узкими диапазонами регулирования скорости и момента, также при их использовании неизбежны сложности с моделированием различных законов движения выходного вала, преобразуемого в линейное перемещение платформы.

На данном этапе ввиду экономических соображений было принято решение выбрать в качестве привода линейного перемещения именно мотор-редуктор в комбинации с кривошипно-ползунным механизмом с регулируемой длиной кривошипа, что позволяет изменять амплитуду движения платформы. Для реализации различных законов движения на основе данных элементов планируется использование различных кулачков с геометрическим замыканием. Преобразование от кривошипно-ползунного механизма к кулачковому планируется осуществлять следующим образом: с выходного вала двигателя снимается кривошип, затем отсоединяется шатун; далее на вал двигателя крепится кулачок и соединяется с предварительно закрепленным на платформе толкателем, двигатель же, установленный на свои направляющие, способен перемещаться для компенсации разницы в длинах звеньев.

Из широкого выбора мотор-редукторов, представленными различными компаниями, выбор был остановлен на модели IG-42GM, поставляемой фирмой «Электропривод». Он состоит из реверсируемого коллекторного двигателя постоянного тока и планетарного редуктора. При варианте с редуктором, выдающим передаточное отношение 17, получаем максимальную скорость 346 об/мин при крутящем моменте 7,6 кг·см, что по предварительным расчетам отвечает необходимым требованиям. К мотор-редукторам также может быть подключен блок управления.

Блоки управления позволяют осуществлять:

- управление скоростью двигателя;
- управление направлением движения;
- ограничение момента на валу двигателя.

В качестве альтернативы направляющих линейного перемещения предлагается применение двух столов японской фирмы ТНК, в которых движение основывается на применении тел качения. Шарики разделены сепаратором и защищены от попадания возможного продукта износа образцов. На данный стол, с помощью штатных резьбовых отверстий, планируется установка вспомогательной платы с направляющей типа «ласточкин хвост», на которую, в свою очередь, будет фиксироваться специальная оснастка с закрепленным образцом. Применение направляющей типа «ласточкин хвост» позволяет осуществлять быструю и удобную замену образцов на оснастке.

Нагрузочный узел предлагается модернизировать следующим образом. Вертикальная нагрузка на образцы производится через выходной вал линейного шагового двигателя. Двигатель крепится на плате, базирующейся на четырех стойках, являющихся направляющими для вертикального перемещения контробразца. Угол шага двигателя составляет $7,5^\circ$, который преобразуется в 0,0417 мм линейного перемещения с максимальным ходом 40 мм.

Передача усилия от двигателя к платформе производится через датчик силы с двусторонним вводом нагрузки, работающим как на растяжение, так и на сжатие. На выходном штоке датчика закреплен специально подготовленный шарик от шарикоподшипника. Шарик участвует в формировании высшей кинематической пары и опирается в специальный подпятник, закрепленный на платформе, осуществляющей вертикальной перемещение.

Для снятия закона движения нижнего образца предлагается использование расстрогового датчика фирмы «Рифтэк» (Республика Беларусь), а для установления силы взаимодействия образцов горизонтальное движение верхней платформы предлагается ограничить датчиком силы, аналогичным применяемому в узле нагружения.

Заключение

Были подобраны новые элементы конструкции, а именно: выбраны направляющие горизонтального перемещения и привод, осуществляющий поступательное движение платформы через преобразование вращательного движения, подобраны элементы узла нагружения, выбраны датчики снятия сигналов.

Литература

1. Пат. 2244290, МПК⁷ G 01 N 19/02. Устройство для испытания материалов на трение.
2. Мусалимов В.М., Валетов В.А. Динамика фрикционного взаимодействия / Монография. – СПб: СПбГУ ИТМО, 2006. – 191 с.

ВЛИЯНИЕ МИКРОГЕОМЕТРИИ ПОВЕРХНОСТИ НА ЕЕ ИЗНОСОСТОЙКОСТЬ – ПЛАНИРОВАНИЕ ЭКСПЕРИМЕНТА

А.В. Терещенко

Научный руководитель – д.т.н., профессор В.А. Валетов

В статье освещен вопрос планирования эксперимента для определения влияния микрогеометрии на износостойкость поверхности, нахождения оптимальной микрогеометрии для данного свойства с учетом имеющегося оборудования на базе использования непараметрических критериев. Также заложена основа для последующего построения теоретической модели получения заданной износостойкости с учетом шероховатости.

Введение

Оптимизация микрогеометрии поверхностей – это один из немногих экономически выгодных в настоящее время способов повышения качества изделий. Традиционно применяемые методы, как, например, использование новых материалов или повышение точности изготовления, либо неспособны удовлетворить объем запросов производства, либо практически подошли к порогу экономической целесообразности своего применения. Поэтому, если учесть тот факт, что микрогеометрия играет важнейшую роль в таких свойствах поверхностей, как адгезия, трение и износостойкость при скольжении и качении, контактная жесткость, отражательная способность, износостойкость при переменных нагрузках, коррозионная стойкость, качество лакокрасочных покрытий и т.д., всего не менее двадцати функциональных свойств; то создание оптимального микро рельефа рабочих поверхностей деталей, наряду с применением термических и физико-механических способов улучшения их поверхностного слоя, является одним из основных и наиболее эффективных резервов значительного повышения качественных показателей машин и приборов.

В настоящее время возможности управления микрогеометрией для повышения качества производимых изделий используются мало и неэффективно, не в последнюю очередь из-за малоинформативных стандартных критериев, неспособных адекватно точно и полно отражать изменения реального микрорельефа в процессе обработки либо под воздействием эксплуатационных нагрузок. В статье рассматривается вопрос планирования эксперимента для определения влияния микрогеометрии на износостойкость поверхности с использованием современного оборудования, полученного кафедрой технологии приборостроения СПбГУ ИТМО в рамках государственного гранта, и нахождения оптимальной микрогеометрии для данного свойства с учетом имеющегося оборудования на базе использования непараметрических критериев, составляющих основу методики автоматизированной оценки микрогеометрии поверхностей, реализованной в программном продукте «Профиль». Также заложено основание для последующего построения теоретической модели получения заданной износостойкости с учетом шероховатости.

Износостойкость, ее важность, влияние микрогеометрии

Износостойкость (износоустойчивость) – сопротивление материалов деталей машин и других трущихся изделий износу. Износостойкость оценивается, например, по убыли массы детали за время ее работы [1]. Износостойкость является эксплуатационным свойством, определяющим способность поверхностных слоев деталей сопротив-

ляться разрушению при трении скольжения, трении качения, а также при микроперемещениях, обусловленных воздействием вибраций. В результате износа деталей машин понижается КПД, теряется точность, снижается прочность, увеличиваются динамические нагрузки, которые являются следствием увеличения зазоров в сопряжениях, повышается шум при работе. Поэтому несомненно, что это свойство имеет огромную экономическую важность для всех без исключения изделий и производств.

В настоящее время износостойкость преимущественно повышают за счет различных видов упрочнения поверхностного слоя – физических, химических, комбинированных. При этом возможности улучшения износостойкости за счет изменения микрогеометрии поверхности используются слабо. Это связано с тем, что износостойкость зависит от всего диапазона микронеровностей – отклонений формы, волнистости, шероховатости, их направлений и взаимосочетаний, при этом существующие стандарты не дают достаточно точных инструментов, позволяющих полно и точно описать все их многообразие. Давно доказано, что нынешние широко распространенные параметры не позволяют различать даже совершенно различные в эксплуатационном плане микрогеометрии, присваивая им одинаковые значения параметров [2, 3]. Очевидно, что, не имея точного и полного описания, невозможно различать различные микрогеометрии, а при невозможности их различить – невозможно и точно устанавливать связь между величиной износостойкости и значениями показателей микрогеометрии. Это, в свою очередь, и обуславливает непопулярность ориентированных на микрогеометрию методов управления величиной износостойкости.

При взаимном перемещении контактирующих плоских или цилиндрических поверхностей, имеющих микронеровности (шероховатость), в первоначальный момент происходит срез, отламывание и пластический сдвиг вершин неровностей, так как их контакт происходит по вершинам неровностей. Сначала сравнительно быстро происходит начальное изнашивание (приработка). При правильном режиме смазывания изнашивание протекает медленно, что обусловлено образованием равновесной шероховатости. Этот период времени определяет срок службы детали. Наконец, на последнем этапе жизни детали происходит так называемое катастрофическое изнашивание пары. Такова классическая теория изнашивания, но исследование этого процесса с применением намного более точных показателей – непараметрических критериев – показало, что не все так однозначно: там, где традиционные параметры показывают установившуюся «равновесную шероховатость», на самом деле происходит циклический процесс смены различных микрогеометрий [3].

Именно поэтому целью данной работы является разработка плана исследования влияния микрогеометрии на износостойкость поверхностей деталей на качественно новом уровне – с использованием современного оборудования (профилометра, машины трения) и, главное, программного обеспечения – системы «Профиль» – реализующего анализ микрогеометрии при помощи непараметрических критериев, таких как плотности и функции распределения ординат и тангенсов углов наклона профиля, позволяющих заметить и проанализировать мельчайшие отклонения различных микрогеометрий друг от друга в автоматизированном режиме [4]. Кроме этого, будет показана схема проведения оптимизации микрогеометрии для данного функционального свойства с учетом имеющегося оборудования – процесса, который позволяет получить максимально возможное значение свойства на данном предприятии (наборе оборудования и технологических приемов).

План эксперимента по определению влияния микрогеометрии на износостойкость при трении скольжения и схема ее оптимизации с учетом имеющегося оборудования

Для исследования влияния микрогеометрии на износостойкость необходимо зафиксировать все параметры заготовок, кроме шероховатости (материал, состояние поверхности, вид и параметры обработки), изменения которой можно достичь, например, меняя один из параметров режима резания.

С учетом экономических соображений и наличия оборудования выбран способ обработки – фрезерование без СОЖ, скорость и подача фиксированы, меняется глубина резания. Для достоверности, с учетом требований теории планирования эксперимента и данных, полученных в работах [3, 5], целесообразно создать четыре комплекта образцов с разной шероховатостью по три образца в каждом и снимать по три профиля на образец.

Требования к заготовкам и режимам резания назначены, исходя из удобства их обработки на фрезерном станке, а также из экономических соображений.

Заготовки: конструкционная сталь 45, прямоугольный параллелепипед $60 \times 50 \times 10$ мм. Обработка – на вертикально-фрезерном станке цилиндрической фрезой 50 мм из быстрорежущей стали с 12 косыми зубьями. Режимы резания: скорость резания $v = 19$ м/мин, подача $S = 36$ мм/мин = $0,024$ мм/зуб, глубины резания $t_1 = 0,4$ мм, $t_2 = 0,6$ мм, $t_3 = 0,8$ мм, $t_4 = 1$ мм для каждого из наборов, соответственно. Помимо фрезерного станка, в состав требуемого оборудования входят профилометр с возможностью обмена данными с ПК, машина трения, аналитические или лабораторные весы.

Согласно [3], для получения профилей выбираем базовую длину 10 мм – для учета волнистости, также влияющей на износостойкость. Измерения будем проводить в поперечном направлении, так как критерий стационарности выполняется только для него, а продольное направление непригодно для оценки микрогеометрии фрезерованной поверхности.

После получения набора образцов пошаговый план выполнения эксперимента выглядит следующим образом:

- 1) получение трех профилей поверхности до испытания в поперечном направлении, расчет непараметрических критериев и стандартных критериев Ra, Rq, Rm по ГОСТ 2789-73 в системе «Профиль», измерение массы образца;
- 2) испытание образца на машине трения без смазки по стандартному циклу согласно документации;
- 3) получение трех профилей поверхности после испытания в поперечном направлении, расчет непараметрических критериев и стандартных критериев Ra, Rq, Rm по ГОСТ 2789-73 в системе «Профиль», измерение массы образца – масса, которая была потеряна и станет критерием износостойкости.

Целесообразно ввести систему обозначения образцов для фиксации данных эксперимента в виде: №комплекта–№образца в комплекте–{Д или П}, где последняя буква обозначает, соответственно, образец до или после испытания. Например, обозначение вида 1-2-П будет обозначать первый комплект образцов, второй образец в комплекте, после испытания. По результатам эксперимента полученные данные будут занесены в таблицы.

Рассчитанные непараметрические критерии будут храниться в электронном виде в формате системы «Профиль».

На основании полученных данных можно будет установить взаимосвязь микрогеометрии поверхностей и измеренных уровней износостойкости поверхностей, сделать выводы о том, какие формы микрорельефов дают наиболее высокую износостойкость и что, в технологическом плане мы сможем сделать для получения такой микрогеометрии в будущем. При этом; если отбросить некоторые данные, которые мы получаем в результате этих опытов, становится очевидным, что перед нами – работающий метод оптимизации микрогеометрии для требуемого свойства (износостойкости) на конкретном парке оборудования и технологических приемов. Несмотря на искусствен-

ность выбора параметров резания, вызванную необходимостью контролировать условия эксперимента, в условиях реального производства схема будет абсолютно такой же: формирование набора образцов, обработанных на оборудовании и с применением традиционных технологических приемов, характерных для данного производства; получение непараметрических критериев каждого из образцов, последующие испытания на нужное нам свойство, выбор наилучшего образца, а по нему и технологических условий его получения и характерной для него микрогеометрии, описанной непараметрически. После этого остается только задать допустимые пределы отклонения от этой, эталонной, микрогеометрии и производить уже серийные изделия, контролируя микрогеометрию при помощи полученного таким образом шаблона, что при современном уровне развития вычислительной и измерительной техники может выполняться в полностью автоматическом режиме.

Образец/критерий	Ra, мкм	Rq, мкм	Rm, мкм	Масса, г

Таблица 1. Журнал испытаний

Критерий (среднее значение)		Ra, мкм	Rq, мкм	Rm, мкм	Масса, г
1-й комплект	до испытаний				
	после испытаний				
	изменение в %				
2-й комплект	до испытаний				
	после испытаний				
	изменение в %				
3-й комплект	до испытаний				
	после испытаний				
	изменение в %				
4-й комплект	до испытаний				
	после испытаний				
	изменение в %				

Таблица 2. Расчетные значения

Модель износостойкости, учитывающая микрогеометрию

На основании испытаний, описанных в предыдущем разделе, возможно будет установить взаимосвязь износостойкости поверхности и ее микрогеометрии на качественно новом уровне за счет применения более точного и полного ее описания.

Эта взаимосвязь ляжет в основу построения модели зависимости износостойкости от микрорельефа поверхности, конечной целью создания которой является получение инструмента, позволяющего на входе задавать требуемый уровень свойства, а на выходе – технологические условия его получения вплоть до режимов обработки. Такая модель позволит реализовать автоматизированный режим подбора технологических приемов на этапе ТПП, гарантируя при этом получение максимально возможного уровня свойства на выходе. Реализована она может быть как в виде математических зависимостей непараметрических критериев профиля с уровнем износостойкости (потери массы при испытаниях), так и в виде экспертной системы с набором эвристик, реали-

зующих выявленные закономерности, в том случае, если их формализация на языке математики будет невозможна или нецелесообразна.

Все это приведет к снижению затрат на производство с одновременным повышением качества выпускаемой продукции за счет сокращения времени принятия решения и более обоснованного выбора качества обработки поверхности. Уже было показано, что микрогеометрия, выбираемая на основе традиционных подходов, может быть неоптимальной и одновременно более дорогой в изготовлении, чем оптимальная для данного свойства [6].

Заключение

Мы рассмотрели экономические предпосылки целесообразности исследования взаимосвязи микрогеометрии поверхности и ее износостойкости на основе принципиально нового подхода с применением непараметрических критериев оценки микрогеометрии.

Предложен план эксперимента по установлению взаимосвязи, приведены фактические сведения, необходимые для его реализации на технической базе кафедры технологии приборостроения СПбГУ ИТМО, а также любой другой организации, предприятиях, обладающих соответствующим оборудованием.

Показана практическая схема проведения оптимизации микрогеометрии для требуемого функционального свойства в условиях конкретного парка оборудования и технологических приемов, изложены предпосылки создания обобщенной модели взаимосвязи микрогеометрии и износостойкости, показаны возможные пути реализации этой задачи.

Литература

1. Большой энциклопедический словарь. – Режим доступа: <http://dic.academic.ru/>
2. Валетов В.А. Оценка шероховатости, волнистости, отклонений формы поверхностей с помощью ЭВМ / В кн.: Технология судостроения, судового машиностроения, обработка металлов и сварка. – Лен. кораблестроительный ин-т, 1980. – С. 133–135.
3. Валетов В.А. Оптимизация микрогеометрии поверхностей деталей в приборостроении. Учебное пособие. – Л.: ЛИТМО, 1989. – С. 100.
4. Терещенко А.В. Программный продукт для автоматизированного анализа и контроля микрогеометрии поверхностей с использованием непараметрического подхода // Инструмент и технологии. – 2004. – №19–20. – С. 109–112.
5. Мусалимов В.М., Валетов В.А. Динамика фрикционного взаимодействия: Монография: В 2-х частях. – СПб: Издательство ПИМаш (ЛМЗ-ВТУЗ), 2006. – 168 с.
6. Валетов В.А. Влияние исходной микрогеометрии на коэффициент сопротивления качению и долговечность роликовых направляющих // Трение и износ. – 1982. – №5. – С. 914–918.

ИНТЕГРАЦИЯ КОНСТРУКТОРСКИХ И ТЕХНОЛОГИЧЕСКИХ САПР НА ОСНОВЕ ФОРМАТА 3DXML

Н.А. Шувал-Сергеев

Научный руководитель – д.т.н., профессор Д.Д. Куликов

Проблема несовместимости форматов и связанные с ней сложности передачи данных о 3D модели в настоящее время является актуальной. Применение формата 3DXML способно решить вопрос о способе передачи информации о проектируемом объекте между конструкторско-технологическими системами.

Введение

В последнее время на отечественных производственных предприятиях широкое распространение получили российские САД/САМ-системы, которые призваны уменьшить трудоемкость и сократить время на разработку конструкторской и технологической документации. Однако существующие возможности отечественных систем не способны обеспечить автоматизацию всех этапов технологической подготовки.

Современные конструкторские разработки основываются на 3D моделировании. Полученная в результате проектирования модель является базой для дальнейшей подготовки технологической документации. Проблемой, с которой сталкиваются на производстве, является невозможность использования данных о 3D модели без специального программного обеспечения. Большинство существующих форматов хранения и передачи данных между САД/САМ/САЕ-системами не всегда способны удовлетворять всем требованиям разработчика [1]. Основные сложности при передаче графической информации между системами связаны с:

- (а) отсутствием или ограниченными возможностями трансляторов, обеспечивающих обмен данными;
- (б) утратой информации при ее трансформации из одного математического представления в другое;
- (в) разной интерпретацией правил чтения и записи форматов файлов;
- (г) различными версиями форматов файлов и трансляторов.

Приведенные проблемы приводят к увеличению количества ошибок при проектировании, продлению сроков подготовки производства и т.д., что влечет за собой заметные экономические потери и существенно влияет на себестоимость продукции.

Основной целью данной работы является исследование возможности использования 3DXML формата в качестве основы получения данных о 3D модели объекта для дальнейшего использования их в конструкторско-технологических системах [2].

Почему 3DXML?

3DXML – универсальный формат, разработанный компанией Dassault Systemes в 2004 г. и предназначенный для обмена трехмерными моделями, созданными в различных САПР. Формат основан на XML (eXtensible Markup Language, расширяемый язык разметки) для быстрого и простого представления 3D информации [3]. В основе технологии этого языка, активно развивающейся в последние годы, лежит иерархичность структуры документа, разделенного на составные элементы.

Данные могут быть прочитаны и записаны с использованием стандартных инструментов и могут легко интерпретироваться приложениями. Проиллюстрируем применение XML на примерах.

– Широко распространенные браузеры (Internet Explorer, Mozilla и Opera) имеют возможность интерпретации XML.

– Современные ERP и PDM/PLM-системы имеют встроенные средства работы на языке XML (в широко известной системе Vaan реализована оболочка Vaan Open World, обеспечивающая универсальный XML-интерфейс между ERP-системами Vaan и сторонними приложениями).

– СУБД Oracle обладает встроенными средствами для работы с документами XML [4].

В настоящее время ведутся активные разработки систем управления базами данных, работающими на основе языка XML.

В описании формата обеспечивается предустановленный набор признаков, которые всегда могут быть расширены для добавления пользовательских свойств и параметров модели, что обеспечивает практически неограниченные возможности для хранения специализированных данных.

Используя современные алгоритмы сжатия информации, этот формат позволяет пользователям создавать и передавать данные о модели, используя файл, который до 95% меньше, чем подобные файлы в существующих форматах. Например, модель, изображенная на рис. 1, в формате 3DXML занимает примерно 500 кБ памяти, в то время как представление этой же детали в формате IGES заняло бы несколько мегабайт.

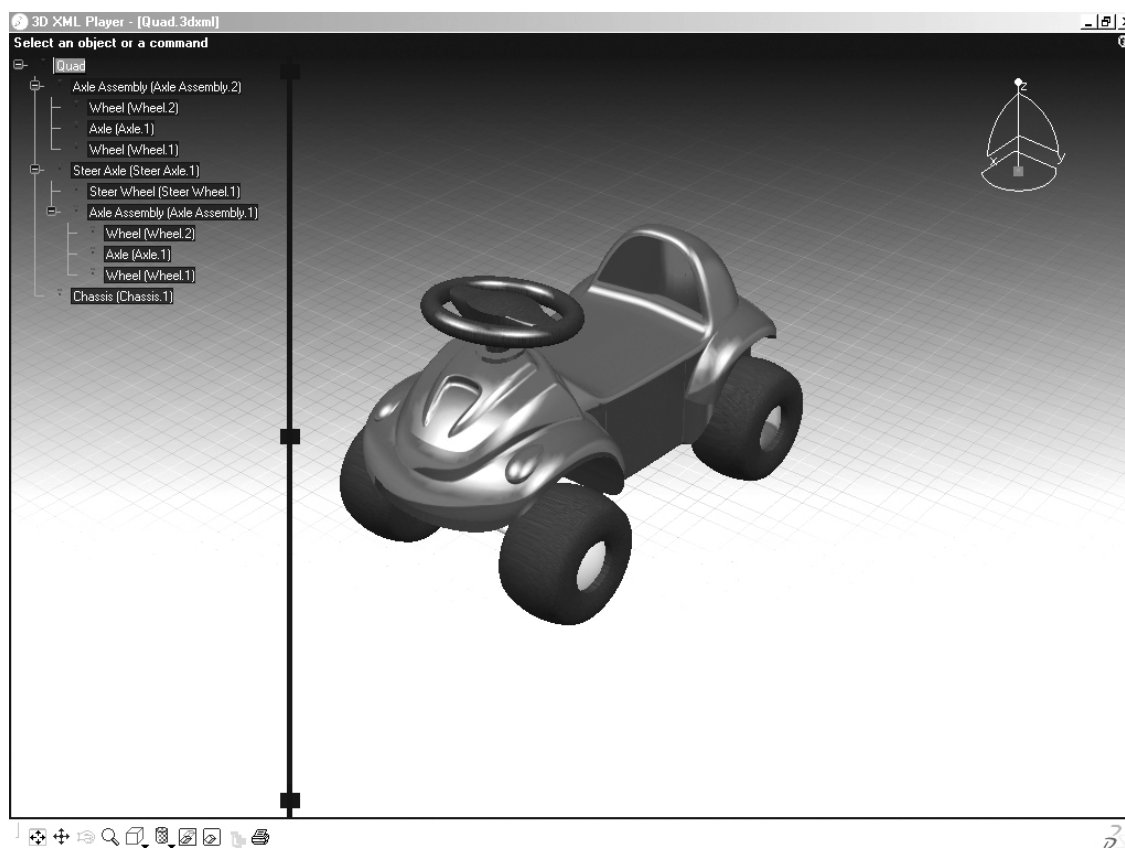


Рис. 1. Модель автомобиля

Большинство современных приложений обмена конструкторской информацией используют удаленный доступ, поэтому скорость передачи данных через сеть, а соответственно, и размер файлов становится одним из важнейших параметров выбора способа хранения данных.

Описание модели

Основой для описания модели является ее геометрия. В XML файле она представляет собой набор примитивов. Поверхности тела описываются с помощью триад,

имеющих общие вершины, а ребра модели – с помощью ломаных линий. Для описания ребер в 3D объекте используется элемент Polyline (ломаная линия). Для того чтобы получить данные о геометрии этого элемента, необходимо в соответствующем XML файле модели найти тег <Edges>, в котором хранится информация об элементах Polyline. Информация о вершинах одной линии находится в теге <Polyline vertices> и записывается в виде перечисления координат (X,Y,Z) точек линии.

Чтобы описать поверхность в 3DXML применяется элемент Triangle (триада). Информация о геометрии триад находится в теге <Face triangles>, где хранятся ссылки на координаты, которые, в свою очередь, располагаются в тэге <Positions>. При заполнении поверхности модели вершины различных триад многократно совпадают друг с другом, что проиллюстрировано на рис. 2, где точка (0) является общей вершиной сразу четырех триад.

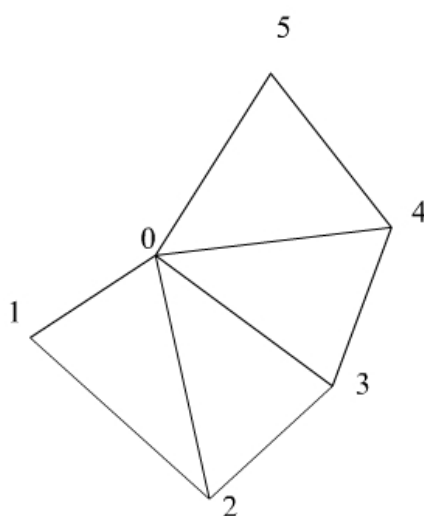


Рис. 2. Пример заполнения поверхности триадами

Такая форма записи способна значительно сократить объем хранимой информации за счет исключения возможности дублирования координат различных точек объекта. При этом следует отметить, что от количества триад зависит качество 3D модели при отображении ее программой-просмотрщиком.

Рассмотренный способ описания геометрии модели в XML файлах значительно облегчает интерпретацию внешними приложениями информации об объекте и обеспечивает точность передачи данных по сравнению со способами представления координат в других форматах. Например, в достаточно распространенном формате IGES внутренний способ представления элемента в системе может отличаться от того, как этот элемент представляется в самом формате [5]. В частности, дуга окружности в какой-то системе может быть определена через центр, радиус и начальный и конечный углы, но в IGES она определяется через центр, начальную точку и конечную точку. В этом случае специализированный IGES-конвертор должен выполнить преобразование с использованием параметрического уравнения дуги. Такое преобразование должно выполняться дважды (при прямой и обратной конвертации), причем каждый раз из-за ошибок усечения и округления значения параметров дуги искажаются. В 3DXML такой проблемы не возникает по причине того, что все точки модели имеют абсолютные координаты в единой системе.

Отдельное описание «каркаса» модели внутри файла позволяет приложениям-обработчикам быстро считывать координаты и, не используя дополнительных матема-

тических преобразований, представить данные для дальнейшей обработки. Это значительно экономит затрачиваемые вычислительные ресурсы.

Структура файла

Структура файла представляет собой дерево (пример на рис. 3), конечными элементами которого являются отдельные модели деталей, при необходимости объединенные в группы элементов.

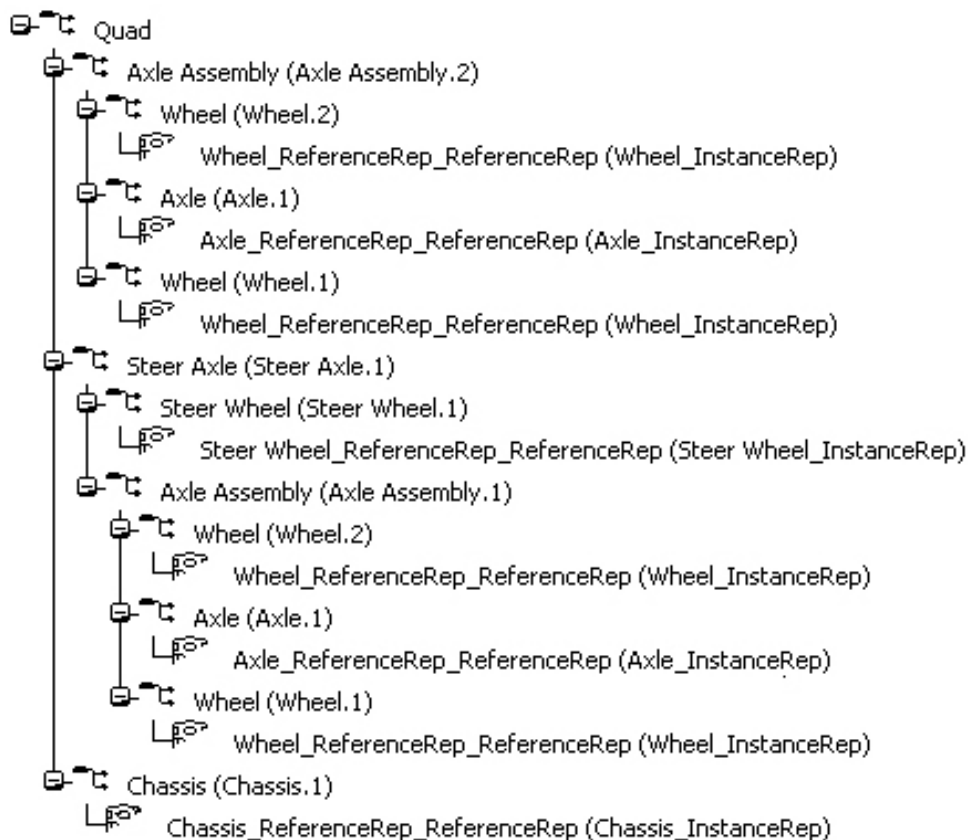


Рис. 3. Структура модели в 3DXML

В описание модели, помимо геометрических параметров, есть возможность включить данные о типе материала, а также любую другую информацию, необходимую разработчику.

Файл 3DXML (рис. 4) представляет собой ZIP-архив, в котором хранятся файлы, описывающие структуру продукта, его геометрию, изображения текстур и сведения о материалах.

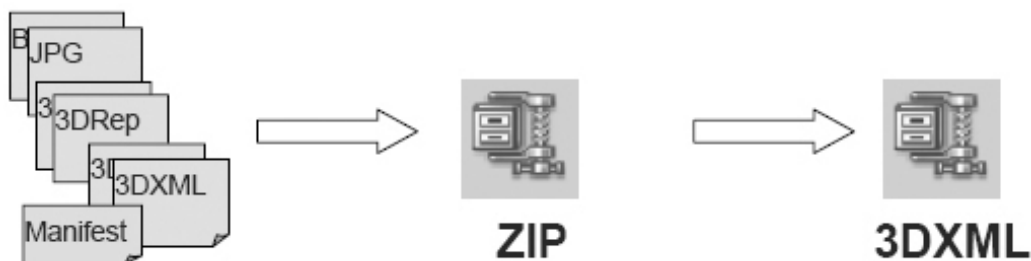


Рис. 4. Состав файла 3DXML

В одном 3DXML файле возможно описание не только модели детали, но и целой сборки, причем для каждого отдельного элемента сборки будет создан свой 3DRep-файл, описывающий геометрию модели, что значительно упрощает и ускоряет работу с файлом 3DXML внешним приложениям. Известно, что при подготовке производственного процесса зачастую приходится вносить изменения в элементы конструкторской документации [6]. Описанный выше способ представления информации о моделях сборки в 3DXML дает возможность вносить поправки и передавать информацию лишь об откорректированной модели или группе элементов, что значительно экономит время. Дополнительно это открывает широкие возможности для систематизации и хранения отдельных элементов в базах данных, поскольку в соответствии с концепцией CALS-технологий (Continuous Acquisition and Lifecycle Support), которые в последнее время получили широкое распространение, для отдельного предприятия необходимо наличие единого хранилища данных об изделии в электронном виде [7]. В противном случае затрудняется обмен данными между ними, делается невозможным сквозное автоматизированное отслеживание и контроль всех процессов проектирования изделий на предприятии, возникают дополнительные трудности при реализации обмена данными с серийными заводами. Возможность раздельного хранения также облегчает создание ассоциативных связей между разработанными 3D моделями и технологическими процессами. Применение сборок в 3DXML решает проблему передачи информации о конкретных моделях для их производства на станках с ЧПУ.

Заключение

Исследование проблемы передачи данных о 3D модели между различными системами показало, что использование формата 3DXML является наиболее выгодным на данный момент, поскольку этот формат имеет ряд преимуществ, таких как малый размер файла и простота интерпретации хранящихся в нем данных о модели. Основанный на XML, он позволяет удаленно корректировать необходимую часть модели без необходимости передачи всего элемента в целом. Возможность добавления в файл описания дополнительных параметров значительно облегчает адаптацию этого формата практически под любой тип производства. При передаче конструкторской документации на производство предприятия-соисполнители смогут отказаться от таких трудоемких и срочных работ, как чтение конструкторских спецификаций предприятия-разработчика, их интерпретация и ручной ввод в свою базу данных; соответственно исчезнут и связанные с этим ошибки.

Применение формата 3DXML позволяет получать данные о 3D модели объекта без использования специализированного программного обеспечения, а также дает возможность создания собственного программного продукта, способного передавать данные в конструкторско-технологические системы.

Литература

1. Ли К. Основы САПР (CAD/CAM/CAE). 1-е изд. – СПб: Питер, 2004. – 560 с.
2. 3DXML User's Guide, Dassault Systemes, 2007 – 76 с.
3. Хабибуллин И. Самоучитель XML. – СПб: БХВ-Петербург, 2003. – 336 с.
4. Джермейни Д., Бурлесон Д. Oracle Application Server 10g. – М: Лори, 2006. – 398 с.
5. «Initial Graphics Exchange Specification IGES 5.x», U.S. Product Data Association US PRO, 2006. – 754 с.
6. Колотнев В., Винничек Л., Кочетова Г. Организация и управление производством. – М.: Колос, 2005. – 464 с.
7. Бакаев В.В., Судов Е.В., Гомозов В.А. Информационное обеспечение, поддержка и сопровождение жизненного цикла изделия. – М.: Машиностроение-1, 2004. – 624 с.

ФОРМИРОВАНИЕ БАЗЫ ЗНАНИЙ ДЛЯ ТПО

Р.В. Бондаренко, Ф.В. Киселев
Научный руководитель – д.т.н., профессор Д.Д. Куликов

В статье рассматривается способ формирования базы знаний для ТПО на основе системы «Т-95», разработанной на кафедре технологии приборостроения.

Проектирование технологических процессов (ТП) – это одна из важнейших функций ТПП. Результатом проектирования технологического процесса является комплекс технологической документации. В этот комплекс могут входить различные документы в зависимости от сложности технологического процесса: технологические карты; ведомости оснастки; карты с операционными эскизами, карты наладки оборудования и карты с управляющими программами для станков с ЧПУ и т.д.

Самым простым вариантом хранения информации о технологическом процессе является использование текстовых файлов, фиксирующих технологические карты как текстовой документ. Такой «документальный» подход к хранению информации о технологическом процессе обладает рядом преимуществ, к которым относятся простота просмотра и вывода на печать технологических карт, а также возможность их корректировки с помощью обычных текстовых редакторов. Однако указанный способ хранения требует большого объема памяти.

Другим способом хранения информации о технологическом процессе является хранение информационной модели технологического процесса, выраженной на формальном языке. Использование для хранения информационных моделей позволяет на порядок сократить объем базы данных для технологических процессов.

На кафедре ТПС была разработана система «Т-95». Эта система предназначена для моделирования технологических процессов (единичных, типовых и групповых), в основном для механической обработки деталей. С помощью системы получают модель технологического процесса, которую будем называть фактографической моделью технологического процесса (ФМТП). Модель ТП вводится в базу знаний и используется для автоматизированного решения технологических задач. Технологические карты, хранимые в базе данных как текстовые файлы, будем называть текстовыми моделями ТП. Достоинствами фактографической модели являются:

- независимость ФМТП от форм технологических документов; на порядок меньший объем информации по сравнению с текстовой моделью ТП;
- простота и удобство редактирования модели;
- возможность вывода любых технологических карт, принятых на конкретном предприятии;
- возможность поэтапного моделирования ТП.

Система имеет свой внешний входной язык для описания фактографических моделей ТП. Кратко рассмотрим грамматику, семантику и лексику входного языка. Как известно, грамматика разделяется на морфологию и синтаксис.

Морфология. Алфавит данного языка содержит следующие элементы: прописные и строчные буквы русского и латинского алфавита; цифры от 0 до 9; специальные символы: + (плюс), – (минус), . (точка); разделители: запятая, пробел; знаки отношений: =, >, <, !, #.

Синтаксис. Слова входного языка представляют собой конечные последовательности букв, цифр и специальных символов. Вводятся следующие виды слов: целое число, действительное число, алфавитное слово, пустое слово. Целые и действительные числа могут быть со знаком и без знака. Число может содержать не более шести цифр. В действительном числе целая часть от дробной части отделяется точкой. Короткое алфавитное слово – произвольная последовательность букв и цифр, содержащая не более

четырёх символов. Длинное алфавитное слово – произвольная последовательность букв и цифр, содержащая не более 16 символов. Пустое слово состоит только из пробелов.

Модель ТП разделяется на структурные единицы, называемые фреймами. В свою очередь, фреймы содержат заголовок и тело фрейма. Заголовок содержит обозначение, называемое меткой (короткое алфавитное слово), и название, которое может быть полным или сокращенным. Тело фрейма содержит слоты, каждый из которых обычно выражает какой-либо параметр ТП и содержит полное обозначение, краткое обозначение (имя), отношение и значение параметра. В качестве значения могут быть целые или действительные числа, а также длинные алфавитные слова.

Входной язык имеет табличную форму описания ТП, при которой информация о технологическом процессе фиксируется в таблице, форма которой указана в табл. 1.

<метка> <полное название фрейма>			
<полное обозначение слота>	<имя>	<отношение>	<значение>

Таблица 1. Форма табличной записи фрейма

В шапке таблицы записывается метка и наименование фрейма. В первой колонке записывается наименование параметра, во второй обозначение параметров, в третьей – обозначение отношения, в четвертой – значение параметра. Перечень возможных отношений и их обозначения приведены в табл. 2. Пример записи в табличной форме приведен в табл. 3.

Отношение	Обозн.	Отношение	Обозн.	Отношение	Обозн.
равно	=	меньше	<	интервал	bet
больше	>	меньше или равно	<=	список	in
больше или равно	>=	не равно	# или !=		

Таблица 2. Отношения и их обозначения

10 РО Токарная			
Номер операции	НОМ	=	10
Номер цеха	ЦЕХ	=	126
Номер участка	УЧС	=	2
Операция	НАИМ	=	Токарная
Код операции	КОД	=	4110

Таблица 3. Фрагмент табличной записи ТП

Лексика ВХЯ представляет собой словарный состав языка, зафиксированный в словаре параметров системы «Т-95». Словарь хранится в базе данных. Наличие словаря позволяет организовать лексический контроль при вводе модели ТП в базу данных и облегчить ввод за счет выполняемого в диалоге переноса имен параметров в модель ТП. Кроме того, за каждым параметром в словаре закреплены тип, размерность, номер классификатора для значений параметра (если это нужно) и ряд других реквизитов параметра. Словарь помещен в базу данных «Образцы, Классификаторы и Словарь» (БОКС).

Наличие словаря позволяет раскрыть семантику применяемых операторов и параметров, так как в словаре за каждым параметром закреплено его полное наименование.

Под семантикой ВХЯ понимается смысловая сторона языковых единиц, которая определяется понятиями, принятыми при проектировании технологических процессов. В семантическом плане технологический процесс рассматривается как система, обладающая общими характеристиками, и состоящая из операций, между которыми установлено отношение следования: за i -ой операцией следует (выполняется) j -ая операция.

Выделены следующие структурные объекты или просто объекты: операция, группа операций, переход, блочный переход, рабочий ход, вспомогательный ход. Под группой операций понимаются операции одного функционального назначения и выполняемых обычно по типовой технологии. К ним относятся заготовительные операции, операции термообработки, гальваники и лакокраски. Для блочного перехода предполагается, что он состоит из одновременно выполняемых инструментальных переходов, при этом обычно несколько инструментов зафиксированы в одном инструментальном блоке. Кроме того, введено понятие псевдоперехода – несуществующий переход, который вставляется в модели между двумя блочными переходами для разделения последних.

Указанные разновидности структурных объектов в системе «Т-95» фиксируются как классы, и им присвоены соответствующие коды. Структура процесса выражается в линейном виде как последовательность операций, причем вслед за каждой операцией записываются переходы в порядке их выполнения.

На кафедре технологий приборостроения разрабатывается модуль определения последовательности обработки поверхностей комплексных элементов на основе типовых планов обработки.

Типовой план обработки поверхности – это направленный граф, у которого вершины p_{zj} отображают состояния поверхностей, а дуги p_i отображают переходы, необходимые для того, чтобы поверхность перешла из состояния p_{zj} в состояние p_{zj+1} .

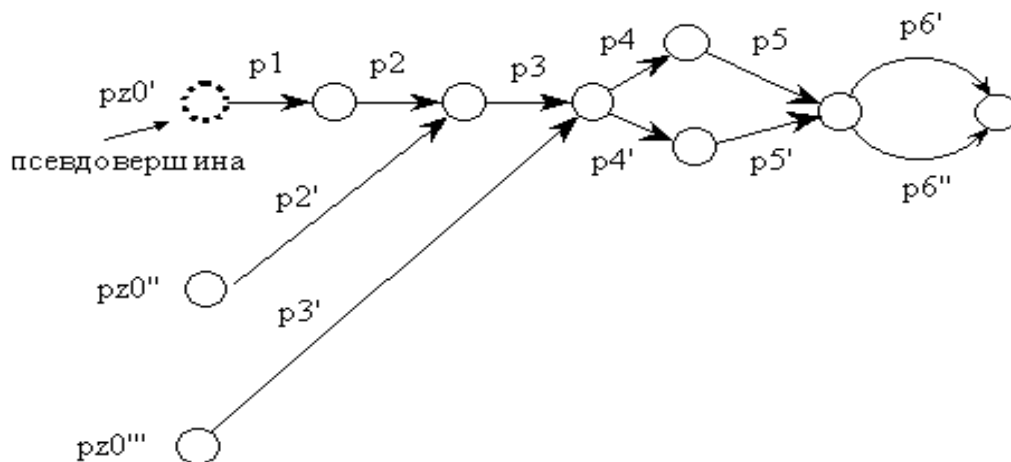


Рис. 1. Типовой план обработки, представленный в виде графа

Главный вопрос, который нужно решить при разработке модуля – как представить в электронном виде граф на рис. 1. Объединим переход и достигаемое им состояние поверхности в единую, совместно информационно обрабатываемую пару. Тогда можно составить таблицу, строки которой – эта самая пара со ссылкой (ссылками) на предыдущее состояние поверхности (предыдущая вершина на графе), фактически на предыдущую пару.

Если бы не стояла задача по вводу типового плана, а только по просмотру, то на этом можно было бы остановиться. В случае же ввода или редактирования ссылка не берется из какой-либо базы данных, а формируется. Также возникает сложность с наличием нескольких исходных заготовок. Тогда граф можно представить в виде двух

таблиц с информацией о различных возможных заготовках и со списком переходов. Для первой достаточно трех столбцов, в которых указывается наименование заготовки, кодовое обозначение наименования и особая метка, не несущая в себе никакой другой информации, кроме идентификации определенной заготовки. Для простоты было решено, что метка будет выражаться числами. Заметим, что порядок чисел и само число не важны, главное, чтобы числа были уникальны и не повторялись.

Во второй таблице содержатся наименования перехода и обрабатываемой поверхности, их кодовые обозначения и номер перехода, являющийся тоже меткой. Последняя, в случае параллельных дуг графа, принимает одинаковое значение для переходов, соответствующих этим дугам. Как видно из рис. 3, во второй таблице также присутствуют столбцы «1», «2», «3». Они введены для возможности отметки первого перехода, который следует назначить при определенном виде заготовки.

Вид заготовок:

Наименование	Обозн.	
		1
		2
		3

Рис. 2. Вид первой таблицы программного модуля, в которой идентифицируется заготовка

Список переходов:

№	Наименование	КодП	Поверхность	КПов	1	2	3
					<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
					<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Рис. 3. Вид первой таблицы программного модуля, в которой содержится список переходов

Теперь рассмотрим, какую информацию вообще должна содержать модель типового плана (рис. 4). Информация о фреймах «заготовки» и «переходы» уже представлена (рис. 2–3). «Параметры состояния» можно выразить в виде таблицы (рис. 5).

Для описания «параметров заготовки» и «параметров перехода», а также для определения уже неоднократно упоминавшегося кодового обозначения вновь обратимся к системе «Т-95». Применяя систему «Т-95» и ее терминологию, приведем описание фрейма МАТР (материал и заготовка) и фрейма ПЕРЕХ (содержание перехода).



Рис. 4. Модель типового плана обработки

Состояние:

Наименование	Обozn.	Отн.	Вел.	
Шероховатость	ШЕРХ	= ▾	<input type="text"/>	Rz ▾
Точность		= ▾	<input type="text"/>	

Рис. 5. Вид первой таблицы программного модуля, в которой отображаются параметры состояния

Наименование	Обозначение	Тип
Марка материала	МАРК	текст
ГОСТ на материал	ГМ	текст
Код группы материала	КОД	целое
Масса детали, кг	МАС	действ.
Наименование заготовки	НАИМ	текст
Код вида	ВИД	целое
Наименование профиля	ПРОФ	текст
Код профиля	КОДП	целое
Размеры:		
длина, мм	L	действ.
ширина, мм	B	действ.
диаметр, мм	D	действ.
высота, мм	H	действ.
диаметр отв. трубы, мм	ДОТ	действ.
Кол-во деталей из заготовки	КОЛ	целое
Масса заготовки, кг	МЗАГ	действ.
Коэф. использования материала, %	ИСП	действ.
Норма расхода материала на деталь, кг	РАСХ	действ.
Обозначение сортамента по стандарту предприятия	СП	текст

Таблица 4. Параметры фрейма МАТР (материал и заготовка)

Для каждой заготовки используется свой набор параметров. Если заготовка – литье или штамповка, то параметры ПРОФ, КОДП и СП не записываются. Если необходимо записать точность размеров заготовки, то для этого используют обозначения точности по стандартам ЕСП. Например, если для диаметра $D=24$ необходимо указать

точность h14, то записывают $D=24h14$. Если нужно указать верхнее и нижнее отклонения, то используют разделитель / для верхнего и \ для нижнего отклонения. Например, длина заготовки L равна 50, а нижнее отклонение равно $-0,6$, тогда $L=600\backslash-0,6$.

Укажем в качестве примера кодовые обозначения некоторых параметров (табл. 5–8).

Наименование	Код
Отливка	01
Горячая штамповка	02
Пруток	03
Проволока	04
Труба	05
Лист или полоса для холодной листовой штамповки	06

Таблица 5. Кодовые обозначения для параметра НАИМ

Наименование	Код
Круг	0
Квадрат	4
Шестигранник	6
Треугольник	3

Таблица 6. Словарь обозначений профиля и коды профиля (параметры ПРОФ и КОДП)

Наименование	Обозначение	Тип значения	Таблица для записи значения параметра
Номер перехода	НОМ	целое	
Наименование перехода	НАИМ	текст	
Вид перехода	ВИД	целое	33
Количество одновременно обрабатываемых поверхностей	КОЛ	целое	
Количество последовательно обрабатываемых поверхностей	КОЛП	целое	
Обрабатываемый объект	ПОВ	целое	34
Обозначение размеров обрабатываемых поверхностей	РАЗМ	целое	
Дополнительная информация об объекте	ДОП	целое	35
Способ выполнения перехода	СПОС	целое	36
Порядковый номер перехода	НОМП	целое	

Таблица 7. Параметры фрейма ПЕРЕХ

Для указания обозначения перехода используется параметр ВИД. В табл. 5 приведены некоторые кодовые обозначения для этого параметра. Для указания количества одновременно или последовательно обрабатываемых поверхностей используется параметр КОЛ или КОЛП. Если $КОЛ = 1$, или $КОЛП = 1$, то параметры не записываются, а по умолчанию предполагается, что обрабатывается одна поверхность или конструктивный элемент.

Наименование	Код	Наименование	Код
Зенкеровать	10	Расточить	26
Зенковать	11	Сверлить	27
Подрезать	17	Строгать	28
Развернуть	22	Точить	30
Рассверлить	25	Шлифовать	33

Таблица 8. Кодовые обозначения для параметра ВИД

Параметр ПОВ предназначен для указания обрабатываемого объекта. В качестве такого объекта могут выступать предметы производства (деталь, заготовка), обрабатываемые поверхности и конструктивные элементы (пазы, канавки, зубья и т.д.). Некоторые коды для обрабатываемых объектов даны в табл. 6. Если для операции выполнена графическая часть, т.е. имеется операционный эскиз, то для указания условных обозначений выдерживаемых размеров или поверхностей используются выражения типа «Поверхность 1» или «Выдерживаемые размеры 1 и 2». Для кодирования таких выражений используется параметр РАЗМ.

Наименование	Код	Наименование	Код
Галтель	5	Поверхность	22
Канавка	12	Резьба	26
Лыска	16	Торец	30
Отверстие	18	Фаска	32
Отверстия	19	Цилиндр	35
Паз	20		

Таблица 9. Кодовые обозначения для параметра ПОВ

Для указания дополнительной информации об обрабатываемом объекте используется параметр ДОП, для кодирования этой информации также существуют коды. Например, «внутренняя» – 1; «глухое» – 2; «наружная» – 6; «сквозное» – 11; «фасонная» – 20. Для указания способа выполнения перехода применяется параметр СПОС. Некоторые кодовые обозначения: «окончательно» – 1; «предварительно» – 6; «на глубину» – 11. При кодировании конкретных переходов используют тот минимальный набор параметров, который необходим для последующего правильного понимания содержания параметра и формирования текста.

Кодирование перехода осложняется тем, что в полном наименовании перехода присутствуют размеры и точность обрабатываемых поверхностей. Поэтому в оператор ПЕР вводятся в качестве параметров обозначения размеров $D, D1, D2, \dots, L, L1, L2, \dots, B1, B2, \dots, P, P1, P2, \dots, C, C1, C2, \dots, \text{УГОЛ}, H1, H2, \dots, M, Z, PN, B$. Например, если в переходе встречается выражение $d = 10 + 0,2$, то к указанным ранее параметрам необходимо добавить: $D = 10 + 0,2$, причем значение параметра является текстом. Если есть выражение «...90° + 30», то добавляют: УГОЛ = 90 + 30–30 мин. Для фаски 1×45 записывают $D = 40 + 0,05 - 0,05$. Под одним номером перехода могут идти несколько переходов, выполняемых одновременно (блочный переход) или последовательно. Для выделения этих переходов используется параметр НОМП, обозначающий порядковый номер перехода (инструментального перехода) внутри общего перехода с номером НОМ.

Кодирование перехода проводится следующим образом: записывается метка, оператор перехода и параметр НОМ, далее НОМП = 1, после этого записывают все параметры первого перехода, далее записывают НОМП = 2 и параметры следующего пере-

хода и т.д., пока не закончится кодирование всех инструментальных переходов. Необходимо помнить, что метка у всех параметров должна быть одна и та же.

Заключение

На базе данного представления разработаны соответствующие приложения, которые позволяют на основе типового выбрать рабочий план обработки, из которого в последствии формируется маршрут обработки детали.

Литература

1. Митрофанов С.П., Гульнов Ю.А., Куликов Д.Д., Падун Б.С. Применение ЭВМ в технологической подготовке серийного производства. – М.: Машиностроение, 1981.
2. Документация на систему кодирования «Т-95».

ПРОЕКТИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ С ИСПОЛЬЗОВАНИЕМ УДАЛЕННОЙ САПР ТП

В.В. Богданов

Научный руководитель – д.т.н., профессор Д.Д. Куликов

В статье рассматриваются возможности современных систем автоматизированного проектирования технологических процессов, предлагается новый подход к реализации процесса проектирования технологических процессов и организации систем автоматизированного проектирования.

Введение

Системы автоматизированного проектирования технологических процессов (САПР ТП) являются связующим звеном между процессами конструирования и производства в компьютерно-интегрированной производственной среде. Целью применения САПР ТП является автоматизация процесса проектирования технологического процесса таким образом, чтобы функции выбора инструмента, оборудования, последовательности операций и переходов и т.д. выполнялись без участия человека. Современные САПР ТП, основанные на использовании различных подходов к реализации процесса проектирования, достигли некоторых успехов в выполнении интеллектуального автоматизированного проектирования технологических процессов, однако многие функции остаются не автоматизированными. Системы не обладают интеллектуальными возможностями, такими как возможность автоматической адаптации технологического процесса в зависимости от имеющихся в наличии ресурсов или организации базы знаний, которые могли бы использоваться всеми модулями системы, занятыми в процессе проектирования.

Средством повышения интеллектуальности систем автоматизированного проектирования может послужить реализация систем на основе распределенного решения технологических проблем. В основе методологии распределенного подхода к решению проблем лежит принцип декомпозиции крупных задач на более мелкие и распределении этих задач между набором специализированных модулей [1]. Модули для решения задач, полученных в результате декомпозиции, называются узлами или агентами. Кооперация агентов происходит на уровне разделения и совместного использования знаний о проблеме и методах ее решения. Каждый агент может решать достаточно сложные задачи, но задачи, которые необходимо решить для глобального повышения уровня автоматизации проектирования технологических процессов, таковы, что ни один из агентов не имеет достаточных данных, ресурсов и интеллектуальности для полноценного их решения. В то же время использование сети агентов позволит решить набор субпроблем и затем объединить полученные результаты в единое решение. Решение наследственно-распределенных проблем аналогично процессу принятия решений на уровне организации, когда решения, принятые несколькими ответственными сотрудниками, оказывают влияние на подразделения компании или бизнес-события.

На сегодняшний день существуют достаточно развитые и отработанные средства для автоматизации и компьютеризации принятия решений (так называемые ODSS системы и их распределенный вариант – DDSS системы). Использование технологий и методов, применяемых в DDSS системах для организации САПР ТП, позволит значительно повысить уровень автоматизации технологической подготовки производства.

Обзор современных САПР ТП

САПР ТП является неотъемлемой частью компьютерно-интерированной среды производства. Технологический процесс и его параметры, сформированные при помощи САПР ТП, необходимы для организации работы современного производственного

предприятия. На сегодняшний день существует два метода проектирования, применяемых в системах автоматизированного проектирования технологических процессов: метод поиска или вариантный метод и метод синтеза [2].

В системах, использующих метод поиска, применяется принцип групповой технологии, т.е. все детали объединяются в группы, и каждая из групп имеет типовой план обработки. Ограничением таких САПР является то, что новая деталь может быть обработана только в том случае, если она принадлежит к какой-либо из групп, информация о которых занесена в БД САПР ТП; требуется определенная степень модификации плана обработки при появлении новой детали. Хотя эти системы в значительной мере облегчают труд технолога, они не обеспечивают интеграции процесса проектирования детали и процесса разработки технологического процесса.

В системах, использующих метод синтеза, различная информация о процессе и его параметрах (режущий инструмент и технологическая оснастка, последовательность операций и т.д.) формируется в процессе проектирования. Ограничениями таких САПР ТП являются сложность описания геометрии детали и нехватка важных функциональных модулей системы, таких, как, например, подсистема автоматического выбора оборудования. Некоторые САПР ТП могут генерировать список всех возможных операций, применимых к данной группе деталей, и затем на основе эвристического анализа выдавать рекомендации о степени их пригодности для обработки конкретной детали. Однако современные САПР ТП не в состоянии адаптировать такой список к имеющемуся в наличии оборудованию. Математические методы, предложенные для построения последовательности операций и группировки деталей, не подходят для решения других технологических задач.

Проектирование технологических процессов можно представить в виде иерархической структуры, на нижних уровнях которой выполняются четко заданные наборы действий, например, выбор оборудования, инструмента, режимов резания и т.д., в то время как на верхних уровнях происходят процессы, которые требуют принятия достаточно сложных решений [2]. С одной стороны, жесткая иерархическая структура с четким подчинением уровней обеспечивает быстроту обработки информации, в то время как, с другой стороны, такая структура ограничивает возможности внесения изменений в систему.

Комплексность систем с иерархической структурой быстро растет в зависимости от размеров системы, что приводит к росту затрат на разработку, внедрение и поддержку таких систем. Другим препятствием на пути создания САПР ТП является необходимость интеграции различных источников данных, которые используются в работе различных подсистем САПР ТП. Прямая интеграция источников данных невозможна в связи с различным представлением и детализацией данных. Повышение уровня автоматизации САПР ТП требует большей автономности от систем и подсистем.

Децентрализованный подход к построению САПР ТП

В противовес традиционному централизованному подходу к построению САПР ТП можно предложить методологию, успешно применяемую в системах принятия бизнес-решений DDSS. Эта методология основана на применении децентрализованной группы интеллектуальных агентов, каждый из которых выполняет функции по решению какой-либо части поставленной задачи. Как показано на рис. 1, каждый агент должен состоять из модуля принятия решения (модуля бизнес-логики), локальной базы знаний и модуля, обеспечивающего связь с другими агентами [3].

Процесс проектирования технологического процесса обеспечивается координацией функционирования локальных агентов. Глобальный запрос, поступающий группе агентов, подвергается декомпозиции, и полученные субзапросы передаются соответст-

вующим агентам. Частные решения, полученные локальными агентами, объединяются и формируют ответ на полученный глобальный запрос.

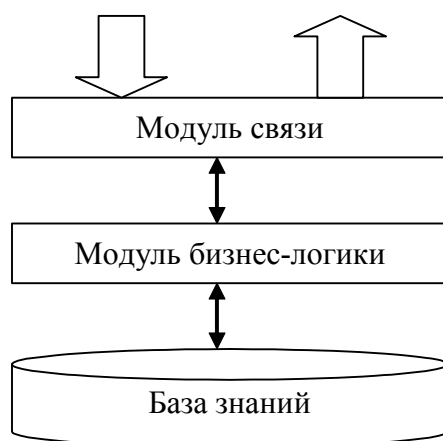


Рис. 1. Схема организации агента

Такой подход обладает следующими особенностями.

- Повышается адаптивность системы.
- Система становится модульной.
- За счет параллельной работы подсистем повышается скорость обработки.
- Повышается надежность системы.
- Можно снизить потребление ресурсов системой за счет организации обмена прогнозируемой информацией и знаниями между агентами.
- Контроль за выполнение процесса автоматизированного проектирования упрощается.

В современных САПР ТП для выполнения проектирования в автоматическом режиме необходимо выполнить две задачи. Первая задача состоит в анализе модели детали, полученной при помощи САД системы. Вторая задача состоит в обработке полученных после выполнения первой задачи данных и формировании технологического процесса. В рамках выполнения этих двух задач САПР ТП должна выполнять, в том числе, следующие действия:

- взаимодействие с процессом проектирования для получения более полной информации о детали;
- формирование процесса для каждого этапа обработки;
- выбор инструмента и оснастки;
- выбор режимов обработки.

Проблемы, которые должна решать САПР ТП, распределены в нескольких различных областях знаний. Применение распределенной сети агентов, каждый из которых функционирует в своей области знаний, при обеспечении качественного координации работы этих агентов позволит решить задачу автоматизации проектирования технологических процессов более полно. Узлы, представляющие отдельные области знаний, способны сформировать качественные решения в своей области знаний, которые затем составят единый результат. Кроме того, от современной САПР ТП требуется возможность формирования нескольких вариантов технологического процесса. При использовании сети распределенных агентов этого можно добиться за счет того, что каждый агент может формировать несколько решений и передавать их координирующему модулю. Координирующий модуль, производя композицию полученных от агентов решений, сможет сформировать большее число вариантов процесса. Кроме того,

наличие большого выбора частных решений позволит решать возникающие в процессе проектирования проблемы более эффективно [3].

В условиях современного рынка в процесс создания продукции вовлекаются многочисленные внешние участники – от поставщиков комплектующих, которые должны иметь возможность оперативно реагировать на изменения в требованиях к конечному продукту, до самих заказчиков, которые хотят получить доступ к процессам формирования этих требований. Для многих крупных производителей «виртуальное предприятие» становится реальностью – они выносят за скобки собственного производственного процесса разработку и выпуск комплектующих, а подчас и собственно сборку готового изделия, оставляя за собой базовые операции выработки концепции и проектирования продукции. Передача части своих функций на аутсорсинг не отменяет необходимости контролировать и интегрировать все процессы. Такой подход к организации производства требует от САПР ТП возможности организации удаленной работы. При использовании децентрализованного подхода к организации САПР ТП возможно территориальное разделение агентов, что позволит реализовать кооперацию территориально удаленных участников процесса разработки технологического процесса.

Реализация децентрализованной САПР ТП при помощи технологии web-сервисов

С учетом требований, предъявляемых к САПР ТП следующего поколения, а также ориентируясь на современные методы построения информационных систем для производственных предприятий, представляется целесообразным разрабатывать САПР ТП на основе технологии web-сервисов. Web-сервис – это программный компонент, доступный через глобальную (или локальную) вычислительную сеть и не привязанный к каким-либо конкретным языкам программирования или операционным системам. Web-сервис использует стандартизованную систему обмена сообщениями, в которой применяется расширяемый язык разметки XML. Технология web-сервисов основана на создании гибких платформонезависимых решений, позволяющих значительно снизить затраты на внедрение и общую стоимость владения программным обеспечением [3].

Системы, построенные по технологии web-сервисов, являются, как правило, модульными и многокомпонентными. Модули такой системы могут выполнять следующие роли:

- реестр сервисов (Service Registry) – централизованный каталог, в котором перечислены все компоненты системы, предоставляющие сервисы, а также описание этих сервисов, их свойства и другая разнородная техническая информация, необходимая для правильной работы системы;
- поставщик сервисов (Service Provider);
- потребитель сервисов (Service Requestor).

При этом один и тот же модуль может выступать как в качестве поставщика сервисов, так и в качестве потребителя. Например, модуль расчета режимов резания будет выступать потребителем сервисов по отношению к модулю работы с базой оборудования и одновременно поставщиком сервисов для модуля, использующего результаты расчетов. Для связи отдельных модулей в единую информационную систему применяется система обмена XML сообщениями. Система обмена сообщениями может использовать различные технологии передачи данных, а именно:

- удаленный вызов процедур XML (XML RPC);
- протокол доступа к объектам сервисов (SOAP);
- передача XML документов посредством базовых методов стандартного интернет-протокола HTTP.

Все три технологии передачи данных могут использовать в качестве транспорта один из общепринятых транспортных Интернет-протоколов – HTTP, FTP, SMTP – или специально

разработанный протокол связи ВЕЕР. Применение стандартных протоколов связи сети Интернет позволяет осуществлять обмен информацией между компонентами системы вне зависимости от технических особенностей организации каналов связи и их удаленности.

Каждый узел сети агентов и координирующий модуль (ядро системы) при построении САПР ТП на основе децентрализованного подхода и технологии web-сервисов будет представлен в виде web-сервиса. Для обеспечения целостности системы все модули будут зарегистрированы в служебном реестре сервисов. Реестр может быть реализован как часть управляющего модуля системы или в виде UDDI реестра сервисов предприятия. Схема такого решения приведена на рис. 2.

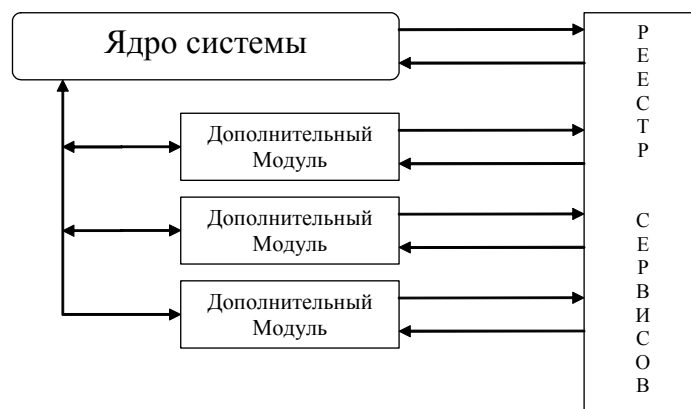


Рис. 2. Реализация САПР ТП при помощи web-сервисов

Технология web-сервисов предполагает открытый доступ к функциям, методам и описанию сервисов, что значительно упрощает процесс разработки новых модулей и организацию межпрограммного взаимодействия в рамках построения единой информационной системы предприятия. Все компоненты системы регистрируют предоставляемые ими сервисы в едином системном реестре сервисов. В процессе работы каждый компонент системы посылает в реестр запрос на обнаружение необходимого сервиса. Реестр сервисов формирует ответ, в котором указывает расположение запрашиваемого сервиса и метод связи с ним. На основании данных, полученных из реестра сервисов, потребитель сервисов формирует пакет данных и отправляет его поставщику сервисов. Обработанные данные поставщик сервисов передает в ответном пакете данных. Таким образом, система может быть не только распределенной, но и динамически изменяемой.

Заключение

Распределенная децентрализованная система автоматизированного проектирования технологических процессов является гибким инструментом, позволяющим значительно повысить уровень автоматизации технологической подготовки производства, а так же организовать взаимодействие предприятий-партнеров.

Литература

1. Rojer Hamilton, Jerald Wasserman. Automating business processes and decision making, IBM Press, 2003.
2. Куликов Д.Д., Падун Б.С., Яблочников Е.И., Скуратов А.К., Тихонов А.Н. Методы автоматизации ТПП в приборостроении и машиностроении. – СПб: СПбГУ ИТМО, 2006.
3. Yasuhiko Takahara, Yongmei Liu. Foundations and Applications of MIS, Springer Press, 2006.

ПРОЕКТИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В PDM-СИСТЕМЕ ЛОЦМАН: PLM

Б.А. Ильин, С.А. Мелих

**Научные руководители – д.т.н., профессор В.А. Валетов,
д.т.н., профессор Д.Д. Куликов**

На этапе технологической подготовки производства мы, как правило, имеем большую часть информации о жизненном цикле изделия. В данной статье освещены преимущества ведения проектирования технологических процессов при использовании PDM-систем. Также внимание акцентируется на конкретных компонентах PDM-системы ЛОЦМАН: PLM.

Как правило, на этапе технологической подготовке производства мы получаем большую часть информации о жизненном цикле изделия. Сейчас речь пойдет о том, какие преимущества мы можем получить, ведя проектирование технологических процессов при использовании PDM-систем. Эти системы предназначены для управления информацией об изделиях на всех жизненных стадиях изделия (управление инженерными данными). Рассматриваемой PDM-системой в данной статье является распространенная и широко используемая система ЛОЦМАН: PLM, разработанная компанией «АСКОН» (Россия).

Благодаря развитому инструментарию PDM-системы организовано эффективное взаимодействие между такими компонентами, как электронный архив, средства автоматизированного документооборота, CAD/CAM/CAE-системы, средства Internet-технологий. Рассмотрим подробнее, каким требованиям отвечает ЛОЦМАН: PLM.

Важной компонентой ЛОЦМАН: PLM является электронный архив, функционирующий на базах трех СУБД MsSQL, InterBase, Oracle (архитектура «клиент- сервер»). С помощью электронного архива осуществляется авторизованный доступ ко всем информационным ресурсам технологической подготовки производства. Возможность быстрого доступа к чертежам деталей, операционным эскизам, технологическим картам является неоспоримым плюсом в работе технологов и необходимым условием для эффективной реализации принципа преемственности технологических решений. Просмотр объектов осуществляется с помощью вьюверов (просмотрщиков), позволяющих просматривать и графические файлы основных CAD/CAM систем. Также в ЛОЦМАН: PLM организован удобный гибкий поиск по реквизитам учетной карточки объектов.

Возможны различные подходы к организации баз знаний: экспертные системы, табличные алгоритмы, фреймовое представление знаний и т.д. Для САПР ТП характерно представление знаний на основе табличных алгоритмов. Задание правил в виде таблиц соответствий, информационно-логических таблиц, алгоритмических таблиц и других форм табличных алгоритмов позволяет достаточно просто записывать технологические правила и организовывать их хранение в базе знаний. Доступ к базам знаний довольно просто осуществить, используя дерево проектов PDM-системы. Если во время работы с САПР ТП появляется необходимость в корректировке либо дополнении технологических правил, то используется модуль сопровождения базы знаний. Таким образом, постепенно накапливаются как общие, так и локальные технологические правила, отражающие специфику предприятия. База знаний организуется на основе объектно-ориентированного подхода.

Сложность решения задач управления процессами автоматизированного проектирования технологии характеризуется такими факторами, как существование большого количества файлов с моделями разных видов и программных средств их обработки, трудность отслеживания логической связи между файлами, необходимость организации хранения разнородной информации, необходимость учета факторов, связанных с контролем реализации стадий жизненного цикла изделия.

С помощью PDM- системы осуществляется удобное ведение проекта изделия, древо проекта выводится на экран (на рис. 1). Каждая вершина дерева – это папка, содержащая модели объектов или образы документов.

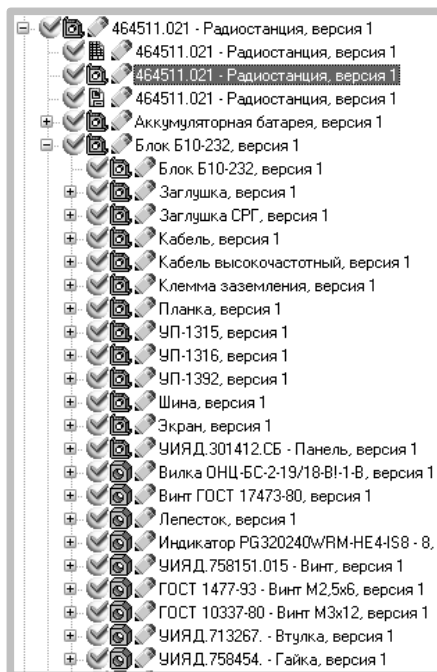


Рис. 1. Дерево изделий

Для каждой папки имеется учетная карточка, содержащая атрибутивную информацию. Каждый объект, входящий в дерево проектов, в своей учетной карточке имеет поле для фиксации состояния объектов: «в работе», «утвержден», «взят на изменение», «серийное производство» и т.д. В учетной карточке фиксируется имя автора, дата начала работы над объектом и дата завершения работы над ним. Эти атрибуты позволяют руководителям отслеживать жизненный цикл любого документа.

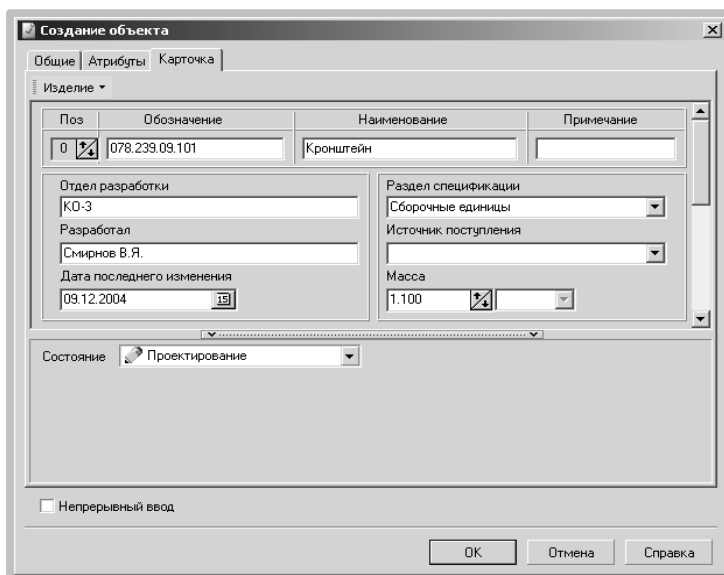


Рис. 2. Учетная карточка

Каждый объект дерева проекта принадлежит какому-либо классу объектов, например, «Маршрутная карта», «Операционная карта» и т.д. Поиск объектов по их классам позволяет ускорить нахождение нужного объекта в дереве проекта. В итоге исполь-

зование дерева проекта позволяет обеспечить высокую степень наглядности и удобство работы в многопользовательском режиме работы.

Организация взаимодействия между проектантами и контроль их деятельности в ЛОЦМАН: PLM ведется с помощью модуля Workflow. Эта технология основана на составлении сложноподчиненной связи делового процесса, в которой фиксируется последовательность прохождения документов. Благодаря этому модулю организуется согласованное взаимодействие пользователей и обеспечивается жесткий контроль над процессом проектирования технологического процесса и его утверждение у конкретных ответственных лиц. PDM-система дает возможность распараллелить процесс согласования документов и синхронизировать проведение изменений на основе приема ReLine («красная линия»). С помощью этого приема могут работать такие специалисты, как нормоконтролер, главный конструктор, главный технолог и прочие, т.е. специалисты, осуществляющие контроль и утверждение различных рабочих чертежей и документации. Если, к примеру, чертеж требует доработки, либо выполнен с ошибками, специалист, осуществляющий его проверку, не будет вынужден править его ручкой и идти с ним к конструктору, который допустил эту ошибку, а может всего лишь воспользоваться аннотированием этого чертежа с помощью технологии «красного карандаша», т.е. указать на ошибку и написать аннотацию (например, «Исправить немедленно»), не отходя от рабочего компьютера в системе ЛОЦМАН: PLM. Аннотацию можно проводить на всех стадиях хождения документа, таких аннотаций может быть не одна, а несколько, и их можно просматривать в системе ЛОЦМАН: PLM. Для того чтобы утвердить какой-либо документ или чертеж, существует электронная цифровая подпись (ЭЦП), что также позволяет сократить время на согласование документов.

Заключение

Исследования, проведенные на кафедре технологии приборостроения СПбГУ ИМТО, показали, что применение PDM-систем позволяет в 1,5–2,5 раза сократить трудоемкость и стоимость проектирования технологических процессов и повысить качество разрабатываемой технологической документации (сокращение ошибок), а также существенно сократить время на выпуск документации.

Литература

1. Куликов Д.Д., Падун Б.С., Яблочников Е.И. Методологические основы построения АСТПП // Научно-технический вестник СПбГУ ИТМО. – 2007. – №22. – С. 114–121.
2. Зильбербург Л.И., Молочник В.И., Яблочников Е.И. Реинжиниринг и автоматизация технологической подготовки производства в машиностроении // Компьютербург. – 2003. – С.152.

ПОСТРОЕНИЕ БАЗЫ ДАННЫХ РЕЖУЩЕГО ИНСТРУМЕНТА В АСТПП

А.В. Смагин

Научный руководитель – к.т.н., доцент Е.И. Яблочников

Большинство программных продуктов, составляющих современные комплексы АСТПП на предприятии, используют ту или иную информацию о режущем инструменте, что порождает разрозненность баз данных режущего инструмента и дублирование данных в них. В работе рассматривается распределение данных о режущем инструменте по подсистемам АСТПП и организация управления данными в едином информационном пространстве предприятия.

Введение

Современные системы АСТПП промышленных предприятий представляют собой сложные структуры, включающие множество компонентов, решающих определенные задачи. САД-системы применяются для разработки трехмерных моделей и выпуска чертежно-графической информации. В САМ-системах разрабатываются управляющие программы для оборудования с программно-числовым управлением. САЕ-системы позволяют проводить различные инженерные расчеты и исследования, используя конечно-элементные методы. САРР-комплексы сокращают объем рутинной работы технологов при проектировании технологических процессов.

Корпоративные информационные системы (КИС) МRP, МRPИ и ERP – системы комплексного управления предприятием, включающие финансовые, логистические, складские и т.п. модули. PDM-система управляет данными об изделии и играет объединяющую роль построения единого информационного пространства предприятия (ЕИП) [1].

В процессе проектирования технологического процесса механообработки различные подсистемы АСТПП оперируют определенными технологическими данными, в том числе данными о режущем инструменте (РИ) (рис. 1).



Рис. 1. Режущий инструмент в АСТПП и КИС

При проектировании технологического процесса механообработки технолог в САРР-системе из существующей базы данных режущего инструмента (БД РИ) выбирает необходимый для каждой операции РИ. Процесс выбора может происходить в ре-

жиме поискового диалога либо подбираться по определенным правилам экспертной системой [2].

При проектировании операций механообработки на оборудовании с ЧПУ в САМ-системе технолог-программист формирует набор режущего инструмента, необходимый для формирования управляющей программы (УП), из существующей библиотеки РИ САМ-системы.

При осуществлении проверки корректности УП до отправки ее на станок в системе верификации УП технолог-программист использует тот же набор инструмента, что и при проектировании УП. Таким образом, система верификации УП также имеет свою библиотеку РИ.

В процессе планирования производства в ERP-системе для своевременного обеспечения механообрабатываемых операций также должны учитываться данные об используемом инструменте.

И, наконец, PDM-система – объединяющее звено ЕИП предприятия – имеет БД РИ, используемого при производстве изделий, в привязке к каждому изготавливаемому наименованию детали.

Таким образом, в едином информационном пространстве предприятия возникает пять (по числу рассматриваемых компонентов) технологических баз данных режущего инструмента.

Зачастую на предприятиях применяется программное обеспечение различных производителей, имеющее значительные различия в структурах баз данных и использующее различные СУБД. Такое многообразие БД не позволяет организовать единую унифицированную централизованную или распределенную базу данных, которая могла бы использоваться всеми подсистемами, что влечет за собой многочисленные ошибки, вызванные большим числом типоразмеров (порядка 10 тысяч) режущего инструмента, а наполнять библиотеки приходится разным исполнителям. Для сокращения ошибок пополнения БД РИ и более функционального использования данных об инструменте в ЕИП предприятия необходимо определить необходимый набор параметров для каждой подсистемы, разработать структуру централизованной БД РИ и обеспечить синхронизацию с ней всех библиотек прикладного программного обеспечения.

Основная часть

Необходимый в работе каждой из подсистем АСТПП набор параметров режущего инструмента определяется из функционального назначения программы.

В ходе проектирования технологического процесса механообработки в системе САРР, технолог определяет необходимый инструмент. Критериями выбора для этой системы являются:

- тип операции (токарная, фрезерная, и т.д.),
- вид операции (расточка, сверление, и т.д.),
- геометрия заготовки (диаметр прутка, толщина листа, и т.д.),
- геометрия инструмента (диаметр, вылет, и т.д.),
- материал заготовки.

Эти критерии позволяют определить параметры режущего инструмента, которые должны храниться в БД САРР: *обозначение* инструмента для точной идентификации его в системе, *геометрия* инструмента, *режимы резания* (в привязке к материалу заготовки), *схематическое изображение* инструмента для наглядного его представления.

Обозначение инструмента представляет собой буквенно-цифровой идентификатор, сформированный по правилам классификации, регламентированным ГОСТами или стандартами предприятия. Он может содержать основные геометрические характеристики инструмента, его материал и, для однозначной идентификации, название производителя.

При необходимости выполнения механообработки на программно-управляемом оборудовании технолог-программисту необходимо разработать управляющую программу. В процессе разработки УП в САМ-подсистеме технолог-программист должен использовать набор инструментов, определенный на этапе проектирования операционной технологии в САРР-системе. Возникает необходимость интеграции двух подсистем АСТПП на уровне передачи технологических данных о проектируемой операции, в том числе и об используемом режущем инструменте.

Современные САМ-системы предлагают широкий выбор различных стратегий обработки, учитывающих технологические параметры процесса резания. САМ-система формирует УП, основываясь на данных о режимах резания и параметрах оборудования.

В процессе разработки УП САМ-системы используют математические методы моделирования удаления материала заготовки инструментом, построенным по его геометрическому описанию. Для сложного профильного инструмента, который невозможно описать параметрически, используется представление в виде твердотельной модели, построенной в САД-системе и определенным образом описанной в БД [3].

При необходимости изготовления специализированного инструмента используются соответствующие модули САД/САМ-программ, которые позволяют на основе модели детали со сложной геометрией разработать форму РИ, используемого при производстве таких деталей. Такие модели также должны иметь привязку к базе данных, формируя обозначение, геометрические параметры и режимы резания спроектированного РИ.

Таким образом, в технологической базе данных РИ САМ-системы (рис. 2) необходимо хранить: идентификатор инструмента, геометрические параметры, режимы резания (в привязке к обрабатываемому материалу), схематическое наглядное изображение, *твердотельную модель* (для сложного профильного инструмента).

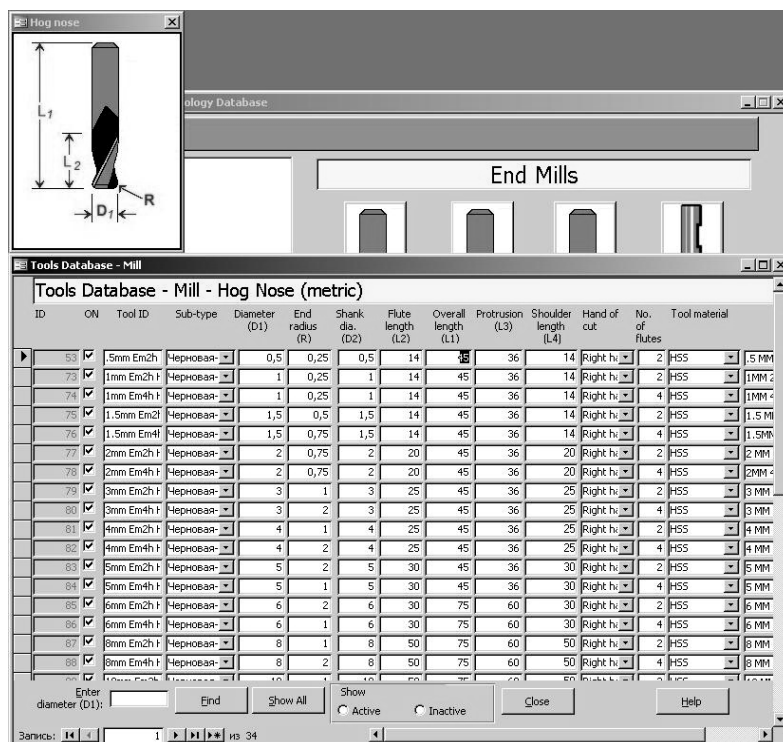


Рис. 2. Выбор инструмента в БД САМ системы

После постпроцессирования управляющей программы в машинные коды станка с ЧПУ рекомендуется проводить верификацию УП для проверки корректности и предотвращения повреждения оборудования. Для этих целей существует целый ряд программных продуктов верификации УП.

Входной информацией для программы-верификатора является управляющая программа в машинных кодах, наименование станка и перечень используемого инструмента, сформированный в САМ-системе. При наличии прямого интерфейса верификаторы вызываются непосредственно из САМ-программы, и вся необходимая информация передается напрямую. При отсутствии такой интеграции передача необходимых данных может быть осуществлена посредством PDM-системы.

В соответствии с УП на трехмерной модели реального станка обрабатываются все перемещения рабочих органов и режущего инструмента. Кроме контроля столкновения агрегатов станка, система-верификатор позволяет выявлять зарезы технологической оснастки. Кроме того, верификаторы УП позволяют проводить оптимизацию режимов резания и траектории перемещения органов станка. Для проведения полноценной верификации в программе-верификаторе должны быть модели станка, заготовки, технологической оснастки, режущего инструмента.

Исходя из функционального назначения программ проверки управляющих программ, технологическая база данных режущего инструмента должна содержать идентификаторы, режимы резания в привязке к материалу и твердотельные модели инструмента [4].

После завершения разработки технологии механообработки формируется перечень необходимых материалов и производственных ресурсов. В ERP-системе производится учет, а также оформление документации для хранения, закупки и перемещения всех материальных ценностей предприятия. БД режущего инструмента используется для учета хранения на складах и формирования документов внутреннего перемещения между складами и производственными площадками, а также закупочных ведомостей. Необходимыми параметрами такой базы данных являются обозначение РИ и его внутренний идентификатор.

Связующим звеном для всех модулей АСТПП предприятия является PDM-система, концентрирующая все данные жизненного цикла изделия. Программа управления данными об изделии осуществляет передачу данных между всеми подсистемами АСТПП и КИС, реализуя функции интеграционных интерфейсов. В хранилище PDM помещаются твердотельные модели деталей изделия (ДСЕ), технологическая документация (технологические процессы изготовления), управляющие программы для станков с ЧПУ (при обработке на программно-управляемом оборудовании), все сопутствующие данные (проекты САЕ) и др.

PDM является ключевым элементом построения единого информационного пространства предприятия, вследствие чего все компоненты АСТПП должны иметь интеграцию, прежде всего, с системой управления данными об изделии [1].

База данных РИ PDM-системы является концентрацией всех библиотек компонентов АСТПП и КИС. Она должна содержать все необходимые параметры РИ для всех подсистем:

- *обозначение инструмента,*
- *вид операции,*
- *тип операции,*
- *геометрия инструмента,*
- *режимы резания,*
- *схематическое изображение,*
- *твердотельная модель.*

База данных инструмента в PDM-системе формируется в соответствии с классификацией, принятой на предприятии.

Таким образом, система управления данными об изделии позволяет передавать необходимую информацию об используемом в производстве режущем инструменте между подсистемами АСТПП и КИС, реализует возможность синхронизации разроз-

ненных библиотек прикладных программ и, тем самым, сокращает количество ошибок в технологических базах данных РИ.

Ведущие производители режущего инструмента поставляют каталоги своей продукции в электронном виде. Такие электронные каталоги содержат всю необходимую и достоверную информацию о РИ, включая и твердотельные модели в форматах наиболее распространенных САД-систем. Интерфейсные оболочки позволяют экспортировать необходимые данные во внешние приложения. Существенным недостатком такого решения является ограниченность поддерживаемых программ, список которых ограничивается наиболее распространенными САМ-системами и системами верификации УП. Но эта проблема может быть решена написанием приложения для импорта-экспорта данных в требуемые библиотеки.

Еще одним решением автоматизированного управления данными о режущем инструменте являются так называемые «менеджеры режущего инструмента» (TDM, Tool Data Manager, рис. 3). Эти системы имеют интерфейсы с электронными каталогами ведущих производителей инструментов, популярными САД, САМ и приложениями верификации УП. TDM-системы имеют модули проектирования инструмента, конфигурирования сборок и магазинов инструмента, хранения технологических параметров, трехмерных моделей, планирование производства и закупки инструмента [5].

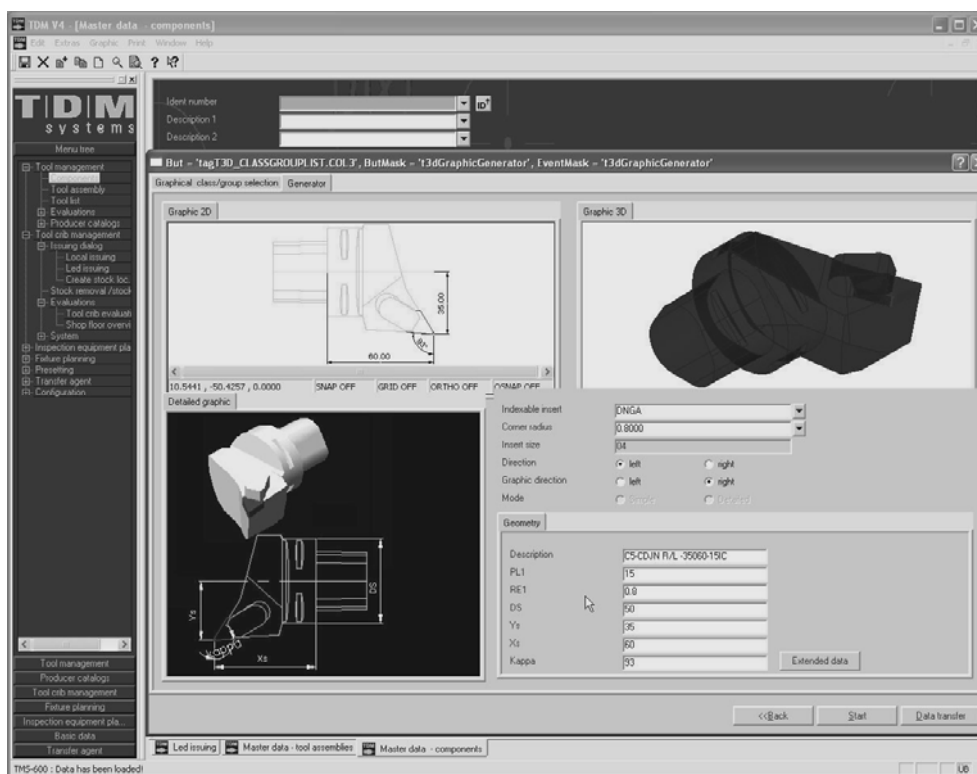


Рис. 3. Представление инструмента в TDM системе

Применение таких программных продуктов должно быть реализовано в интеграции с PDM-системой предприятия.

Многообразие модулей АСТПП и КИС предприятия порождает многообразие БД РИ. Реализация управления данными о режущем инструменте в едином информационном пространстве предприятия должна осуществляться на основе централизованной БД PDM-системы с синхронизацией библиотек прикладных программ. Пополнение и актуализация данных о РИ должна производиться на основании данных производителя инструмента из электронных каталогов. Использование централизованной БД РИ и синхронизация библиотек инструмента PDM-системой позволит значительно сократить количество ошибок в технологических базах данных.

Литература

1. Яблочников И.Е. Методологические основы построения АСТПП / Методическое пособие. – СПб: СПбГУ ИТМО, 2005. – 84 с.
2. Воскресенская Е.А., Воскресенский А.Д., Степанов А.В., Рева В.Н. Опыт внедрения PLM-системы на промышленном предприятии. / CAD/CAM/CAE Observer. – 2005. – № 6. – С. 19–23.
3. Обзор по материалам разработчиков и зарубежных САМ-изданий. Интеллектуальные возможности современных САМ-систем. / CAD/CAM/CAE Observer. – 2002. – № 4. – С. 2–7.
4. <http://www.cgtech.com> /Электронный ресурс. – Режим доступа: свободный. – Яз. англ.
5. <http://www.tdmsystems.com> /Электронный ресурс. – Режим доступа: свободный. – Яз. англ., нем., франц.

МИКРОГЕОМЕТРИЯ ПОВЕРХНОСТИ И ЕЕ ФУНКЦИОНАЛЬНЫЕ СВОЙСТВА

О.С. Юльметова

Научный руководитель – д.т.н., профессор В.А. Валетов

В статье приведен анализ существующих способов описания микрогеометрии, рассмотрены этапы процесса оптимизации, включающего поиск, нормирование, технологическое обеспечение и контроль оптимальной микрогеометрии.

Введение

Повышение качества выпускаемой продукции является важнейшей задачей промышленности. Эта задача особенно актуальна в приборостроении. В общем случае качество определяется большим набором показателей. В рамках данной статьи под качеством понимается набор функциональных свойств, а именно: твердость, прочность, коррозионная стойкость и т.д. В каждом конкретном случае одна из характеристик может выступать более существенной по сравнению с остальными.

Ю.Г. Шнейдером доказано [1], что на два десятка функциональных свойств поверхности, таких как износостойкость, коррозионная стойкость, отражаемость световых лучей, теплоотражаемость, электрическая прочность, контактная жесткость, пылеудаляемость и др., существенно влияет микрорельеф поверхности. Поэтому создание оптимального микрорельефа функциональных поверхностей деталей, наряду с улучшением других характеристик поверхностного слоя, является одним из основных и наиболее эффективных способов значительного повышения качества изделий.

Методы описания микрогеометрии

В силу ряда многочисленных и часто независимых причин профиль поверхности есть реализация случайной функции. На сегодняшний день известно два способа описания микрорельефа поверхности: параметрический и непараметрический. Параметрический подход широко применяется на практике, что обусловлено следующими факторами:

- (1) традиционность;
- (2) стандартизация (ГОСТ 2789-73);
- (3) наличие аппаратной поддержки.

Основным преимуществом этого метода является простота. Суть метода заключается в указании совокупности усредненных параметров профиля, таких как R_a , R_z и др. (рис. 1). На рисунке обозначено: R_{max} – расстояние между линией выступов профиля и линией впадин; R_a – среднее арифметическое отклонение профиля (y_i); S_m – среднее арифметическое значение шага неровностей профиля; S – среднее арифметическое значение шага неровностей профиля по вершинам; R_z – сумма средних арифметических абсолютных отклонений точек пяти наибольших минимумов (H_{imin}) и пяти наибольших максимумов (H_{imax}).

В рамках параметрического подхода технологическое обеспечение функционального свойства поверхности сводится к достижению значения шероховатости, указанной на чертеже. А это фактически означает невозможность оптимизации микрорельефа для конкретного функционального свойства с помощью стандартных параметров. Ситуация усугубляется еще и тем, что совокупность параметров часто не отражает действительный характер рельефа. Подтверждается это простым примером: два зеркальных профиля, описывая абсолютно разный рельеф, имеют одинаковый набор параметров (рис. 2).

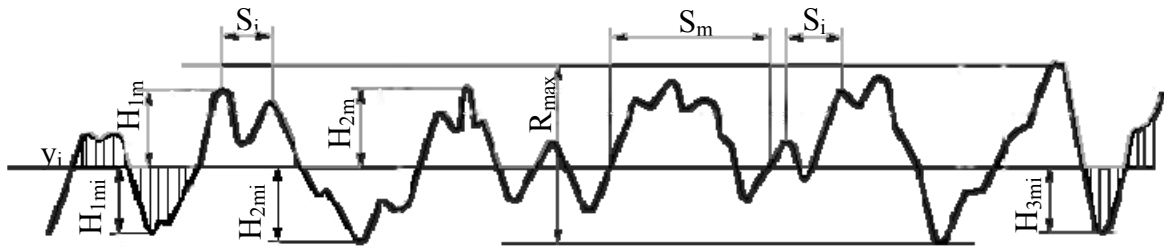


Рис. 1. Профиль шероховатости поверхности и его характеристики

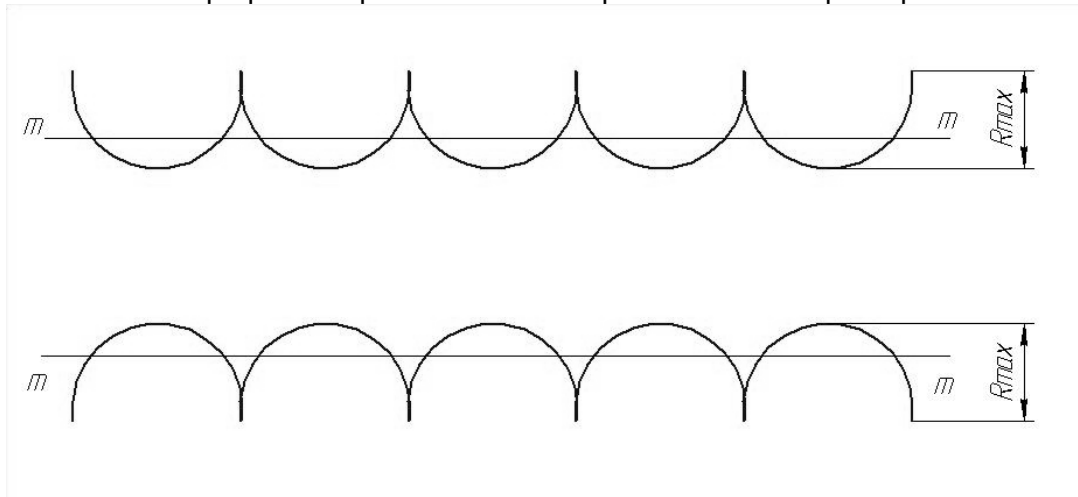


Рис. 2. Профили двух зеркально противоположных поверхностей

Возникла необходимость в создании нового метода. В.А. Валетовым предложен непараметрический метод, в рамках которого детально отработаны все этапы процесса оптимизации, включающего поиск, нормирование, технологическое обеспечение и контроль оптимальной микрогеометрии [2]. Суть непараметрического метода состоит в следующем.

- (1) При описании микрорельефа вместо усредненных характеристик использовать графические изображения плотностей распределения ординат или углов наклона профилей, а лучше сами профили или микротопографии поверхностей.
- (2) Нормировать нужно конкретные функциональные свойства, например, проставляя на знаках шероховатости их стандартные номера.
- (3) При экспериментальном определении наилучшего из возможных микрорельефов для конкретного функционального свойства мы не только находим этот микрорельеф, но и фиксируем технологию его получения.
- (4) Контроль микрогеометрии серийной продукции производим наложением графического изображения функции контролируемой поверхности на эталонное изображение микротопографии.

Непараметрический метод дает возможность решить ряд конкретных проблем в приборостроении, связанных с оптимизацией микрогеометрии поверхностей с целью повышения качества приборов [3].

Практическая значимость

Проблема повышения качества с использованием непараметрического подхода актуальна как никогда, тем более что другие источники повышения качества исчерпаны. Необходимость в непараметрическом методе выявилась при разработке технологии изготовления ротора микромеханического гироскопа – разработка ЦНИИ «Электро-

прибор». Диаметр кремниевой пластинки ротора составляет 3 мм, толщина 60 мкм. На этом диске химическим травлением необходимо получить гребенки, ширина которых составляет 0,01 мм. Неоптимальная микрогеометрия на стенке паза вызывает необходимость приложить для работы механизма усилия, в десятки раз превышающие расчетные. Параметрический подход бессилён, так как не отражает действительный характер микрорельефа, и с его помощью микрогеометрию паза не описать. Непараметрический метод позволит снизить существующий процент брака, экспериментально выявив оптимальную микрогеометрию для такого свойства поверхности, как гладкость.

Методика поиска оптимальной микрогеометрии:

- (1) изготовить серию кремниевых пластин с различной исходной микрогеометрией, при этом методы и режимы изготовления зафиксировать;
- (2) снять профиль поверхности с каждого образца;
- (3) подвергнуть все образцы одинаковому функциональному воздействию;
- (4) выявить образец с наилучшим уровнем функционального свойства поверхности;
- (5) принять микрорельеф наилучшего образца как оптимальный.

Эта последовательность действий позволит выявить технологию получения наилучшей микрогеометрии и, одновременно, критерий ее оценки и контроля в виде конкретного графического изображения принятой в качестве критерия функции.

Заключение

В ходе проделанной работы:

- (1) проанализированы существующие методы описания микрогеометрии;
- (2) рассмотрены этапы оптимизации микрогеометрии в рамках непараметрического способа описания микрорельефа;
- (3) разработана методика поиска оптимальной микрогеометрии для конкретного функционального свойства поверхности.

Литература

1. Шнейдер Ю.Г. Эксплуатационные свойства деталей с регулярным микрорельефом. – СПб: СПбГИТМО (ТУ), 2001. – 264 с.
2. Валетов В.А., Мурашко В.А. Основы технологии приборостроения. Учебное пособие. – СПб: СПбГУ ИТМО, 2006. – 180 с.
3. Мусалимов В.М., Валетов В.А. Динамика фрикционного взаимодействия. – СПб.: ПИМаш, 2006. – 168 с.

ОБ АРХИТЕКТУРЕ РОБОТОВ И ИНТЕЛЛЕКТУАЛЬНОМ УПРАВЛЕНИИ ИМИ

Ю.С. Окуловский, А.С. Шека

(Уральский государственный университет им. А.М. Горького)

Научный руководитель – д.ф.-м.н., профессор В.Ю. Попов

(Уральский государственный университет им. А.М. Горького)

Описана архитектура колесного робота и системы управления им. Рассмотрены основные устройства, используемые для создания колесных роботов. Приведена классификация языков для управления роботами. Изложены подходы к применению интеллектуальных алгоритмов для решения основных задач управления, связанных с колесными роботами.

Введение

Наука об архитектуре, создании и управлении роботами называется робототехникой. Робототехника использует достижения многих других предметных областей: механики, электротехники, теории управления, распознавания образов, системного программирования и так далее. Некоторые области робототехники хорошо изучены, например, создание автоматических станков и линий производств. С другой стороны, в управлении автономными устройствами до сих пор продолжаются активные исследования, и нет каких-либо устоявшихся, хорошо изученных моделей.

В связи с растущей актуальностью использования роботов в современном мире особую важность приобретает подготовка специалистов в этом направлении. Для формирования общего восприятия проблемы необходимо изучать не конкретные модели роботов, но некую типовую, обобщенную и расширяемую модель. При этом, с одной стороны, требуется знакомство будущего специалиста с различными аспектами робототехники: механикой, электротехникой, системами управления, программным обеспечением и так далее. С другой стороны, для специализации в каком-либо направлении требуется возможность детального изучения одного из аспектов без углубленного изучения других. В этом смысле колесные роботы выгодно отличаются от, например, андронидных роботов. Колесные роботы обладают сравнительно простой механикой и электроникой. Более того, качественная механика для колесных роботов (колесные базы с подвесками, амортизаторами и т.д.) может быть заимствована из моделей, которые используются в соревнованиях радиомоделистов. Подобные колесные базы доступны по цене, обладают хорошим качеством исполнения и могут быть приспособлены для целей робототехники.

Сравнительно простая модель колесного робота вызывает, тем не менее, большой интерес в мировом сообществе. Так, например, широко известны соревнования колесных роботов, играющих в футбол [1], или задачи на движение по пересеченной местности автономных автомобилей [2]. Колесные роботы имеют и прикладное значение: исследование труднодоступных областей (таких, как пирамида Хеопса или поверхность Марса), автоматическое движение по городу в соответствии с правилами дорожного движения («автоматическое такси»), и т.д.

Также особую важность имеет изучение интеллектуальных алгоритмов применительно к робототехнике. Для колесных роботов остаются актуальными многие сложные задачи управления. В частности, даже движение по прямой линии является весьма сложной задачей, поскольку сцепление колес с поверхностью зависит от изменяющихся свойств этой поверхности, в результате чего простые алгоритмы оказываются неспособными обеспечить устойчивое прямолинейное движение. Кроме того, колесный робот является автономным устройством, и для него актуальны такие проблемы, как исследование помещений, ориентирование на местности и т.д. Наконец, колесный робот может быть снабжен дополнительными устройствами, например, ковшом экскаватора

или крановой установкой. Дополнительные устройства позволяют решать еще больший спектр задач и при этом не вызывают существенного изменения механической структуры робота. Решение подобных задач традиционными методами не всегда эффективно. Использование нейронных сетей, генетических алгоритмов [3], алгоритмов коллективного разума [4] и других интеллектуальных систем позволяет решать эти задачи с большей эффективностью. Кроме того, в области приложения интеллектуальных систем к робототехнике имеется большой потенциал для исследований, поскольку на сегодняшний день эти темы исследованы в недостаточно полном объеме, как в нашей стране, так и за рубежом.

В первой части данной работы изложена архитектура устройств колесного робота и систем управления. Во второй части приведена классификация языков управления. В третьей части изложены подходы к применению интеллектуальных систем в робототехнике.

1. Архитектура колесных роботов

1.1. Архитектура электронных устройств

Принципиальная схема взаимосвязей основных компонентов робота изображена на рис. 1.

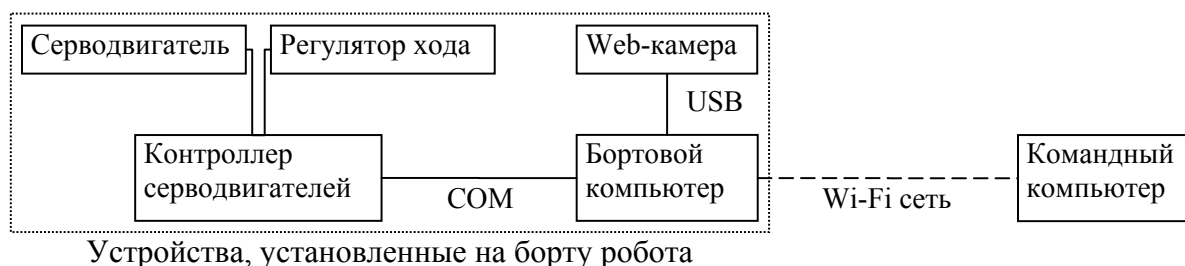


Рис. 1. Схема взаимосвязей основных компонентов робота

1.2. Серводвигатели

Серводвигатель – это любой электрический двигатель, скорость или положение которого управляется с помощью замкнутого контура обратной связи. Серводвигатели бывают нескольких стандартных размеров с большим диапазоном рабочих параметров. Определенным преимуществом использования таких серводвигателей является то, что из-за применения их в любительских подвижных механизмах они относятся к продукции массового производства, а значит, стоят относительно недорого. Такие серводвигатели имеют трехконтактный разъем, по двум подается питание, а по третьему входной сигнал.

Первоначально эти устройства были разработаны для корректировки поверхностей управления моделей самолетов во время полета. Управляющий интерфейс должен был быть простым и надежным, при этом управление несколькими элементами должно было происходить одновременно с помощью простых радио сигналов. Время жизни аккумулятора и затраты на производство тоже являлись важным критерием.

Так как в большинстве случаев для корректировки в полете нужны только небольшие изменения положения, диапазон вращения серводвигателей был ограничен 90°. Эти вращения часто переводились в линейные движения – перемещение закрылок (вверх и вниз), поднятие, и опускание шасси и т.д. Редукционная передача давала возможность относительно небольшому двигателю на коротком расстоянии производить большой крутящий момент.

Управляющий входной сигнал серводвигателя представляет собой последовательность квадратных импульсов длиной от 1 мс до 2 мс, частота повторения которых

лежит в диапазоне от 20 Гц до 60 Гц. Положение выходного вала пропорционально ширине входного импульса – это управление с широтно-импульсной модуляцией (ШИМ).



Рис. 2. Импульс 1 мс

Импульс длительностью 1 мс поворачивает выходной вал вправо, т.е. по часовой стрелке (рис. 2).

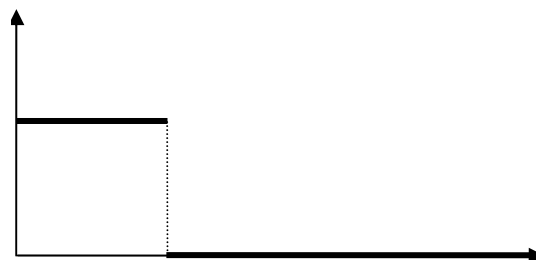


Рис. 3. Импульс 2 мс

Импульс длительностью 2 мс поворачивает выходной вал влево, т.е. против часовой стрелки (рис. 3).



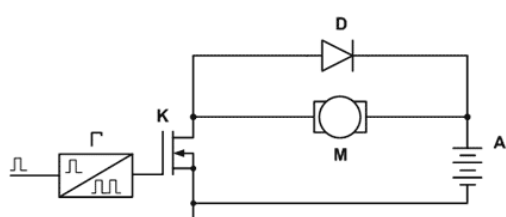
Рис. 4. Импульс 1,5 мс

Импульс длительностью 1,5 мс поворачивает выходной вал к центру его вращения (рис. 4).

1.3. Регулятор хода

Регулятор хода используется для регулирования количества оборотов на силовом двигателе, который обеспечивает движения вперед и назад. Он включается между аккумулятором и силовым электродвигателем. Регулятор имеет такой же трехконтактный разъем, как и у серводвигателя, и такой же входной сигнал управления ШИМ. В простейшем случае в задачу регулятора входит регулирование потока мощности от аккумулятора к двигателю. При длительности входного импульса 1 мс двигатель выключен,

при 2 мс двигатель развивает максимальную мощность. В промежутке мощность плавно изменяется. Типовая структурная схема включения представлена на рис. 5.



Г – управляющий генератор;
 К – силовой ключ;
 D – диод;
 М – электродвигатель;
 А – аккумулятор.

Рис. 5. Схема регулятора хода

Входной импульс поступает на управляемый генератор импульсов Г. Частота импульсов постоянна, а вот их длительность зависит от длительности входного импульса следующим образом (рис. 6).

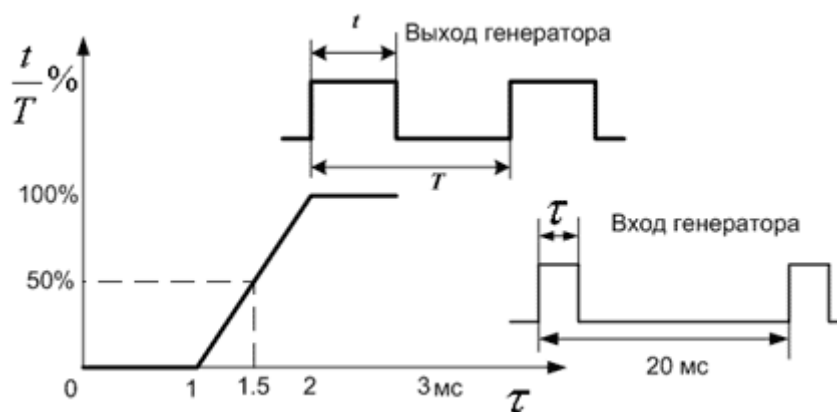


Рис. 6. Зависимость входного и выходного импульсов в регуляторе хода

При длительности выходного импульса 0% на выходе генератора их просто нет – там низкий уровень управляющего ключом сигнала. Ключ К закрыт, ток через двигатель М не течет. При длительности выходного импульса 100% от периода на выходе генератора тоже импульсов нет, но уровень управляющего ключом сигнала высокий. Ключ К открыт, и все напряжение от аккумулятора А приложено к двигателю М. Он развивает при этом максимальную мощность.

При промежуточном значении длительности канального импульса на выходе генератора присутствуют импульсы с длительностью, определяемой по приведенному графику. Скажем, при канальном импульсе 1,5 мс на выходе генератора присутствуют импульсы, длительность которых составляет половину периода. Соответственно, ключ К половину периода открыт, половину – закрыт. Напряжение в точке 1 повторяет эту форму. Данная организация работы регулятора хода и устройство коллекторного двигателя позволяет организовать плавное изменение скорости вращения двигателя. При этом двигатель будет крутиться с постоянной скоростью, без подергиваний, на всех скоростях.

Регуляторы хода также являются продукцией массового производства и стоят относительно недорого. Разные модели регуляторов немного отличаются, и поэтому на управляющей аппаратуре требуется небольшая калибровка.

1.4. Контроллер сервоприводов

Контроллер сервоприводов – это микроконтроллер с прошитым программным обеспечением. Он генерирует сигналы ШИМ для серводвигателей и регуляторов хода согласно спецификации. В продаже имеется несколько контроллеров, которые способ-

ны выполнять такую деятельность. Они принимают сигнал на COM порт и вырабатывают соответствующие сигналы для механики. Но у них всех различные протоколы общения через COM порт, что привязывает конечную архитектуру к конкретной модели сервоконтроллера.

В нашем случае используется контроллер ZX-SERVO16. На начальной стадии этот контроллер удовлетворяет потребностям. Но для более свободной архитектуры нужно использовать контроллеры с собственной прошивкой, так как это позволяет нам расширять и изменять функционал контроллера.

Задача программирования контроллера для управления сервоприводами является наглядной, что может быть использовано для увеличения эффективности обучающего процесса. Программирование контроллера происходит на низкоуровневом языке, что демонстрирует различные архитектуры.

1.5. Web-камера

Web-камера используется для снятия изображения с борта машины. В случае ручного управления видео передается в полном объеме. При использовании автоматических систем управления видео передается частично, что позволяет сэкономить трафик. В простом варианте центр изображения цветной, а края черно-белые. Задача о фильтрации изображения имеет определенный интерес. Необходимо выяснить, какая часть изображения, в какой цветовой палитре и как часто должна отсылаться. Для решения этой задачи используются нейронные сети и генетические алгоритмы.

1.6. Бортовой компьютер

На борту робота установлен компьютер архитектуры x86. Эта распространенная архитектура, что облегчает работу с ней. Для нее существует большое количество инструментов программирования и периферийных устройств. Это позволяет с минимальными затратами проводить расширение и изменение архитектуры робота. Бортовой компьютер использует две операционные системы: Windows и Linux. Первая операционная система более популярна и универсальна для рядовых пользователей, что позволяет с легкостью осваивать робота, но ее через функциональность и закрытость вызывает ряд проблем, связанных с производительностью и расширением. В целом Windows используется для «обкатки» программ. В свою очередь, Linux – открытая операционная система, но на начальной стадии работы с ней возникают проблемы. Linux имеет разнообразные дистрибутивы, что позволяет собрать систему с максимальной производительностью. Кроме того Linux используется на встраиваемых системах, отличных от архитектуры x86, что уменьшит затраты в случае формирования решения, основанного на дешевых системах.

Бортовой компьютер предполагает расширение робота, что влечет за собой использование множества различных устройств, поэтому необходим легко расширяемый протокол общения. Все данные от бортового к командному компьютеру передаются с помощью одного TCP/IP соединения. Формат данных от командного к бортовому и от бортового к командному компьютерам одинаков. Пакеты имеют следующий формат:

ID	Длина пакета	Данные устройства
← 1 байт →	← 1 байт →	← не более 253 байт →

1.7. Командный компьютер

Модель робота снабжена рулевым сервоприводом и регулятором хода силового двигателя. Оба устройства подключены к контроллеру ZX-SERVO16. Этот контроллер получает сигналы от материнской платы бортового компьютера через COM-интерфейс

и выполняет управление сервоприводом и регулятором хода. Также к материнской плате бортового компьютера подключена Web-камера через шину USB. Наконец, бортовой компьютер снабжен Wi-Fi адаптером для связи с другими компьютерами по беспроводной сети.

Преимуществом такой организации является расширяемость модели. Так, контроллер ZX-SERVO16 позволяет осуществлять управление 16 сервоприводами. Соответственно, в случае расширения модели робота дополнительными манипуляторами (экскаваторный ковш, крановая установка, система управления Web-камерой) необходимые сервоприводы могут быть подключены к имеющемуся контроллеру. Также существуют специальные контроллеры для оцифровки сигналов, которые позволяют преобразовывать аналоговые данные в цифровые и передают их по СОМ-интерфейсу. Соответственно, робот может быть снабжен датчиками ускорения, лазерными дальнометрами, гироскопами и другими устройствами. Также преимуществом данной модели является использование стандартных устройств и интерфейсов. Соответственно, для расширения модели не требуется пайка или аппаратное программирование микросхем.

2. Языки управления роботами

Предлагается шестиуровневая программная модель управления роботами: микроконтроллерный уровень автоматных состояний, элементарных движений, структурного моделирования, сценариев, интеллектуального управления.

Микроконтроллерный уровень предполагает обеспечение взаимодействия аппаратуры робота (сервоприводов и т.д.) с программными комплексами более высокого уровня. Этот уровень включает в себя всю аппаратно-зависимую часть программного обеспечения и позволяет использовать программные комплексы более высокого уровня вне зависимости от конкретных моделей роботов и установленных на нем устройств. В частности, для управления сервоприводами используется язык контроллера сервоприводов. Команды на этом языке передаются через интерфейс СОМ.

Уровень автоматных состояний позволяет программировать элементарные действия исходя из специфики аппаратной части, но не из конкретики модели. Например, уровню автоматных состояний соответствует команда «повернуть руль на 10° влево». Соответственно, программное обеспечение, выдающее подобные команды, способно работать с любой новой моделью колесных роботов, если для этой новой модели будет создан язык микроконтроллерного уровня.

Уровень элементарных движений полностью абстрагирован от аппаратной части и позволяет управлять роботом заданием конкретной последовательности движений.

Уровень структурного моделирования позволяет выделять специфические аппаратные комплексы и узлы, давать им определенные задания, разрешать конфликты между узлами, обеспечивать решение задач в динамическом режиме и автоматически решать широкий спектр первичных задач жизнеобеспечения робота. Для программирования на уровне структурного моделирования предполагается использовать современные языки (С#, Java и другие). Взаимодействие между этими языками и языком уровня элементарных движений осуществляется путем объявления соответствующих программных интерфейсов, которые обеспечивают объектно-ориентированное представление робота.

Программирование на пятом уровне предполагает создание сценариев, состоящих из конкретных задач без спецификации способа их реализации. Последний, шестой уровень предназначен для формирования цели, планирования, обучения и интеллектуального решения алгоритмически трудных задач, возникающих в рамках реализуемых сценариев. Также шестой уровень предназначен для общего мониторинга деятельности робота и задания глобальных параметров функционирования системы.

3. Задачи интеллектуальных алгоритмов в робототехнике

3.1. Интеграция с интеллектуальными вычислениями

Для выполнения интеллектуальных вычислений используется специализированная библиотека GANS [5, 6]. Данная библиотека позволяет собирать интеллектуальные системы (нейронные системы или генетические алгоритмы) из простых компонентов. Библиотека написана на C# и имеет открытый исходный код.

При использовании объектно-ориентированного представления робота, с одной стороны, и библиотеки интеллектуальных вычислений – с другой, становится возможной разработка программ управления роботом «изолированно» от тонкостей алгоритмов искусственного интеллекта и особенностей взаимодействия с устройством.

Известно, что решение задач с помощью методов искусственного интеллекта является достаточно ресурсоемким процессом. С другой стороны, системы искусственного интеллекта часто без труда могут быть распараллелены. Поэтому необходимо предусмотреть возможность распараллеливания этих алгоритмов на узлы высокопроизводительной компьютерной техники. Библиотека GANS содержит возможность распараллеливания, и был разработан специальный инструментарий для запуска интеллектуальных алгоритмов на узлах кластера [7].

3.2. Распознавание изображений

Обработка изображений, полученных с камеры робота, необходима для решения задач управления. В частности, необходимо выделение препятствий, оценка расстояния до них и т.д.

Наша обработка изображений основывается на использовании нейронных сетей. Первоначально необходимо выполнить преобразование «фон/фигура», т.е. отделить значимые изображения от общего фона. Для выполнения этого преобразования была использована нейронная сеть, обученная на галерее изображений. Эта сеть анализирует пиксели изображения и определяет принадлежность каждого пикселя к фону или значимому объекту. Затем связные области пикселей, принадлежащих объектам, группируются и направляются на последующий анализ.

Для построения полосы препятствий использовались такие объекты, как крупные кубики, кегли и т.п. Для определения типа объекта на фотографии используется батарея нейронных сетей. Каждая из них умеет распознавать свой тип объектов (т.е. кубик, кеглю и т.д.). Анализатор подает изображение объекта на все нейронные сети и выбирает наиболее уверенный ответ.

Преимуществом подобной архитектуры является ее расширяемость. Так, если изменился фон (например, робот приехал в новую, до сих пор не посещаемую комнату), то анализатор «фон/фигура» может быть дополнительно обучен работе в новых условиях. Аналогично, если появился новый, до сих пор неизвестный тип объектов, он будет идентифицирован как неизвестный, поскольку ни одна нейронная сеть не сможет уверенно распознать его. Соответственно, робот задает вопрос оператору относительно названия этого объекта и обучает дополнительную нейронную сеть.

Распознавание изображения важно также для контроля поведения роботов. Рассмотрим, например, типичную задачу для колесного робота – «сдачу на права». Это означает, что робот должен миновать некоторую полосу препятствий, не повредив и не опрокинув препятствия. Контроль над выполнением этого упражнения может быть возложен на видеокамеру. Такая видеокамера устанавливается так, чтобы в ее обзор входило все пространство, на котором робот осуществляет передвижение. С помощью описанных распознающих алгоритмов контролирующая программа строит карту местности, отмечая на ней робота, препятствия и их состояния, посторонние объекты и т.д. Также эта программа способна отметить направление движения робота, его скорость и

другие параметры движения. С помощью такой контролирующей программы интеллектуальные алгоритмы, ответственные за перемещение робота, получают всю необходимую информацию для коррекции своего поведения.

В случае одного робота та же функциональность может быть обеспечена дополнительными датчиками на борту робота, на препятствиях и так далее. Однако это ведет к увеличению радиотехнической составляющей экспериментов и удорожанию стендов. Кроме того, в случае взаимодействия двух и более роботов контроль над взаимодействием в любом случае приходится возлагать на стороннее устройство.

3.3. Управление колесным роботом

При управлении колесным роботом оказывается, что даже самые простые действия – движение по прямой, поворот, разворот и т.д. – сложны для детерминированных алгоритмов. Движение робота зависит не только от частоты вращения двигателя и поворота рулевых колес, но и от непредсказуемых факторов, например, сцепления колес с поверхностью. Соответственно, отдаваемые команды должны быть скорректированы с реальным движением робота.

Если подобную коррекцию выполняет детерминированный алгоритм, то может оказаться, что реальное движение успело измениться до того, как алгоритм завершил свою работу, и, соответственно, рассчитанная коррекция не может быть применена. Поэтому требуется применение алгоритмов, которые постоянно соотносят свой результат с реальными условиями.

Подобный алгоритм был также построен на основе нейронных сетей. Нейронная сеть постоянно обучается на примерах действительной реакции робота на отданные команды. В результате становится возможным моделировать реальное движение автомобиля с большой скоростью, хотя и с некоторыми ошибками. После этого оптимальная последовательность команд может быть подобрана с использованием генетических алгоритмов.

Литература

1. DARPA Grand Challenge. – Режим доступа: <http://www.darpa.mil/grandchallenge/>
2. Federation of International Robot-soccer Association. – Режим доступа: <http://www.fira.net/>
3. Нейронные сети, генетические алгоритмы и нечёткие системы. / Рутковская Д. и др. – М: Горячая линия-Телеком, 2004.
4. Ant Colony Optimization / M. Dorigo, T. Stützle. MIT Press, 2004.
5. A model and implementation of universal engine for neural systems. / Окуловский Ю.С., 9-й международная конференция "Интеллектуальные системы и компьютерные науки, труды конференции, II том. – Москва, 2006. – С. 21.
6. GANS home page. – Режим доступа: <http://ai.math.usu.ru/wiki/index.php?title=GANS>
7. Комплекс распределения интеллектуальных вычислений на кластер под управлением операционной системы Windows / Морнев М.Л., Окуловский Ю.С. Четвертая сибирская школа-семинар по параллельным и высокопроизводительным вычислениям. Программа и тезисы. – Томск, 2007. – С. 49.

ОПТИМИЗАЦИЯ КОНФИГУРИРОВАНИЯ И РАСПРЕДЕЛЕНИЯ ЗАКАЗОВ ВИРТУАЛЬНОГО ПРЕДПРИЯТИЯ

А.А. Саломатина, Ю.Н. Фомина

Научный руководитель – к.т.н., доцент Е.И. Яблочников

Оптимизация распределения заказов при конфигурировании виртуальных предприятий – задача, заслуживающая отдельного внимания. Авторами предлагается решение на основе использования генетических алгоритмов и мультиагентных технологий. В настоящее время проводятся практические исследования данной методики.

Одним из основных этапов создания промышленных изделий является технологическая подготовка производства (ТПП). Недостаточная оснащенность проектирования и производства, а также отсутствие кооперации с другими предприятиями при решении задач ТПП приводит к увеличению сроков запуска новых изделий в производство, неоправданно высоким затратам и снижению качества изделия [1]. Выход из сложившейся ситуации видится в изменении традиционных форм ведения бизнеса, а именно – в переходе к передовым формам кооперации – виртуальным предприятиям (ВП).

В основе функционирования ВП лежит поиск наиболее выгодных и надежных партнеров для производства требуемых изделий. Оптимальное распределение заказов на выполнение работ (проектирование и изготовление нестандартного оборудования, средств технологического оснащения, проектирование различных видов технологических процессов, выполнение определенных технологических операций и др.) зависит не только от выбора исполнителей, но и от того, как сформированы пакеты заказов на выполнение задач ТПП. Задачи анализа ресурсов и загрузки мощностей предприятия с точки зрения эффективности выполнения заказов, также как и задачи формирования пакетов заказов и выбора исполнителей, не имеют универсальных методов решения, поскольку должны в большей степени опираться на специфику рассматриваемой предметной области. В настоящее время пока не найдены методы решения подобных задач для различных работ ТПП. Целью исследования в данном случае является разработка методов оптимизации конфигурирования и распределения заказов при решении задач ТПП в среде ВП на основе использования:

- мультиагентных технологий,
- генетического алгоритма.

Мультиагентные технологии характеризуются множественными решениями, применяются для динамического управления сетевыми ресурсами. Известно, что мультиагентные технологии обладают рядом преимуществ по сравнению с другими способами организации распределенных вычислений. Несмотря на это, разрабатывать методологию на основе использования исключительно этого подхода не представляется возможным. Одна из основных проблем заключается в том, что для сложных изделий число возможных вариантов комплектования заказов на ТПП может быть достаточно велико. При этом оптимальное распределение заказов зависит как от выбора варианта комплектования, так и от выбора исполнителей.

Не секрет, что сложные задачи проще решаются, если представить их в виде ряда более простых задач. Для решения проблемы оптимизации распределения заказов необходимо найти ответы на следующие вопросы.

1. Каким образом потенциальные исполнители узнают о поступившем заказе?
2. Как определить потенциальных исполнителей заказов?
3. Каким образом сформировать пакеты заказов?
4. На основании каких критериев оптимизации необходимо сделать выбор в пользу тех или иных исполнителей?

Функционирование ВП подразумевает работу всех участников в едином информационном пространстве и наличие открытой информационной среды, где каждый сотрудник может свободно пользоваться необходимой информацией о реализуемых проектах и потенциальных исполнителях. В качестве базового инструмента построения единого информационного пространства была выбрана PDM SmartTeam. Здесь не приводится сравнительная характеристика систем класса PDM, поскольку это не является основной задачей данного исследования.

Итак, после того, как головная компания получает заказ, она должна каким-то образом оповестить всех участников ВП. Для решения поставленной задачи предлагается использовать средства Internet. С целью обеспечения информационной целостности представления данных необходимо в PDM-системе реализовать процесс регистрации потенциальных подрядчиков. Помимо этого, требуется определить, какого рода информацию головная компания (или, как ее еще называют в современной литературе, компания-медиатор или OEM-предприятие) должна передать потенциальным исполнителям. Если поместить заказ в первоначальном виде, то может уйти достаточно большое количество времени на анализ всех поступивших предложений от участников и выбор оптимального исполнителя. Предлагается использовать другой подход [2]. Допустим, компания-медиатор получает заказ на изготовление определенного изделия. Тогда на каждый элемент этого изделия специалистами создается ресурсо-независимый технологический процесс (РНТП), который, в свою очередь, детализируется (любая операция РНТП может содержать одну или более заявку на выполнение работ по ТПП на другом предприятии). После этого информация о заказах в таком виде размещается на сайте компании. Идентификационные параметры заказа фиксируются в паспорте «Заказа». Кроме этого, всем участникам, которые способны выполнить заявку, рассылаются уведомления о поступлении нового заказа. Этому предшествует анализ возможностей потенциальных исполнителей на основе той информации, которую участники представили в своем профиле открытой информационной среды. Это, прежде всего, те виды работ, которые способны выполнить предприятия. Очевидно, что число таких участников может быть огромным, поэтому для экономии времени предполагается, что анализ информации об участнике будет выполнять агент в виде программного модуля PDM-системы. В связи с этим требуется предопределить атрибут кода работ в соответствующих классах PDM-системы. Таким образом, получается, что после того, как на предприятии определен пакет заказов на ТПП для субподрядчиков, этот факт фиксируется в PDM-системе как некоторое событие S_1 , приводящее агентов класса А в активное состояние. То есть выходит, что каждый заказ обрабатывается своим агентом. Каждый агент на основании анализа видов выполняемых предприятиями работ рассылает им уведомления. Общая схема формирования заказа представлена на рис. 1.

Другой вопрос, что сформировать РНТП можно различными способами, чем и объясняется многовариантность ТПП. В распределенной производственной среде проектировщик создает универсальный технологический процесс, поскольку не обладает информацией о конкретных доступных возможностях, ресурсах и мощностях потенциальных исполнителей.

Изучив информацию на сайте компании-медиатора, участники предоставляют сведения, на основании которых они готовы выполнить заказ (рис. 2). Это, прежде всего, стоимость выполнения заказа, сроки реализации, используемое оборудование и его текущая загрузка. При истечении определенного промежутка времени агент собирает всю информацию по откликнувшимся исполнителям и составляет сводную ведомость.

Здесь уже возникает следующий вопрос: каким образом разместить заказ, чтобы получить максимальную прибыль? Для этого требуется оценить каждого производителя, опираясь на полученные в ответ предложения об участии. Таких предложений может быть достаточно много, при этом вариантов комплектования – еще больше. Для

сложных изделий число задач ТПП может составлять несколько тысяч. Однако даже в случае несколько сотен задач очевидно, что число возможных вариантов пакетов заказов настолько велико, что задача оптимизации не может быть решена путем простого перебора всех этих вариантов.

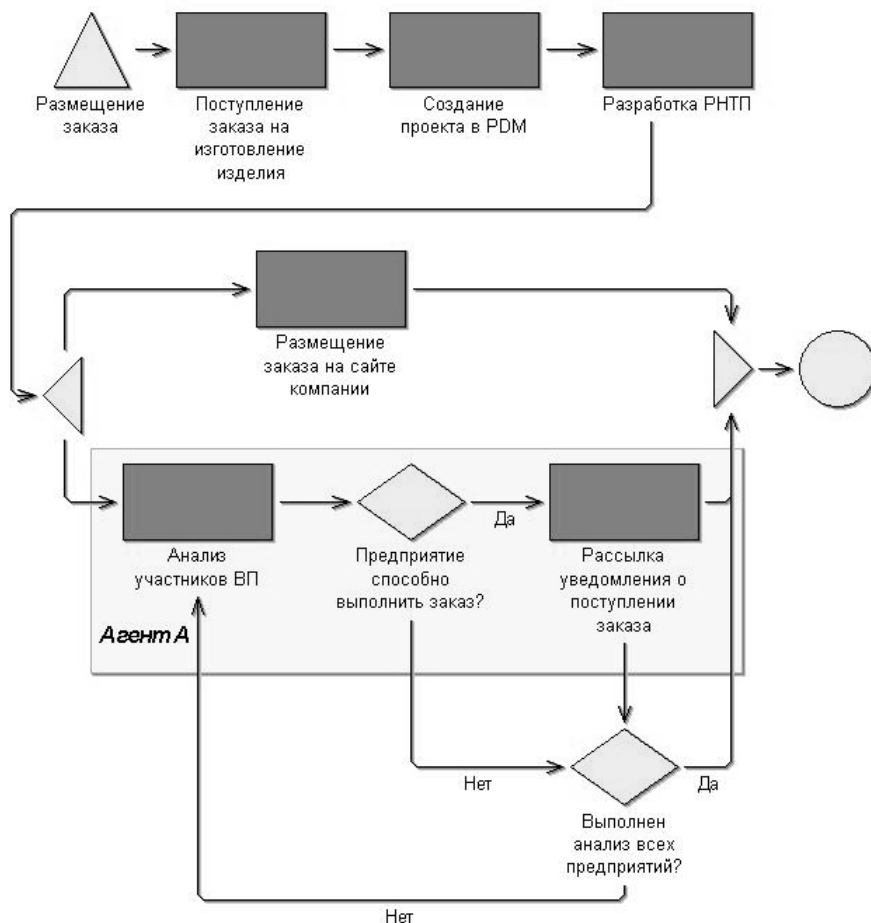


Рис. 1. Формирование заказа

Рис. 2. Атрибуты класса «Предложение»

Как показывает анализ, наибольший экономический эффект при оптимизации распределения заказов достигается за счет использования так называемых генетических алгоритмов. Генетический алгоритм (ГА) – это адаптивный метод функциональной оптимизации, основанный на компьютерном моделировании. Исследование алгоритмов распределения и оптимизации заказов при решении задач ТПП нашло свое отражение в работе [3], в которой представлена методика решения «задачи коммивояжера» с использованием генетических алгоритмов.

ГА представляет собой метод оптимизации, основанный на концепциях естественного отбора и генетики. В этом подходе переменные, характеризующие решение, представлены в виде ген и хромосом. Предлагается на основе анализа поступивших предложений сформировать различные варианты размещения заказа или, используя терминологию ГА, образовать популяцию особей. Затем оценить каждый вариант, т.е. рассчитать целевую функцию особей и популяции в целом. В качестве критериев оценки можно использовать время (T_{Aiucn}) и стоимость (C_{Aiucn}) выполнения заказа, которые предлагают потенциальные исполнители. Также в качестве критериев оценки могут быть приняты во внимание наличие опыта с данным производителем, используемые технологии и оборудование, наличие склада и др. Для упрощения представления целевой функции в качестве критериев оценки будем использовать только время и стоимость выполнения заказа. В результате функция будет выглядеть следующим образом:

$$F_{A,i} = \omega_1 \times \frac{C_{Aizak}}{C_{Aiucn}} + \omega_2 \times \frac{T_{Aizak}}{T_{Aiucn}},$$

где ω_1, ω_2 – весовые коэффициенты ($\omega_1 + \omega_2 = 1$); C_{Aizak}, T_{Aizak} – соответственно, стоимость и время выполнения работ, оптимальные для заказчика.

После расчета целевой функции всех особей происходит их упорядочение по мере уменьшения значений этой функции. Затем выполняется выборка из популяции пары особей с наибольшими значениями целевой функции и составление из них комбинаций. Далее происходит опять расчет целевой функции и упорядочение вариантов. Конечно, чтобы избежать заикливания или достаточно длительных расчетов, предлагается либо определить механизм выхода из цикла, либо задать конечное число итераций, либо же максимальное время, которое будет отведено на выполнение расчетов.

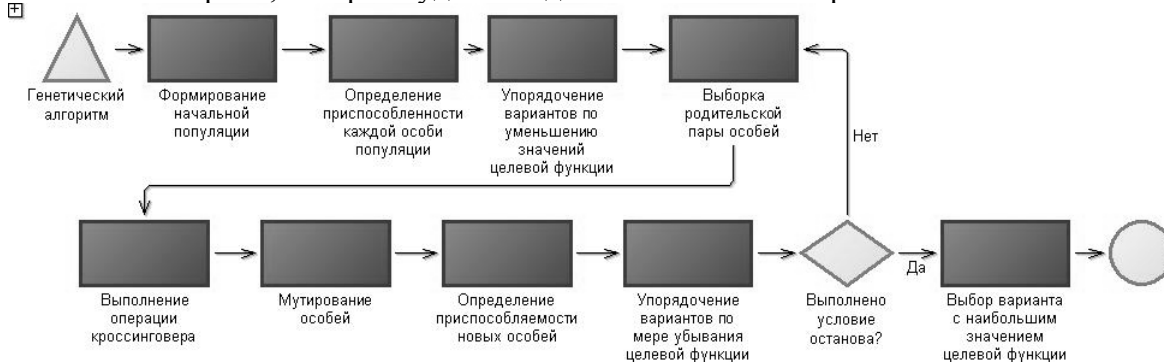


Рис. 3. Схема работы генетического алгоритма

Общая схема работы генетического алгоритма (рис. 3) выглядит следующим образом.

- (1) Формируется начальная популяция, т.е. массив решений некоторым образом заполняется конкретными решениями (особями). В частности, решения могут генерироваться случайным образом.
- (2) Вычисляется приспособленность каждой особи популяции по ее признакам с использованием критериев. Чем лучше значение целевой функции обеспечивает особь, тем выше ее приспособленность.

- (3) Особи ранжируются по критерию приспособленности. Из особей с наиболее высоким уровнем приспособленности отбирается определенное число родительских пар (минимум, одна пара).
- (4) Выполняется операция кроссинговера, которая состоит в случайном выборе точки на хромосоме, относительно которой будет производиться скрещивание. При скрещивании часть новой особи до этой точки формируется за счет одного родителя, а часть после этой точки – за счет другого родителя.
- (5) Новые особи с определенной вероятностью мутируют, т.е. происходит изменение на уровне одного из генов. Вероятность мутации обычно полагают равной порядка 1%.
- (6) Вычисляется приспособляемость новых особей, после чего эти особи заменяют в популяции предыдущие наименее приспособленные особи. Очевидно, что если приспособляемость какой-либо новой особи ниже, чем приспособляемость старой, то это особь не попадает в популяцию.
- (7) Если самое лучшее решение в популяции не признается удовлетворительным, то осуществляется переход к шагу 3 описанного алгоритма. Если же решение признано удовлетворительным, то алгоритм заканчивает работу.

После достижения критерия останова в качестве оптимального решения отбирается вариант, обладающий лучшим сочетанием интересующих нас качеств [4]. Далее иницируется работа следующего агента, функциями которого является формирование для заказчика «сводной ведомости», содержащей информацию по всем субподрядчикам, определенным для выполнения работ по ТПП данного изделия.

Таким образом, предполагается, что конфигурирование ВП должно осуществляться при помощи использования агентов и генетического алгоритма. На начальном этапе, когда нам требуется собрать сведения о потенциальных исполнителях и разослать уведомления, работу выполняет программный агент. Далее сам анализ происходит на основе использования генетического алгоритма. После выбора конечного производителя составляется сводная ведомость всех участников. В настоящее время проводятся практические исследования применимости этой методики.

Литература

1. Зильбербург Л.И., Молочник В.И., Яблочников Е.И. Реинжиниринг и автоматизация технологической подготовки производства в машиностроении. – СПб: Политехника, 2004. – 152 с.
2. Фомина Ю.Н. Исследование алгоритмов оптимизации конфигурирования и распределения заказов при решении задач ТПП в среде виртуального предприятия // Научно-технический вестник СПбГУ ИТМО. – 2007. – Т. 28.
3. Фомина Ю.Н., Яблочников Е.И. Методы распределения заказов на выполнение ТПП в среде виртуального предприятия // Материалы конференции «Фундаментальные и прикладные проблемы надежности и диагностики машин и механизмов» ИПМАШ РАН. – СПб, 2007.
4. Романовский И.В. Алгоритмы решения экстремальных задач – М.: Наука, 1977. – 352 с.

РАЗРАБОТКА УЧЕБНО-МЕТОДИЧЕСКОГО ВЗАИМОДЕЙСТВИЯ УЧЕБНО-МЕТОДИЧЕСКОГО КОМПЛЕКСА С INTERNET

Ю.Е. Котельникова

Научный руководитель – к.т.н., доцент Б.С. Падун

Работа ведется в рамках инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» при создании образовательного модуля «Поиск научных и технических решений». Статья посвящена схеме автоматизированного взаимодействия учебно-методического комплекса с Internet в целях обеспечения адаптивности и эволюционности.

Введение

Образовательный модуль – часть образовательной программы подготовки специалистов по соответствующему направлению подготовки высшего профессионального образования, имеющая определенную логическую завершенность и целостность содержания, технологии обучения, организации и оценивания по отношению к установленным целям и результатам образования модуля. Учебно-методический комплекс образовательного модуля регламентирует цели, ожидаемые результаты, содержание, обеспечивает учебно-методическое сопровождение подготовки обучающихся по модулю.



Рис. 1. Взаимосвязь учебно-методического комплекса с Internet

Учебно-методический комплекс должен обладать свойствами адаптивности и эволюционности, значит, необходимо обеспечить возможность его постоянного обновления. Один из вариантов решения проблемы – контакт учебно-методического комплекса с Internet, откуда можно будет принимать новую информацию. На данном этапе большая роль в этом процессе отводится человеку, ручной работе. Однако это может нарушить целостность работы электронной составляющей учебно-методического комплекса, поэтому требуется автоматизация. В данной работе предложена схема взаимодействия учебно-методического комплекса с Internet и проанализированы методы автоматического анализа текстовых данных (рис. 1).

Проблема обработки текстовых данных

Процесс автоматизации должен коснуться как отправки запросов в Internet, так и обработки полученной информации. Точкой взаимодействия учебно-методического комплекса нам видится электронный глоссарий – электронное издание, содержащее информацию по терминам, базовым понятиям, положениям и т.п., которые связаны с содержанием изучаемого модуля, учебно-методическим пособием и необходимы для его освоения. Из глоссария в Internet посылается поисковый запрос, ответом на который является блок информации, т.е. неструктурированный текст.

Проблема состоит в том, что текстовые документы невозможно преобразовать в табличное представление без потери семантики текста и отношений между сущностями. Поэтому такие документы хранятся в базе данных без преобразования, как текстовые поля (BLOB-поля). Дальнейшей автоматизированной обработке данные поля не подлежат, что возвращает нас к проблеме целостности электронной составляющей учебно-методического комплекта и наводит на мысль о поиске методов анализа неструктурированного текста.

Схема обработки неструктурированного текста

Методы анализа в неструктурированных текстах лежат на стыке нескольких областей: DataMining, обработка естественных языков, поиск информации, извлечение информации и управление данными. В целом процесс работы с неструктурированным текстом, полученным в результате запроса из глоссария, можно представить в виде схемы (рис. 2).

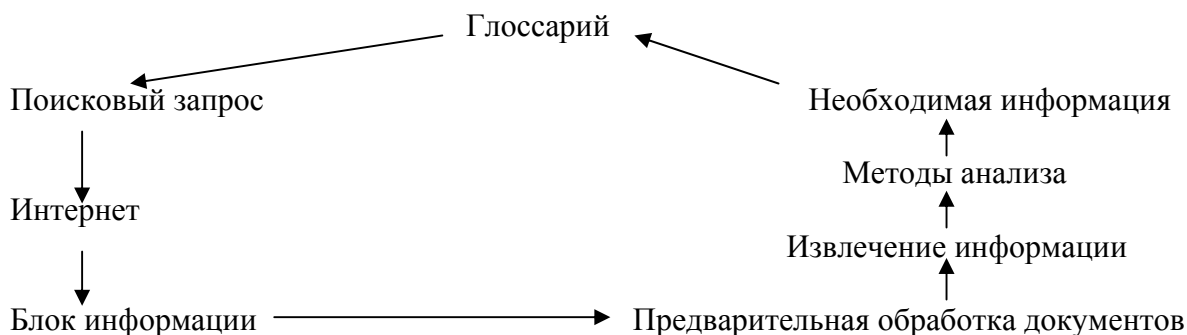


Рис. 2. Схема обработки неструктурированного текста

Методы предварительной обработки документов

Информация, получаемая в виде неструктурированного текста, требует предварительной обработки с целью придания тексту большей строгости. Для этого существуют следующие методы:

1. удаление стоп-слов (слова, которые являются вспомогательными и несут мало информации о содержании документа);
2. стэмминг – морфологический поиск, который заключается в преобразовании каждого слова к его нормальной форме;
3. N-граммы – альтернатива морфологическому разбору и удалению стоп-слов. N-грамма – это часть строки, состоящей из N символов. N-граммы не требуют лингвистического представления слов, что делает данный прием более независимым от языка;
4. приведение регистра.

Методы выделения ключевых понятий

В предварительно обработанном тексте необходимо выделить ключевые понятия, которые могут идентифицировать текст и над которыми в дальнейшем будет выполняться анализ. Для этого используют:

1. определение частых наборов слов и объединение их в ключевые понятия;
2. идентификация фактов в текстах и извлечение их характеристик. Фактами являются некоторые события или отношения;
3. локальный анализ с помощью шаблонов (проходит в 2 стадии) – из текстовых документов извлекаются отдельные факты с помощью лексического анализа; анализ понятий – интеграция извлеченных фактов и/или вывод новых фактов.

Методы TextMining

Дальнейшая работа строится на использовании методов TextMining, среди которых, на наш взгляд, главную роль должны сыграть классификация и кластеризация.

1. Классификация – стандартная задача из области DataMining. Определение для каждого документа одной или нескольких заранее заданных категорий. Частным случаем задачи классификации является задача определения тематики документа.
2. Кластеризация – автоматическое выявление групп семантически похожих документов среди заранее фиксированного множества. Группы формируются только на основе попарной схожести описаний документов.

Большинство методов классификации текстов, так или иначе, основаны на предположении, что документы, относящиеся к одной категории, содержат одинаковые признаки (слова или словосочетания), и наличие или отсутствие таких признаков в документе говорит о его принадлежности или непринадлежности к той или иной теме. Таким образом, для каждой категории должно быть множество признаков. Такое множество признаков часто называют словарем, так как оно состоит из лексем, которые включают слова и/или словосочетания, характеризующие категорию.

Задача методов классификации состоит в том, чтобы наилучшим образом выбрать такие признаки и сформулировать правила, на основе которых будет приниматься решение об отнесении документов к рубрике. Существует 2 противоположных подхода к формированию множества признаков и построению правил:

1. машинное обучение – предполагается наличие обучающей выборки документов, по которому строится множество признаков;
2. экспертный метод – предполагает, что выделение признаков и составление правил производится экспертами.

Возможна и комбинация описанных подходов, когда выделение признаков и составление правил выполняется автоматически на основе обучающей выборки, и в то же время правило строится в таком виде, чтобы эксперту была понятна логика автоматической рубрикации и у него была возможность вручную корректировать эти правила.

Большинство алгоритмов кластеризации требуют, чтобы данные были представлены в виде модели векторного пространства. Это наиболее широко используемая модель для информационного поиска. В этой модели каждый документ представляется в многомерном пространстве, в котором каждое измерение соответствует слову в наборе документов.

Все алгоритмы кластеризации основываются на изменениях схожести по различным критериям. Выделяют 3 основных типа методов кластеризации документов.

- 1) Иерархический – создает дерево со всеми документами в корневом узле и одним документом в узле-листе. Промежуточные узлы содержат различные документы, которые становятся более и более специализированными по мере приближения к листьям дерева. Этот метод полезен, когда исследуют новую коллекцию документов и хотят получить общее представление о ней.
- 2) Агломеративные – кластеризация выполняется, начиная с индивидуальных элементов, группируя их в кластеры (снизу вверх): односвязный метод группирует ближайших членов; полносвязный – дальних членов; среднесвязный – ближайших к середине членов. Результат – дендограмма.

3) Дивизимные – кластеризация выполняется, начиная с одного кластера и разбивая его на несколько (сверху вниз):

- алгоритм дивизимного разделения по главному направлению строит бинарное дерево, в котором каждый узел содержит документы, далее рекурсивно делит каждый лист на 2 дочерних узла, пока сохраняется критерий деления;

- бинарный алгоритм обеспечивает группировку и просмотр документальных кластеров по ссылкам подобия. В один кластер помещаются самые близкие по своим свойствам документы. В процессе кластеризации строится базис ссылок от документа к документу, основанный на весах и совместном употреблении определяемых ключевых слов;

- интерактивная кластеризация обычно создает кластеры, оптимизируя целевую функцию, описанную локально (среди документов одного и того же кластера) или глобально (через все документы). Типичным представителем интерактивных алгоритмов является алгоритм k-средних. Он интерактивно выполняет деление данных на k-кластеров, минимизируя расстояние между элементами кластеров и их центрами;

- нечеткий алгоритм включает каждый документ во все кластеры, но при этом связывает с ним весовую функцию, определяющую степень принадлежности данного документа определенному кластеру.

Многие исследователи все еще придерживаются мнения, что более эффективна обработка текста вручную, а не автоматизированная.

Итак, теперь мы можем определить, насколько найденный текст соответствует изначальному запросу. Обработанная таким образом информация должна быть интерпретирована на естественном языке и занесена в глоссарий.

Заключение

Учебно-методический комплекс должен обладать свойствами адаптивности и эволюционности, значит, необходимо обеспечить возможность его постоянного обновления. Как вариант, надо организовать взаимодействие учебно-методического комплекса с Internet, откуда можно будет принимать новую информацию. На данном этапе большая роль в этом процессе отводится человеку, ручной работе. Однако это может нарушить целостность работы электронной составляющей учебно-методического комплекса, поэтому требуется автоматизация. В данной работе предложена схема взаимодействия учебно-методического комплекса с Internet и проанализированы методы автоматического анализа текстовых данных.

Литература

1. Баргесян А.А. Технологии анализа данных: DataMining, VisualMining, TextMining, OLAP / А.А. Баргесян, М.С. Куприянова, В.В. Степаненко, И.И. Холод. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2008. – 384 с.
2. Зайнутдинова Л.Х. Создание и применение электронных учебников (на примере общетехнических дисциплин): Монография. – Астрахань: Изд-во «ЦНТЭП», 1999.

РАСЧЕТ ИЗГИБНЫХ И КОНТАКТНЫХ НАПРЯЖЕНИЙ ЗУБЧАТЫХ КОЛЕС С НЕСИММЕТРИЧНЫМ ПРОФИЛЕМ ЗУБА МЕТОДОМ КОНЕЧНЫХ ЭЛЕМЕНТОВ

А.В. Антонов, Г.А. Гладышев

Научный руководитель – д.т.н., профессор Б.П. Тимофеев

Основным методом расчета зубчатых передач на контактную прочность является метод, приведенный в ГОСТ 21354-87. В работе методом конечных элементов произведен расчет действующих изгибных и контактных напряжений для зубчатых передач с несимметричным профилем зуба в зависимости от параметров этих передач, таких как угол профиля зуба рейки α и угол наклона линии зуба β . Также проведено сравнение результатов по контактным напряжениям, полученных аналитическим методом по ГОСТ 21354-87 с результатами, полученными методом конечных элементов.

Введение

Использование эвольвентных зубчатых колес с несимметричным зубом позволяет повысить несущую способность зубчатых передач по контактной и изгибной прочности в основном (рабочем) направлении вращения [1, 2]. Применение таких колес выгодно в реверсивных передачах, в которых крутящий момент в основном направлении гораздо более продолжителен по времени, чем в обратном.

В работе начато исследование контактной и изгибной прочности зубчатых колес с несимметричным профилем зуба численным методом. Результаты, полученные при использовании численного метода по контактной прочности, были сравнены и проанализированы с результатами, полученными при использовании аналитического метода.

Расчет аналитическим методом производился на основе существующего ГОСТ 21354-87 «Передачи зубчатые цилиндрические эвольвентные внешнего зацепления. Расчет на прочность» [3].

В основе численного метода расчета напряжений лежит метод конечных элементов (МКЭ). Метод конечных элементов в последние десятилетия получил очень широкое распространение и стал одним из основных методов расчета конструкций. Это обусловлено универсальностью подхода, лежащего в основе МКЭ, заключающегося в представлении геометрии любого деформируемого тела в виде совокупности элементов простейшей формы: треугольной, четырехугольной и др. Элементы бывают одномерными, плоскими и пространственными, с прямолинейными или криволинейными сторонами. Вдоль каждой из них может быть два или более узлов, что обуславливает необходимую степень точности расчетов [4, 5].

Методика расчета

Для расчета контактных и изгибных напряжений методом конечных элементов был использован программный пакет CosmosWorks системы SolidWorks 2007. Данный программный продукт позволяет рассчитывать действующие напряжения в трехмерной модели исследуемого объекта.

Для получения точного эвольвентного профиля зубчатые колеса генерировались с помощью программы SHAFT 2D в системе КОМПАС-3D V9. Обмен с системой SolidWorks 2007 осуществлен через стандартный формат обмена графической информацией IGES. Далее в системе SolidWorks 2007 с использованием импортированных зубчатых колес с эвольвентным профилем зубьев создавались модели исследуемых передач. Для упрощения процесса вычисления напряжений вследствие ограниченности ресурсов используемого ПК исследуемые модели рассматривались как модели зацепления конических зубьев.

Для того чтобы CosmosWorks смог рассчитать действующие напряжения, необходимо задать исходные данные. Так, в прочностном анализе учитываются модуль упругости (модуль Юнга), коэффициент теплового расширения при заданной температуре, коэффициент Пуассона, плотность, коэффициент трения, модуль сдвига, коэффициент внутреннего трения. Необходимые параметры материалов содержатся в соответствующих библиотеках. Свойства могут быть постоянными, нелинейными или зависимыми от температуры. Списки существующих материалов в базе данных могут быть дополнены новыми материалами.

Далее необходимо правильно задать так называемую среду нагружения. Она состоит из нагрузок и ограничений. Для статического структурного анализа доступны следующие виды нагрузок: сила (к поверхности, к линии, к точке), момент (к поверхности, к линии, к точке), давление (к поверхности, к линии, к точке) и «гравитация», т.е. прикладывание к телу силы тяжести. Возможные ограничения: фиксация, ограничение перемещения вдоль каких-либо осей и поворот вокруг них, смещение и так называемая «симметрия». Последнее ограничение запрещает перемещение по нормали к поверхности либо говорит о том, что представленная геометрическая модель является симметричной половиной некоего объекта. Например, применение симметричного ограничения к цилиндрическому отверстию фиксирует ось этого отверстия и не допускает деформацию отверстия, т.е. разрешает только вращение вокруг собственной оси.

Далее необходимо определить набор соприкасающихся поверхностей для установления глобального контакта моделей и четкого определения грани источника и грани цели.

Определив все параметры, необходимо разбить модель на конечные элементы путем создания сеточной модели – совокупности узлов и элементов. Конечно-элементная сетка является основной составляющей расчета методом конечных элементов, так как именно в узлах этой сетки и определяются действующие напряжения.

На подготовительном этапе важно так сформулировать задачу анализа, чтобы, с одной стороны, получить правильное решение, а с другой – чтобы не потребовалось много ресурсов и времени. Поэтому сначала можно попытаться создать крупную сетку, так как в этом случае преимущество заключается в том, что потребуется меньше времени для решения задачи. Однако работа с крупной сеткой может привести к потере значимых физических явлений. В данном исследовании улучшение произведено рядом с точкой контакта. При улучшении используются различные алгоритмы сглаживания сетки, изменение атрибутов узлов и элементов, измельчения и улучшения формы элементов и др. Программа может выполнять контроль соответствия элементов расчетной и сеточной моделей, вводить, удалять и изменять положение конечных элементов, обеспечивая тем самым формирование сетки высокого качества.

Результаты расчета

Расчет действующих контактных и изгибных напряжений был произведен на примере зубчатой передачи верхнего силового привода буровой установки. Как уже говорилось ранее, с помощью среды твердотельного моделирования КОМПАС-3D V9 были смоделированы зубчатые передачи – прямозубая и косозубая:

- $m=7, z_1=13, z_2=22, \alpha=20^\circ, \alpha=20^\circ, \beta=0;$
- $m=7, z_1=13, z_2=22, \alpha=20^\circ, \alpha=20^\circ, \beta=5.$

Далее были проварьированы:

- в прямозубой передаче – угол α от 15° до 35° ;
- в косозубой передаче – угол α от 15° до 35° , угол β от 5° до 35° .

В процессе расчета изменялись:

- прилагаемый крутящий момент;

- положение линии контакта на рабочей поверхности зуба;
- распределение нагрузки вдоль контактной линии.

Результаты, полученные в процессе расчета в пакете программ SolidWorks 2007, сведены в соответствующие графические зависимости: зависимость контактных (рис. 1) и изгибных (рис. 2) напряжений от угла α для прямозубых колес, а также зависимость контактных (рис. 3) и изгибных (рис. 4) напряжений от угла α и угла β для косозубых колес.

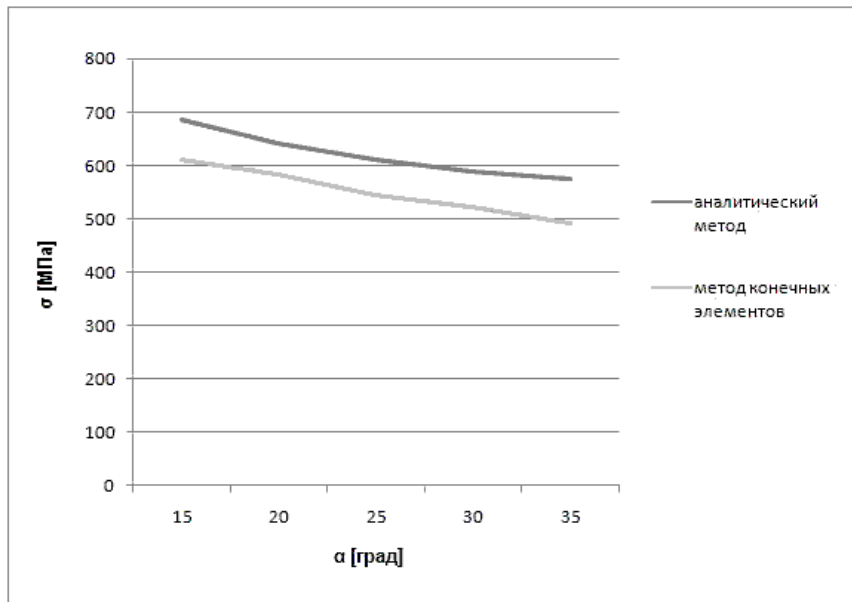


Рис. 1. График зависимости контактных напряжений от угла профиля зуба рейки α

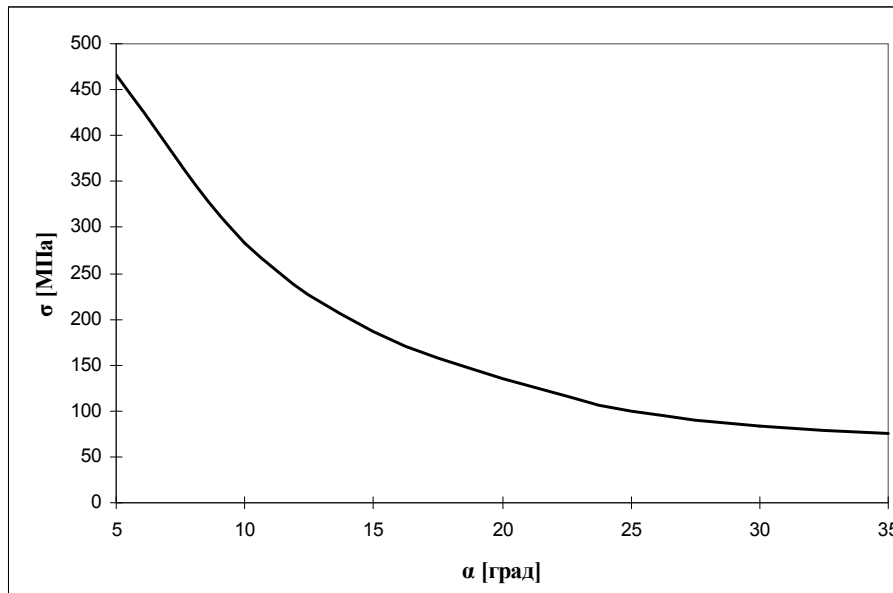


Рис. 2. График зависимости изгибных напряжений от угла профиля зуба рейки α

Полученные результаты показывают, что увеличение угла профиля зуба рейки α ведет к увеличению как контактной, так и изгибной прочности, т.е. к уменьшению действующих контактных и изгибных напряжений.

Полученные результаты показывают очевидную зависимость изгибных напряжений от угла наклона линии зуба β , т.е. при увеличении угла β действующие контактные и изгибные напряжения уменьшаются. Так же результаты показывают, что при разных

углах β минимальные контактные и изгибные напряжения действуют при различных углах α .

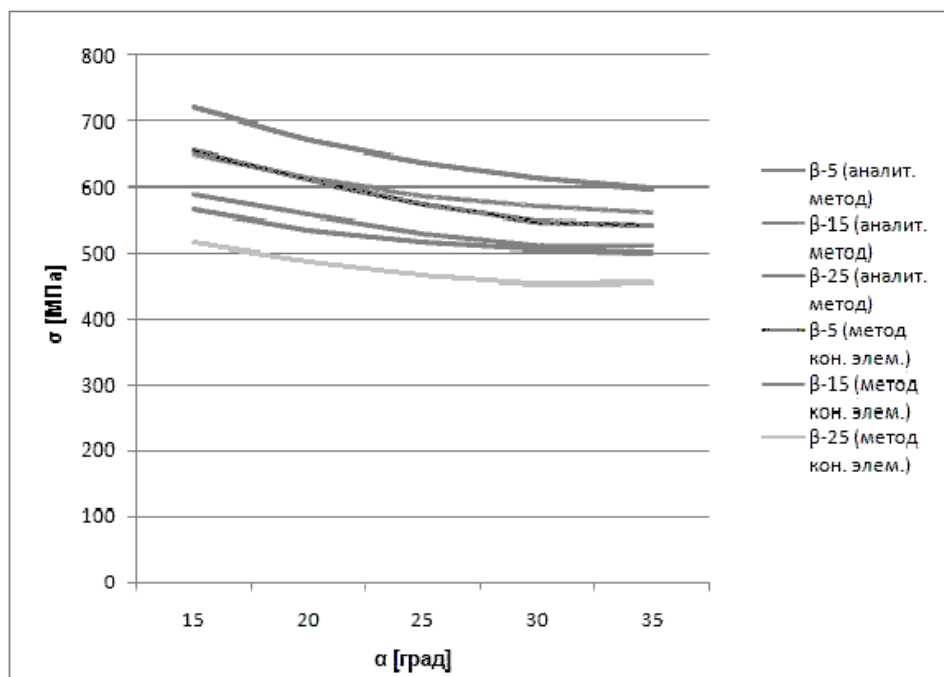


Рис. 3. График зависимости контактных напряжений от угла профиля зуба рейки α и угла наклона линии зубьев β

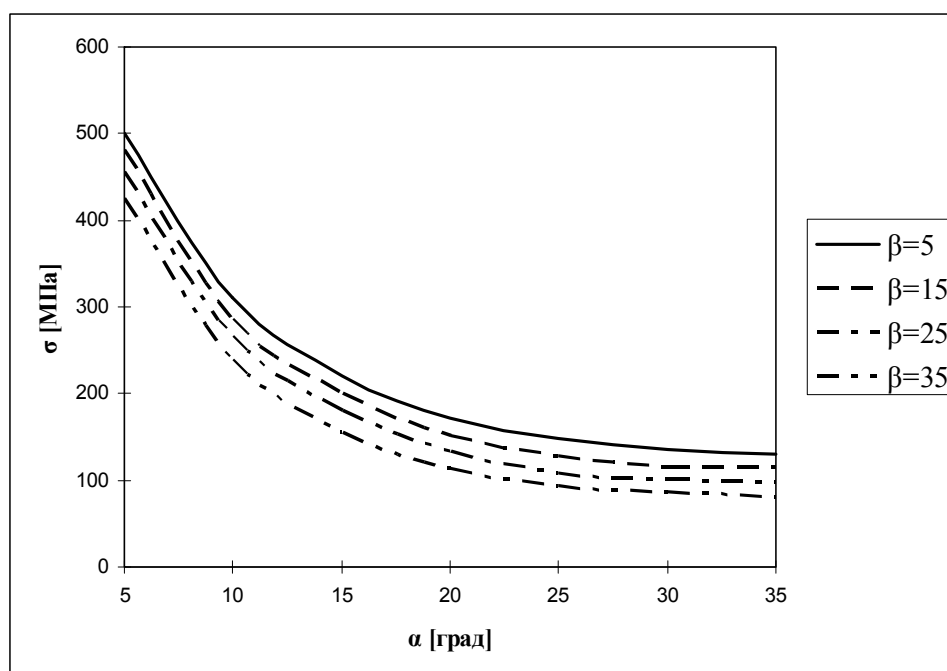


Рис. 4. График зависимости изгибных напряжений от угла профиля зуба рейки α и угла наклона линии зубьев β

На следующих графиках представлены: зависимость контактных (рис. 5) и изгибных (рис. 6) напряжений в косозубой передаче от положения контактной линии на рабочей по-

верхности зуба, а также зависимость контактных (рис. 7) и изгибных (рис. 8) напряжений от неравномерного распределения нагрузки вдоль линии контакта.

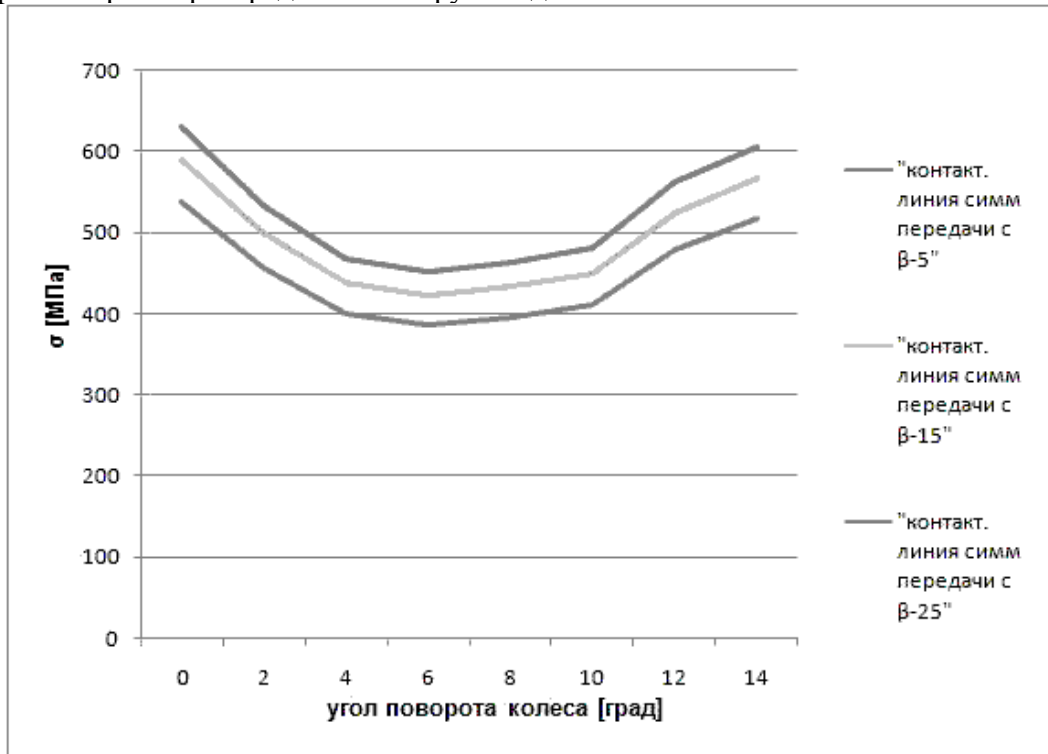


Рис. 5. График зависимости контактных напряжений от угла поворота колеса

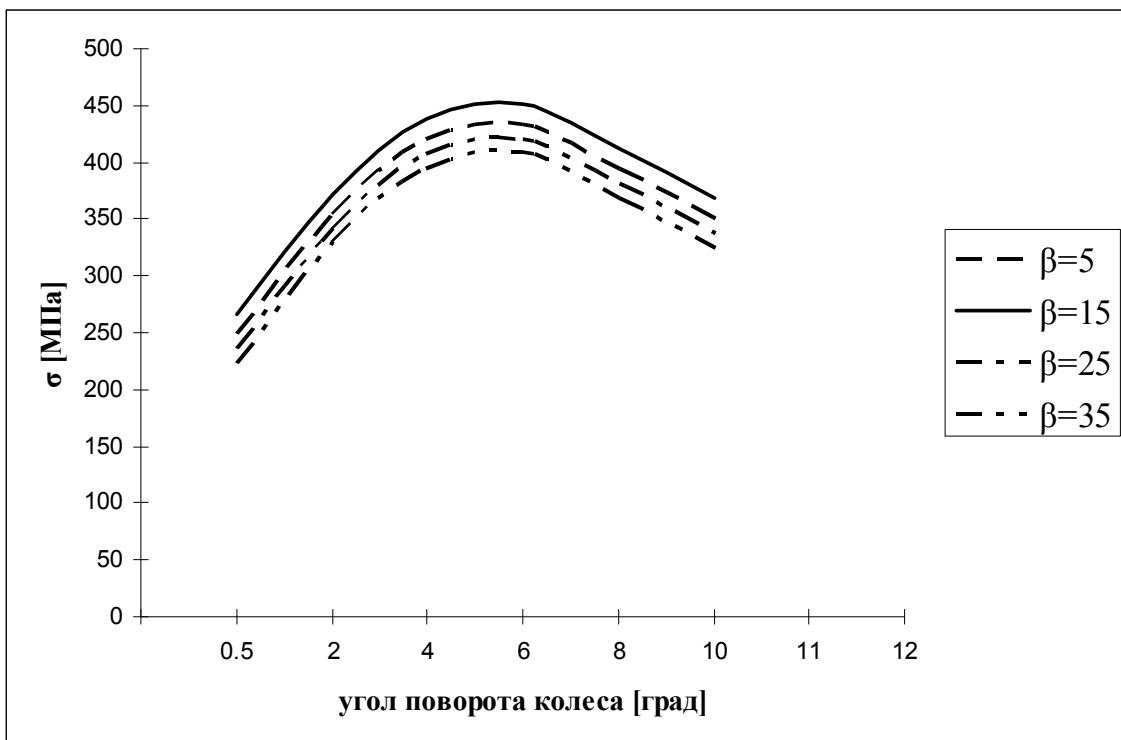


Рис. 6. График зависимости изгибных напряжений от угла поворота колеса

Полученные результаты показывают, что при углах, обозначающих начало и конец процесса зацепления в косозубой передаче, наблюдается увеличение контактных и уменьшение изгибных напряжений, связанное с тем, что в этот момент в зацепление

входит следующий зуб колеса со следующим зубом шестерни. График представлен для косозубых передач с несимметричными профилями зубьев ($\alpha=20^\circ$, $\alpha=30^\circ$). При увеличении угла α от 5° до 35° как контактные, так и изгибные напряжения соответственно уменьшаются, но характер зависимости напряжений от фазы угла зацепления сохраняется.

Вариантов распределения нагрузки вдоль контактной линии существует бесконечное множество. В данном расчете рассматривался вариант с параболической зависимостью $x = ky$ (где $k = 1, 2, 4$) с расположением начала координат на краю рабочей поверхности у одной из торцевых граней зуба таким образом, чтобы рабочей поверхности возле противоположной грани сообщалась максимальная нагрузка.

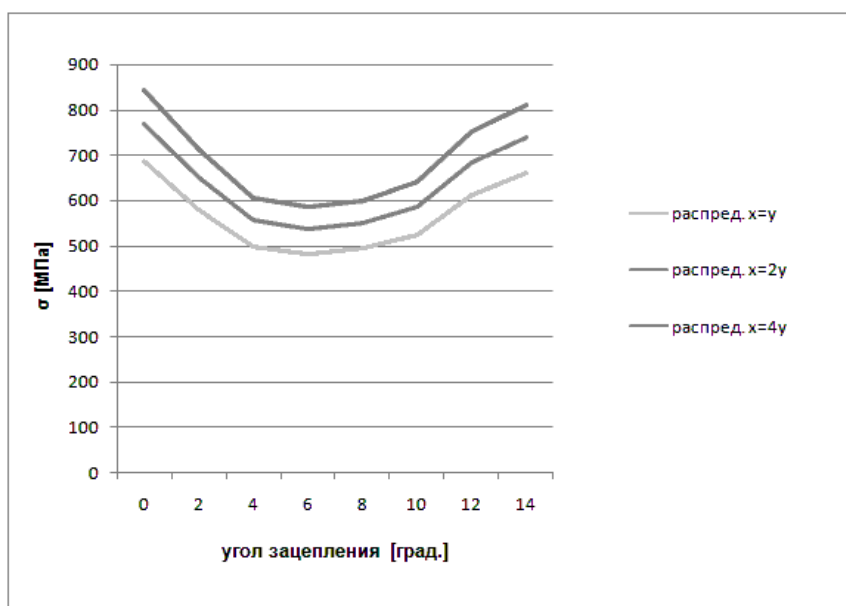


Рис. 7. Зависимость контактных напряжений при неравномерном распределении нагрузки вдоль контактной линии от угла зацепления

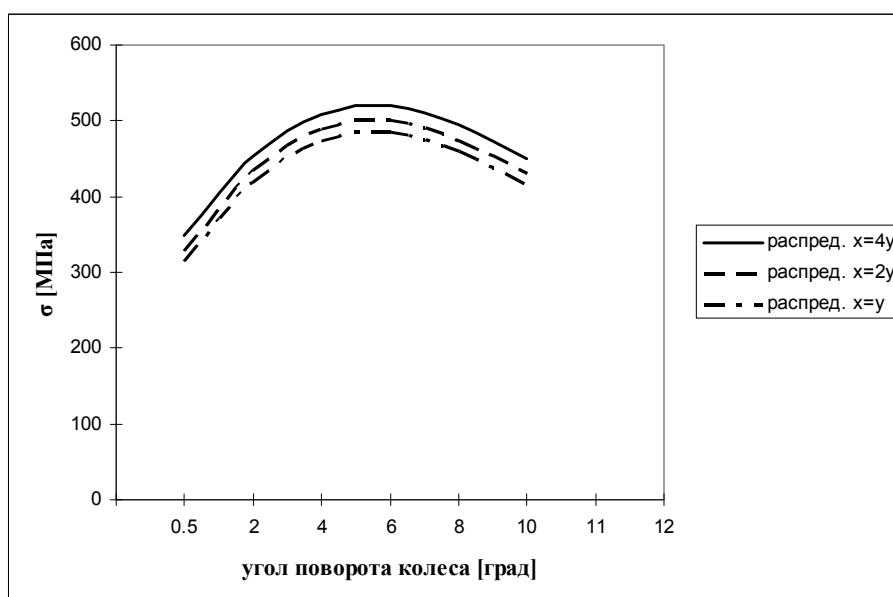


Рис. 8. Зависимость изгибных напряжений при неравномерном распределении нагрузки вдоль контактной линии от угла поворота колеса

По результатам исследования можно сделать вывод, что при неравномерном распределении нагрузки характер распределения изгибных напряжений по фазам зацепления не изменился, но при углах, соотносящихся с серединой фазы зацепления, наблюдаются резкие возрастания изгибных и уменьшение контактных напряжений, пропорциональных коэффициенту k уравнения неравномерной нагрузки.

В качестве итога проведенного расчета изгибных напряжений на рис. 9 представлен график зависимости местоположения зоны с наибольшими изгибными напряжениями в подошве зуба от угла поворота колеса на примере косозубой передачи с симметричным профилем.

В зависимости от угла поворота колеса зона максимальных изгибных напряжений в подошве зуба косозубой передачи перемещается от одного торца зубчатого венца к другому, что позволяет сделать выводы о месте зарождения и дальнейшего развития усталостной трещины, приводящей к поломке зуба.



Рис. 9. График зависимости местоположения зоны с максимальными изгибными напряжениями от угла поворота колеса

Заключение

Результаты расчетов изгибных и контактных напряжений, полученные методом конечных элементов позволяют сделать следующие выводы.

- Метод конечных элементов подтвердил результаты, полученные аналитическим методом расчета зубчатых передач ГОСТ 21354-87.
- Увеличение угла профиля зуба α ведет к уменьшению контактных и изгибных напряжений в зубчатых передачах.
- Увеличение угла наклона линии зуба β также ведет к уменьшению контактных и изгибных напряжений. При этом для разных углов β минимальные контактные и изгибные напряжения действуют при различных углах α . Таким образом, можно подобрать такие α и β , при которых контактные и изгибные напряжения будут минимальными. Для каждой зубчатой передачи эти α и β разные.

Следовательно, если подобрать оптимальные параметры зубчатой передачи, такие как угол профиля зуба рейки α и угол наклона линии зубьев β , можно уменьшить действующие контактные и изгибные напряжения, а, следовательно, повысить долговечность зубчатой передачи.

- При равномерном распределении нагрузки вдоль линии контакта в косозубой передаче при углах, обозначающих начало и конец процесса зацепления, наблюдаются

рост контактных и спад изгибных напряжений, связанный с тем, что в этот момент в зацепление входит следующий зуб колеса со следующим зубом шестерни.

▪ При неравномерном распределении нагрузки вдоль линии контакта характер распределения изгибных напряжений по фазам зацепления аналогичен распределению напряжений при равномерной нагрузке, но при углах, соотносящихся с серединой фазы зацепления, наблюдаются резкие возрастания изгибных и уменьшение контактных напряжений, пропорциональных коэффициенту k уравнения неравномерной нагрузки.

▪ В зависимости от угла поворота колеса зона максимальных изгибных напряжений в подошве зуба косозубой передачи перемещается от одного торца зубчатого венца к другому, что позволяет сделать предположение о точках возможного зарождения усталостных трещин и об их кинетике, которые в последствии приводят к поломке зуба.

Литература

1. Вулгаков Э.Б., Капелевич А.Л. Возможности несимметричных зубчатых передач // Вестник машиностроения. – 1986. – №4. – С. 14–16.
2. Вулгаков Э.Б., Ривкин Г.В. Проектирование зубчатых колес с несимметричным профилем зубьев // Машиноведение. – 1976. – №5. – С. 35–39.
3. ГОСТ 21354-87 Передачи зубчатые цилиндрические эвольвентные внешнего зацепления. Расчёт на прочность.
4. Сабоннадьер Ж.-К., Кулон Ж.-Л. Метод конечных элементов и САПР. – М.: Мир, 1989. – 190 с.
5. Норри Д., де Фриз Ж. Введение в метод конечных элементов. – М.: Мир. 1981. – 304 с.

ТАБЛИЧНЫЙ ПРОЦЕССОР ДЛЯ РЕШЕНИЯ ТЕХНОЛОГИЧЕСКИХ ЗАДАЧ

В.А. Свистунов

Научный руководитель – д.т.н., профессор Д.Д. Куликов

В статье предложен способ хранения нормативно-справочной информации в базе данных. Рассмотрен теоретический подход и структура базы данных.

Введение

Существует ряд технологических задач, которые используют большой объем нормативно-справочной информации (НСИ), таких как расчет режимов резания, расчет операционных припусков, расчет норм времени и т.д. Автоматизация этих задач предполагает ввод НСИ в базу данных. Анализ справочных таблиц показывает, что их трудно привести к реляционному виду для ввода в реляционную БД ввиду их сложности и разнообразия. Выходом является использование системы ввода таблиц и принятия решения по ним. Для этих целей можно использовать экспертные системы, однако более эффективным является создание специализированной оболочки, ориентированной на обработку сложных таблиц и удобной для ввода таких таблиц в базу знаний.

Такая оболочка была разработана на кафедре ТПС и получила название «Табличный процессор». Однако эта программа позволяла вводить не любые таблицы, поэтому была поставлена задача усовершенствовать эту оболочку, попутно переведя ее на технологию «клиент-сервер».

Теоретический подход

Принятие решения может выполняться на основе преобразования таблиц к реляционному виду или к таблице соответствий (ТС). Преобразования к реляционному виду являются сложными и порождают весьма большие наборы данных. Использование ТС позволяет достаточно простым способом ввести информацию из справочных таблиц в базу знаний. Рассмотрим теоретические положения, связанные с ТС.

Таблицу соответствий T можно представить как отношение из множества входных параметров X и множество выходных параметров Y . Традиционная теория ТС предлагает следующую структуру T :

$$T = \langle X, Y, P, R, M, C \rangle,$$

где $P = \{P_1, P_2\}$ – входные условия для первого (P_1) и второго (P_2) входов; $R = \{R_1, R_2\}$ – правила выбора для первого (R_1) и второго (R_2) входов; M – матрица решений; C – кортеж решений.

Входные условия представляются набором отношений вида px_ga , где p – номер условия, x – входной параметр, g – отношение порядка, a – константа. Правила выбора представляются множеством номеров входных условий и имеют вид nN , где n – номер правила, $N = \{n_1, n_2, \dots\}$ – множество номеров условий. Каждое правило из набора R_1 выбирает строку таблицы, а из набора R_2 – столбец таблицы. Матрица решений представляется множеством последовательностей вида $stugc$, где s – номер строки, t – номер столбца, u – выходной параметр, g – отношение «присвоить значение», c – константа. Координаты s и t являются номерами правил из наборов R_1 и R_2 , соответственно. Кортеж решений представляется множеством последовательностей $syugc$, где s – номер строки, y – выходной параметр, g – отношение «присвоить значение», c – константа. Координата s является индексом правила из набора R_1 . Схему выбора решения можно представить так:

$$P_1 \rightarrow R_1 \rightarrow M \leftarrow R_2 \leftarrow P_2, \quad P_1 \rightarrow R_1 \rightarrow C.$$

Такая структура позволяет хранить таблицы, выбор в которых идет по строкам и столбцам, а отдельные параметры выбираются только по строкам.

Однако эта структура позволяет хранить не любые таблицы. Во многих таблицах, например, таблицах для выбора режимов резания, выходные параметры уточняются с помощью поправочных коэффициентов, которых может быть несколько. Таблицы для выбора норм времени содержат вычисления по формуле. Исходя из этого, была разработана структура ТС, которая позволяла бы хранить таблицы любой сложности:

$$T = \langle X, Y, Z, E, B \rangle,$$

где X – множество входных параметров; Y – множество выходных параметров; Z – множество дополнительных параметров; $E = \{E_1, E_2, \dots\}$ – множество входов; B – тело таблицы.

Множество Z используется для промежуточных вычислений внутри ТС, чаще всего оно содержит поправочные коэффициенты. Входы таблицы E_i состоят из условий P_i и правил выбора R_i . Структура P_i аналогична традиционной. Структура R_i дополнена, позволяет применять логическую операцию «ИЛИ» для выбора условий и имеет вид $n\{N_1, N_2, \dots\}$. Тело таблицы представляет собой последовательность матриц решений, вычислений по формуле и вызовов внешних таблиц. Количество входов и матриц решений не ограничено, что позволяет хранить таблицы с любым количеством коэффициентов.

Структура базы знаний

ТС является сложным, иерархическим по структуре объектом, поэтому для его эффективной обработки внутренний язык должен быть достаточно простым. В то же время процесс поиска решения должен быть достаточно простым. Наиболее удобным языком является XML (eXtensible Markup Language – расширяемый язык разметки). Ниже приведена укрупненная структура XML-файла для хранения баз знаний.

```
<?xml version="1.0" encoding="windows-1251"?>
<DOCUMENT>
  <HEAD> <!-- Заголовок --> </HEAD>
  <TABLE>
    <PARAMETER-LIST> <!-- Список параметров -->
      <INPUT-PARAMETERS>
        <!-- Входные параметры -->
      </INPUT-PARAMETERS>
      <OUTPUT-PARAMETERS>
        <!-- Выходные параметры -->
      </OUTPUT-PARAMETERS>
      <EXTRA-PARAMETERS>
        <!-- Дополнительные параметры -->
      </EXTRA-PARAMETERS>
    </PARAMETER-LIST>
    <ENTRANCE-LIST> <!-- Список входов -->
      <ENTRANCE no="1"> <!-- Описание 1-го входа -->
        <CONDITIONS>
          <!-- условия -->
        </CONDITIONS>
        <RULES>
          <!-- правила выбора условий -->
        </RULES>
      </ENTRANCE>
      ...
      <ENTRANCE no="n">
        <!-- Описание n-го входа -->
      </ENTRANCE>
```

```

</ENTRANCE-LIST>
<BODY>
  <!--Тело таблицы -->
</BODY>
</TABLE>
</DOCUMENT>

```

Построение приложения

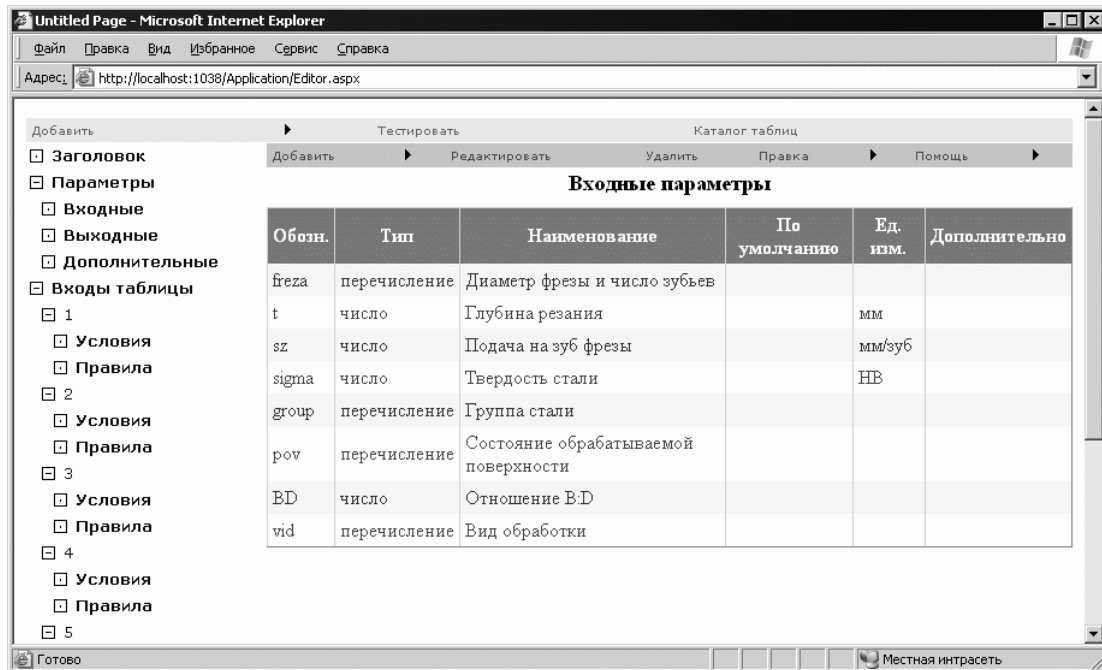


Рис. 1. Модуль ввода таблиц

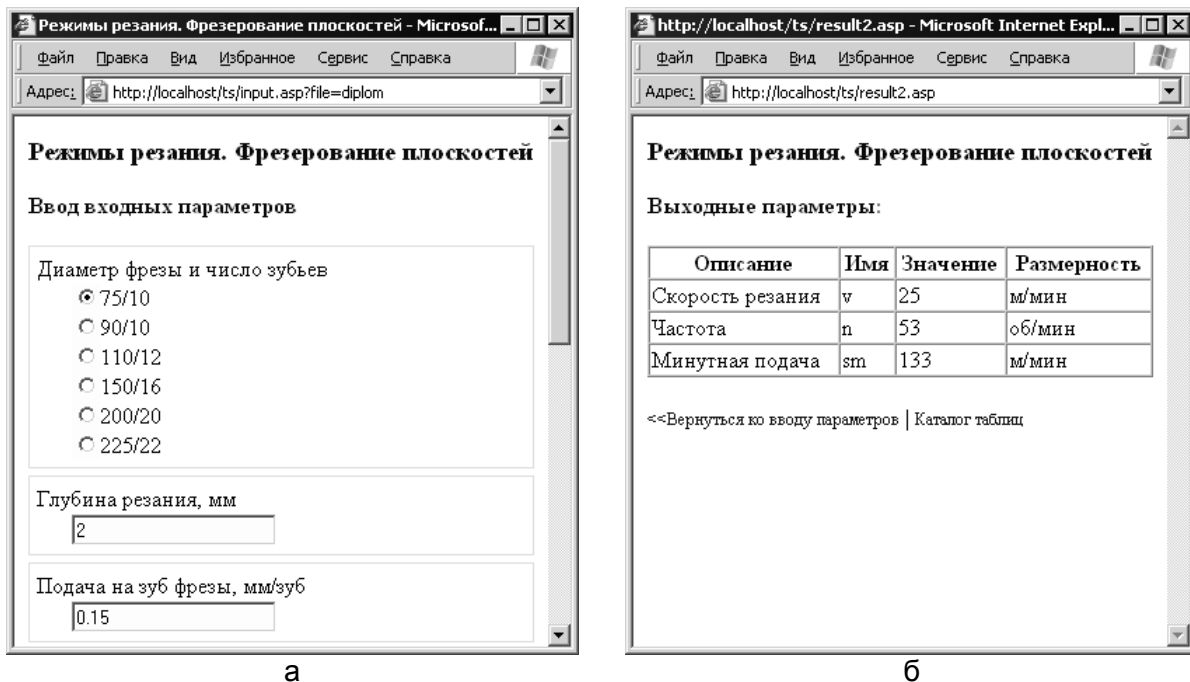


Рис. 2. Модуль принятия решения: а – ввод входных параметров, б – таблица выходных параметров

На основе модели ТС было написано приложение для ввода таблиц и принятия решения по ним. Приложение написано с использованием веб-технологий, так как с их помощью наиболее просто строить клиент-серверные приложения. Для написания модуля поиска решения использовалась платформа ASP, для ввода таблиц – платформа ASP.NET. Общий вид модуля ввода таблиц показан на рис. 1. На рис. 2 показан вид модуля принятия решения.

Заключение

В статье предложен способ хранения нормативно-справочной информации, выраженной в виде таблиц, в базе знаний. Представлена теоретическая структура базы знаний и способ ее хранения в формате XML, описан алгоритм принятия решения на основе базы знаний. Написано приложение для ввода таблиц и принятия решения по ним с использованием веб-программирования.

Литература

1. Митрофанов С.П., Куликов Д.Д., Миляев О.Н., Падун Б.С. Технологическая подготовка гибких производственных систем. / Под общ. ред. С.П. Митрофанова. – Л.: Машиностроение. Ленингр. отд-е, 1987 – 352 с.

ФОРМИРОВАНИЕ ОТЕЧЕСТВЕННЫХ СТАНДАРТОВ КАЧЕСТВА ЗУБЧАТЫХ КОЛЕС И ПЕРЕДАЧ, НЕ ПРОТИВОРЕЧАЩИХ РЕКОМЕНДАЦИЯМ ISO

М.В. Абрамчук

Научный руководитель – д.т.н., профессор Б.П. Тимофеев

В статье сравниваются величины минимального бокового зазора в стандартах ISO/TR 10064-2:1996, ISO 1328-2:1997 и ГОСТ 1643-81. По результатам сравнения делаются выводы о возможности использования рекомендаций ISO при формировании отечественного стандарта качества зубчатых колес и передач.

Введение

Последней редакции отечественного стандарта ГОСТ 1643-81 «Передачи зубчатые цилиндрические. Допуски» [1] исполнилось 27 лет. В нормировании точности зубчатых колес и передач произошли изменения. В настоящий момент производитель, желающий реализовывать свою продукцию на мировом рынке, должен учитывать, по крайней мере, минимальные требования, приведенные в стандартах ISO. Тем не менее, отказаться полностью от отечественных стандартов – ГОСТов мы не можем, да и не должны. ГОСТ 1643-81 содержит подробные таблицы, в которых приведены допуски и предельные отклонения для показателей точности зубчатых передач, в то время как ISO, предполагая наличие государственных, отраслевых стандартов, а также стандартов предприятий, таких конкретных данных практически не содержит.

Необходимость пересмотреть нормативную документацию с учетом современных реалий возникла уже давно. Авторы данной публикации подробно анализировали наиболее существенные недостатки ГОСТ 1643-81 [2, 3], а также проводили сравнительный анализ базового стандарта ГОСТ 1643-81 с рядом международных стандартов (которые были переведены авторами на русский язык) [4–6].

Воспользуемся величинами минимального бокового зазора, приведенными в техническом отчете «ISO/TR 10064-2:1996 Передачи зубчатые цилиндрические. Практическое руководство по приемке. Часть 2: Контроль суммарных радиальных отклонений, биения, толщины зуба и зазора» (далее – ISO/TR 10064-2:1996) [7] и рассчитаем с помощью формул, данных в ГОСТ 1643-81, некоторые параметры точности, нормирующие боковой зазор, для их применения при разработке нового отечественного стандарта. Полученные результаты сравним с величинами из таблиц ГОСТ 1643-81.

Сравнительный анализ позволит установить те диапазоны изменения основных параметров передач и те степени точности, в которых отечественная продукция уже соответствует нормам ISO.

Минимальный гарантированный боковой зазор

Величины минимального бокового зазора $j_{bn\min}$, приведенные в таблице стандарта в ISO/TR 10064-2:1996, можно рассчитать, пользуясь выражением

$$j_{bn\min} = \left(\frac{2}{3}\right) [0,06 + 0,0005a_i + 0,03m_n], \quad (1)$$

где a_i – межосевое расстояние, m_n – модуль зубьев нормальный.

В ГОСТ 1643-81 гарантированный боковой зазор $j_{n\min}$ определяется по формуле

$$E_{Hs} = -\left(\frac{j_{n\min} + k_j}{4 \sin \alpha}\right) \Rightarrow j_{n\min} = -(4E_{Hs} \sin \alpha + k_j), \quad (2)$$

где E_{Hs} – наименьшее дополнительное смещение исходного контура; α – угол зацепления; k_j – компенсация уменьшения бокового зазора, возникающего из-за погрешности изготовления зубчатых колес и монтажа передачи.

Величина компенсации определяется по формуле

$$k_j = \sqrt{(f_a 2 \sin \alpha)^2 + 2f_{pb}^2 + 2F_\beta^2 + (f_x \sin \alpha)^2 + (f_y \cos \alpha)^2}, \quad (3)$$

где f_a – предельное отклонение межосевого расстояния, f_{pb} – предельное отклонение шага зацепления, F_β – допуск на направление зуба, f_x – допуск на параллельность осей, f_y – допуск на перекося осей.

Произведем сравнение величин минимального бокового зазора в ISO/TR 10064-2:1996 и ГОСТ 1643-81, приняв во внимание, что в ISO/TR 10064-2:1996 величина зазора зависит от модуля зубьев m_n и минимального межосевого расстояния a_i , в то время как в нашем стандарте – от вида сопряжения и межосевого расстояния a_w . Сначала возьмем вид сопряжения С. В ячейке таблицы жирным выделены значения зазора, взятые из ГОСТ 1643-81.

m_n , мм	Минимальное межосевое расстояние a_i , мм					
	50	100	200	400	800	1600
	мкм					
1,5	90 74	110 87	–	–	–	–
2	100 74	120 87	150 115	–	–	–
3	120 74	140 87	170 115	240 140	–	–
5	–	180 87	210 115	280 140	–	–
8	–	240 87	270 115	340 140	470 200	–
12	–	–	350 115	420 140	550 200	–
18	–	–	–	540 140	670 200	940 310

Таблица 1. Вид сопряжения С

Как видно из табл. 1, во всех диапазонах модуля зубьев и межосевого расстояния значения гарантированного бокового зазора в ГОСТ 1643-81 меньше соответствующих величин в ISO/TR 10064-2:1996. Причем с увеличением m_n и a_i разница значительно возрастает.

Теперь возьмем вид сопряжения В.

m_n , мм	Минимальное межосевое расстояние a_i , мм					
	50	100	200	400	800	1600
	мкм					
1,5	90 120	110 140	–	–	–	–
2	100 120	120 140	150 185	–	–	–
3	120 120	140 140	170 185	240 230	–	–

5	–	180 140	210 185	280 230	–	–
8	–	240 140	270 185	340 230	470 320	–
12	–	–	350 185	420 230	550 320	–
18	–	–	–	540 230	670 320	940 500

Таблица 2. Вид сопряжения В

Как видно из табл. 2, при модуле зубьев $m_n=3$ мм величины минимального бокового зазора в ISO/TR 10064-2:1996 и гарантированного бокового зазора в ГОСТ 1643-81 практически совпадают. При $m_n<3$ мм минимальный боковой зазор по ISO/TR 10064-2:1996 меньше, чем в ГОСТ 1643-81, при $m_n>3$ мм – больше.

И, наконец, вид сопряжения А.

m_n , мм	Минимальное межосевое расстояние a_{i_s} , мм					
	50	100	200	400	800	1600
	мкм					
1,5	90 190	110 220	–	–	–	–
2	100 190	120 220	150 290	–	–	–
3	120 190	140 220	170 290	240 360	–	–
5	–	180 220	210 290	280 360	–	–
8	–	240 220	270 290	340 360	470 500	–
12	–	–	350 290	420 360	550 500	–
18	–	–	–	540 360	670 500	940 780

Таблица 3. Вид сопряжения А

Как видно из табл. 3, при модуле зубьев $m_n=8$ мм разница между величинами зазора в ISO и ГОСТ составляет 20–30 мкм. При $m_n<8$ мм минимальный боковой зазор по ISO/TR 10064-2:1996 меньше, чем в ГОСТ 1643-81, при $m_n>8$ мм – больше.

Предельное отклонение межосевого расстояния

Предельное отклонение межосевого расстояния f_a в ГОСТ 1643-81 [1, с. 61] определяется по формуле

$$f_a = \pm 0,5 j_{n \min} \cdot \quad (4)$$

Воспользуемся данной формулой и определим значения f_a , взяв в качестве $j_{n \min}$ числовые значения минимального бокового зазора из стандарта ISO/TR 10064-2:1996. Сведем результаты в таблицу. По ячейке: верхнее число – значение f_a по данным ISO/TR 10064-2:1996, нижнее, выделенное жирным, – значение f_a по ГОСТ 1643-81.

Сначала возьмем вид сопряжения С.

m_n , мм	Минимальное межосевое расстояние a_i , мм					
	50	100	200	400	800	1600
	МКМ					
1,5	45 35	55 45	–	–	–	–
2	50 35	60 45	75 55	–	–	–
3	60 35	70 45	85 55	120 70	–	–
5	–	90 45	105 55	140 70	–	–
8	–	120 45	135 55	170 70	235 100	–
12	–	–	175 55	210 70	275 100	–
18	–	–	–	270 70	335 100	470 155

Таблица 4. Вид сопряжения С

Как видно из табл. 4, при всех значениях модуля зубьев m_n и минимального межосевого расстояния a_i величины предельного отклонения межосевого расстояния в ГОСТ 1643-81 меньше соответствующих величин в ISO/TR 10064-2:1996. При модуле зубьев $m_n=1,5$ мм разница между величинами предельного отклонения межосевого расстояния в ISO/TR 10064-2:1996 и ГОСТ 1643-81 составляет 10 мкм. При $m_n>1,5$ мм, с увеличением m_n и a_i , разница значений f_a в обоих стандартах возрастает.

Теперь возьмем вид сопряжения В.

m_n , мм	Минимальное межосевое расстояние a_i , мм					
	50	100	200	400	800	1600
	МКМ					
1,5	45 60	55 70	–	–	–	–
2	50 60	60 70	75 90	–	–	–
3	60 60	70 70	85 90	120 110	–	–
5	–	90 70	105 90	140 110	–	–
8	–	120 70	135 90	170 110	235 160	–
12	–	–	175 90	210 110	275 160	–
18	–	–	–	270 110	335 160	470 250

Таблица 5. Вид сопряжения В

Как видно из табл. 5, при модуле зубьев от $m_n=1,5$ мм до $m_n=3$ мм разница между величинами предельного отклонения межосевого расстояния в ISO/TR 10064-2:1996 и ГОСТ 1643-81 составляет не более 15 мкм. При $m_n>3$ мм значения величины предельного отклонения межосевого расстояния в ГОСТ 1643-81 меньше соответствующих величин в ISO/TR 10064-2:1996. Причем, с увеличением m_n и a_i разница значительно возрастает.

Рассмотрим вид сопряжения А.

m_n , мм	Минимальное межосевое расстояние a_i , мм					
	50	100	200	400	800	1600
	мкм					
1,5	45 100	55 110	–	–	–	–
2	50 100	60 110	75 140	–	–	–
3	60 100	70 110	85 140	120 180	–	–
5	–	90 110	105 140	140 180	–	–
8	–	120 110	135 140	170 180	235 250	–
12	–	–	175 140	210 180	275 250	–
18	–	–	–	270 180	335 250	470 400

Таблица 6. Вид сопряжения А

Как видно из табл. 6, при модуле зубьев $m_n=8$ мм разница между величинами предельного отклонения межосевого расстояния в ISO/TR 10064-2:1996 и ГОСТ 1643-81 составляет не более 15 мкм. При $m_n<8$ мм значения величины предельного отклонения межосевого расстояния в ГОСТ 1643-81 больше соответствующих величин в ISO/TR 10064-2:1996, при $m_n>8$ мм – меньше.

Верхнее предельное отклонение измерительного межосевого расстояния

Верхнее предельное отклонение измерительного межосевого расстояния (для колес с внешними зубьями) $E_{a''_s}$ в соответствии с ГОСТ 1643-81 определяется по формуле

$$E_{a''_s} = +f_i'' , \quad (5)$$

где f_i'' – допуск на колебание измерительного межосевого расстояния на одном зубе.

В стандарте ISO 1328-2:1997 [8] f_i'' определяется по формуле

$$f_i'' = 2,96m_n + 0,01\sqrt{d} + 0,8, \quad (6)$$

где d – делительный диаметр.

Сравним величины f_i'' в ISO 1328-2:1997 и ГОСТ 1643-81. Результаты сведем в таблицу. По ячейке: верхнее число – значение f_i'' по данным ISO 1328-2:1997, нижнее, выделенное жирным, – значение f_i'' по ГОСТ 1643-81.

m_n , мм	Делительный диаметр, d , мм	Степень точности							
		5	6	7	8	9	10	11	12
		мкм							
от 1,5 до 2,5	от 50 до 125	6,5	9,5	13	19	26	37	53	75
		10	14	20	28	36	45	56	71
		15	22	31	44	62	87	123	174
13		18	25	36	45	56	71	90	
св. 4 до 6		24	34	48	67	95	135	191	269
14		20	28	40	50	63	80	100	
св. 6 до 10		6,5	9,5	13	19	27	38	53	75
11		16	22	32	40	50	63	80	
св. 4 до 6		15	22	31	44	62	87	124	175
14		20	28	40	50	63	80	100	
св. 6 до 10	24	34	48	67	95	135	191	270	
16	22	32	45	56	71	90	112		
от 1,5 до 2,5	св. 280 до 560	6,5	9,5	13	19	27	38	54	76
13		18	25	36	45	56	71	90	
св. 4 до 6		15	22	31	44	62	88	124	175
14		20	28	40	50	63	80	100	
св. 6 до 10		24	34	48	68	96	135	191	271
16		22	32	45	56	71	90	112	
от 1,5 до 2,5	св. 560 до 1000	7	9,5	14	19	27	38	54	77
14		20	28	40	50	63	80	100	
св. 4 до 6		16	22	31	44	62	88	125	176
16		22	32	45	56	71	90	112	
св. 6 до 10	24	34	48	68	96	136	192	272	
18	25	36	50	63	80	100	125		

Таблица 7. Колебание измерительного межосевого расстояния на одном зубе

Из табл. 7 видно, что в каждом диапазоне делительных диаметров в малых модулях зубьев (от 1,5 до 2,5 мм) величины f_i'' в обоих стандартах отличаются не более чем на 25–27 мкм. С увеличением модуля зубьев m_n в более грубых степенях точности (9–12) значение величины f_i'' в ISO 1328-2:1997 становится больше аналогичной величины в ГОСТ 1643-81.

Выводы

Величины гарантированного бокового зазора в ГОСТ 1643-81 частично совпадают с аналогичными величинами в стандарте ISO/TR 10064-2:1996 при модуле $m_n=3$ мм для вида сопряжения В и при модуле $m_n=8$ мм для вида сопряжения А. Для вида сопряжения С во всех диапазонах модуля зубьев и межосевого расстояния значения гарантированного бокового зазора в ГОСТ 1643-81 меньше соответствующих величин в ISO/TR 10064-2:1996. Таким образом, в данном случае при некоторых условиях наш стандарт ГОСТ 1643-81 задает более жесткие условия нормирования зазора, чем ISO/TR 10064-2:1996.

Величины предельного отклонения межосевого расстояния различаются с небольшой с разницей в обоих стандартах в видах сопряжения С (при $m_n=1,5$ мм), В (при m_n от 1,5 до 3 мм) и А (при $m_n=8$ мм). Для видов сопряжения С и В во всех остальных случаях и для вида сопряжения А (при m_n от 12 до 18 мм) величины предельного от-

клонения межосевого расстояния в ГОСТ 1643-81 меньше соответствующих величин в ISO/TR 10064-2:1996. Таким образом, в данном случае мы уже сегодня работаем в условиях более жестких требований, чем минимальные требования ISO/TR 10064-2:1996.

При сравнении величин верхнего предельного отклонения измерительного межосевого расстояния (сравнивался параметр f_i'' – допуск на колебание измерительного межосевого расстояния на одном зубе), выяснилось, что их значения при малых модулях (от 1,5 до 2,5 мм) различаются в обоих стандартах не более чем на 25–27 мкм. В подавляющем большинстве остальных случаев значения величины f_i'' в ГОСТ 1643-81 меньше соответствующих величин в ISO 1328-2:1997, что снова говорит о преимуществе ГОСТ 1643-81 в данном вопросе.

За счет наличия видов сопряжения наш стандарт ГОСТ 1643-81 создает значительную вариативность в нормировании бокового зазора, что позволяет детализировать рекомендации ISO, но, в принципе, им не противоречит. В формуле (1) член $0,03m_n$ расширяет боковой зазор в передаче в зависимости от модуля, что должно быть учтено при разработке стандарта нового поколения.

Стандарт ISO/TR 10064-2:1996 более полно учитывает условия эксплуатации зубчатых передач [7, с. 23]. Использование видов сопряжения в ГОСТ 1643-81 расширяет возможности конструктора в части назначения гарантированного бокового зазора $j_{n \min}$ и допуска на боковой зазор T_{jn} . Однако особенно в области малых значений величины T_{jn} следует все время проверять возможность обеспечения этого параметра используемой технологией изготовления зубьев.

Литература

1. ГОСТ 1643-81 Передачи зубчатые цилиндрические. Допуски. – М.: Издательство стандартов, 1989. – 68 с.
2. Абрамчук М.В., Тимофеев Б.П. Анализ недостатков ГОСТ 1643-81 // Вестник конференции молодых ученых СПбГУ ИТМО. Сборник научных трудов. Том 2. – СПб: СПбГУ ИТМО, 2004. – С. 39–42.
3. Тимофеев Б.П., Шалобаев Е.В. Состояние и перспективы нормирования точности зубчатых колес и передач // Вестник машиностроения. – 1990. – № 12. – С. 34.
4. Абрамчук М.В., Тимофеев Б.П. Рекомендации ISO 1328 в части установления параметров точности зубчатых колес и передач // II межвузовская конференция молодых ученых. Сборник научных трудов. Том 2. – СПб: СПбГУ ИТМО, 2005. – С. 127–131.
5. Абрамчук М.В., Тимофеев Б.П. Рекомендации по организации контроля бокового зазора в зубчатых передачах // Научно-технический вестник СПбГУ ИТМО. – 2006. – Выпуск 28. I сессия научной школы «Задачи механики и проблемы точности в приборостроении». – С. 206–210.
6. Абрамчук М.В., Тимофеев Б.П. Сравнение табличных значений параметров точности зубчатых колес и передач в стандартах: ISO 1328 и ГОСТ 1643-81 // Теория механизмов и машин. №1(9). Том 5. – СПб: СПбГПУ, 2007. – С. 60–70.
7. ISO/TR 10064-2:1996. Cylindrical gears. Code of inspection practice. Part 2. Inspection related to radial composite deviations, runout, tooth thickness and backlash.
8. ISO 1328-2:1997. Cylindrical gears – ISO system of accuracy – Part 2: Definitions and allowable values of deviations relevant to radial composite deviations and runout information.

ИСПОЛЬЗОВАНИЕ ЭЛЕКТРОННОЙ МОДЕЛИ ДЕТАЛИ ПРИ РАЗРАБОТКЕ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА

В.С. Гусельников

Научный руководитель – к.т.н., доцент Е.И. Яблочников

Связь разрабатываемого технологического процесса и электронной модели детали (изделия) – одна из задач автоматизации технологической подготовки производства. Автором предлагается решение данной задачи на основе сквозного использования электронной модели.

С помощью современных САПР можно построить электронные модели не только деталей или изделий, но и целых производств. Такие возможности позволяют более точно планировать реконструкцию и реорганизацию существующих производств, а также организацию новых, начиная с разработки планировок и заканчивая моделированием работы и расчетами загрузки технологического оборудования и обслуживающего персонала.

На современных и модернизируемых производствах разработка и хранение технологических процессов (ТП) ведутся с использованием современных PDM и ERP систем. Уже накоплен большой опыт использования технологами-программистами электронных моделей деталей для разработки управляющих программ и подготовки прототипов с использованием технологий быстрого прототипирования. В то же время конструкторские модели деталей и разработанные технологические процессы продолжают существовать параллельно, не являясь единым целым, в результате чего теряется часть информации, затрудняется работа технологических бюро и отделов.

Гибкость, возможность интеграции и работы с параметрами современных САПР позволяют отразить разрабатываемый технологический процесс на электронной модели детали и заготовки, наделив их, наряду с геометрическими, и физическими – технологическими признаками. Благодаря такой модели существенно упрощается разработка операционных эскизов, которые генерируются на основе модели промежуточного состояния заготовки.

В работах [1, 2] отмечается необходимость учета центральной роли электронной модели изделия при решении задач технологической подготовки производства (ТПП). Однако в литературе зачастую понятие электронной модели изделия трактуется неоднозначно, что связано с достаточно быстрым (за последние 10 лет) изменением содержания этого понятия. В связи с этим целесообразно провести краткий анализ процесса развития методов объемного моделирования и структуры электронной модели изделия.

Электронная модель детали (изделия) – документ, содержащий электронную геометрическую модель детали и требования к ее изготовлению и контролю (включая предельные отклонения размеров, шероховатости поверхностей и др.). На рис. 1 приведена схема состава электронной модели изделия [3].

Как известно, первые трехмерные (геометрические) модели имели каркасно-поверхностное представление. Каркасная модель – трехмерная электронная геометрическая модель, представленная пространственной композицией точек, отрезков и кривых, определяющих в пространстве форму изделия [1]. В процессе такого моделирования сначала строится каркас – пространственная конструкция, состоящая из отрезков прямых, дуг окружностей и сплайнов. Каркас играет вспомогательную роль и служит основой для последующего построения поверхностей, которые «натягиваются» на элементы каркаса. Особенность каркасно-поверхностного моделирования состоит в том, что элементы создаваемой модели никак не связаны друг с другом. Изменение одного из элементов не влечет за собой автоматического изменения других, что дает определенную свободу действий, но одновременно значительно усложняет работу с моделью

и требует больших затрат на ее модификацию. Это привело к появлению технологии твердотельного параметрического моделирования.

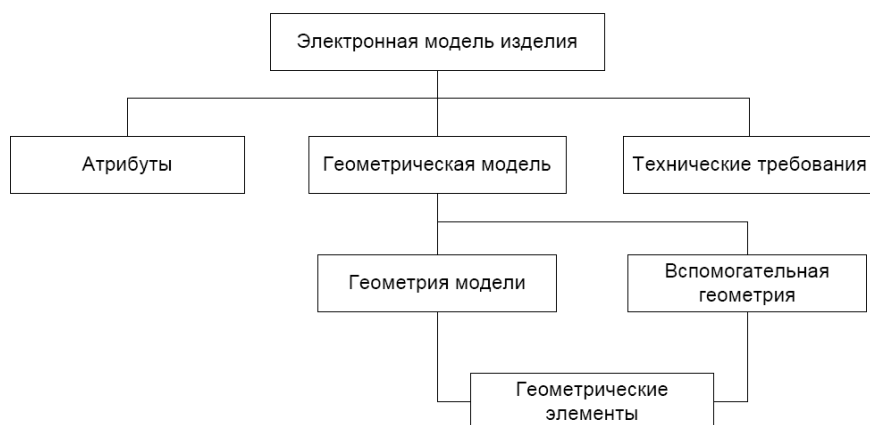


Рис. 1. Схема состава электронной модели изделия

Твердотельное моделирование имеет в своей основе идеологию, которая существенно отличается от идеологии каркасно-поверхностного моделирования. Твердотельная модель – трехмерная электронная геометрическая модель, представляющая форму изделия как результат композиции заданного множества геометрических элементов с применением операций булевой алгебры к этим геометрическим элементам [3]. Всегда можно точно сказать, находится ли точка внутри твердого тела, на его поверхности или вне тела. При изменении в модели любого элемента будут изменяться все другие элементы, которые связаны с ним. В результате изменится форма твердого тела, но сохранится его целостность.

Твердотельное моделирование предполагает возможность установки параметрических зависимостей между элементами твердого тела или нескольких тел. При этом изменение одного из параметров (например, длины элемента) приводит к соответствующей перестройке всех параметрически связанных элементов. Такое моделирование, называемое параметрическим, дает конструктору дополнительные удобства. Так, можно установить параметрические зависимости между элементами твердотельной сборки и, тем самым, автоматизировать контроль собираемости изделия. Кроме того, в твердотельной модели сохраняется история ее построения, которая позволяет возвращаться на любой из шагов проектирования и изменять форму модели методом изменения численных значений параметров или замены элементов, входящих в его историю.

Однако компоненты твердотельных моделей имеют определенные ограничения по сложности представляемых ими пространственных форм. Это привело к созданию технологии гибридного моделирования. При гибридном моделировании обеспечивается возможность одновременной работы с твердотельными объектами и с поверхностями. При этом можно «отрезать» поверхностью часть твердого тела, превращать замкнутый поверхностями объем в твердое тело и т.п. Гибридное моделирование позволяет сочетать все удобства твердотельного моделирования с возможностью построения объектов сколь угодно сложной геометрической формы. На рис. 2 представлена схема взаимосвязи типов представления формы электронной модели изделия.

При анализе рассматриваемой структуры электронных моделей возникает вопрос: что дают эти средства для автоматизации решения конструкторских и технологических задач в ТПП? Отметим, что важность представления и использования знаний в задачах ТПП понималась достаточно давно. В многочисленных работах предлагалось использование знаний в виде таблиц решений, правил, семантических сетей и фреймов для задач проектирования оснастки, инструмента и технологических процессов [4]. Ис-

пользование декларативных знаний обеспечивало гибкость создаваемых систем, их адаптируемость к особенностям предметной области и правилам принятия проектных решений. Однако отсутствие на тот период средств создания электронных моделей изделий не позволяло получить значимый практический эффект от выполненных разработок.

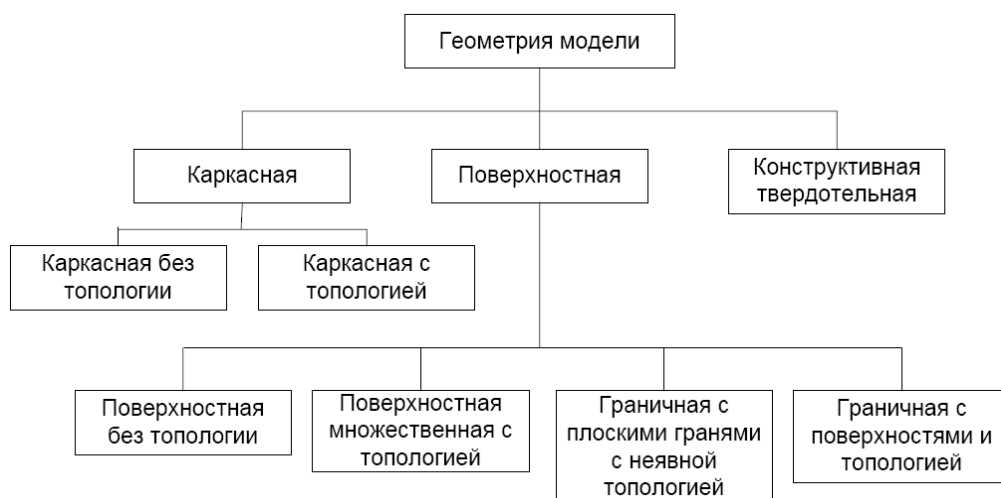


Рис. 2. Состав и взаимосвязь типов представления формы электронной модели изделия

С другой стороны, автоматизация конструкторского проектирования лишь за счет построения электронных моделей и последующего получения чертежей также во многих случаях не приносит должного эффекта в силу недостаточно высокого уровня автоматизации. Частичное улучшение дает разработка и использование специальных процедурных приложений к САД-системе (например, конструирование пакета пресс-формы с использованием баз нормализованных деталей); существенно больший эффект может дать интегрированное использование набора процедурных приложений. Однако этот подход не может быть реализован для всех видов проектных процедур ТПП как в силу их большого числа, так и по причине слабой формализации и типизации многих проектных решений.

Новый подход к решению задач автоматизации проектирования в ТПП за счет интегрированного использования электронных моделей и баз знаний способен привести одновременно и к гибкости создаваемой системы, и к существенному общему повышению уровня автоматизации.

Общая схема интегрированного использования электронных моделей и баз знаний приведена на рис. 3.

Здесь под приложением понимается некоторая проектная процедура ТПП, реализуемая средствами прикладного программного интерфейса (API) САД-системы и решающая конкретную задачу конструкторского или технологического проектирования с использованием базы корпоративных знаний.

На приведенной схеме не конкретизировано, является ли электронная модель моделью основного изделия или моделью изделия ТПП – это зависит от характера решаемой приложением задачи. Приложение может использовать в своей работе несколько моделей, а также вспомогательную информацию – например, геометрические или технологические шаблоны.

Применение данной схемы позволяет строить прикладные САПР, работающие «от технического задания» и генерирующие все необходимые геометрические модели, чертежи, технологические процессы, текстовые или текстово-графические документы.

Реализация каждой конкретной САПР требует определенных усилий, однако в результате достигаются высокий уровень автоматизации проектных решений и гибкость системы, а также создаются условия для решения кадровых проблем в сфере ТПП.

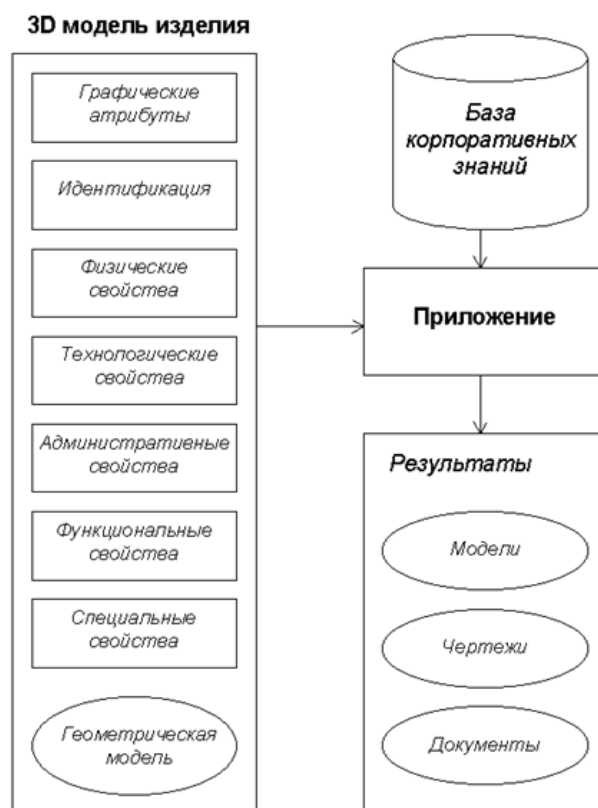


Рис. 3. Схема интегрированного использования электронной модели и базы знаний

Интегрированная среда на базе PDM и CAD/CAM/CAE-систем объединяет конструкторов и технологов и позволяет создавать электронные модели деталей и заготовок с максимальным уровнем технологической проработки.

CAD-система графически отображает в дереве спецификации модели ее структуру на основе примитивов и другие неграфические данные. Дерево спецификации показывает последовательность, в которой примитивы были созданы, и обеспечивает легкий доступ ко всей ассоциированной информации и элементам, лежащим в основе этого примитива. Размеры и отношения, используемые для создания примитивов, сохраняются в модели. Это обеспечивает соответствие модели ее техническому заданию и легкое изменение модели с помощью параметров.

Управляющие размеры – это размеры, используемые для построения примитива. Они включают размеры, ассоциированные как с геометрией эскиза, так и собственно с примитивами. Рассмотрим, например, цилиндрическую призму. Диаметр призмы управляется диаметром окружности в эскизе, а ее высота – глубиной, на которую выдавливается окружность.

Отношения включают в себя такую информацию, как параллельность, касание и концентричность. Такая информация обычно приводится на рисунке с помощью контрольных символов. Отражая эту информацию в эскизе, CATIA позволяет заведомо полностью зафиксировать в модели технические требования на ее проектирование.

Таким образом, ограничившись разработкой и ведением ТП в PDM, технолог все равно не использует всю информацию, которая содержится в электронной модели детали. Для того чтобы сделать проектирование ТП более наглядным, необходимо ис-

пользовать электронную модель в проектировании ТП. При таком подходе появляется возможность отслеживания перехода электронной модели от первоначальной заготовки непосредственно до электронной модели детали. Это на первом этапе дает возможность более точной оценки принятых решений в ТП.

Модель, получаемая в САД-системе, полностью ассоциирована с чертежами и деталями или сборками, на которые она ссылается. Изменения модели автоматически отражаются в ассоциированных с ней чертежах, деталях, сборках, и, наоборот, изменения, выполненные в контексте чертежа или сборки, отражаются на модели. Взаимосвязи между примитивами модели устанавливаются геометрическими ограничениями (такими, как параллельность, перпендикулярность, горизонтальность, вертикальность, концентричность и совпадение), фиксируя положение примитивов, друг относительно друга. Кроме того, можно применять математические уравнения с параметрами.

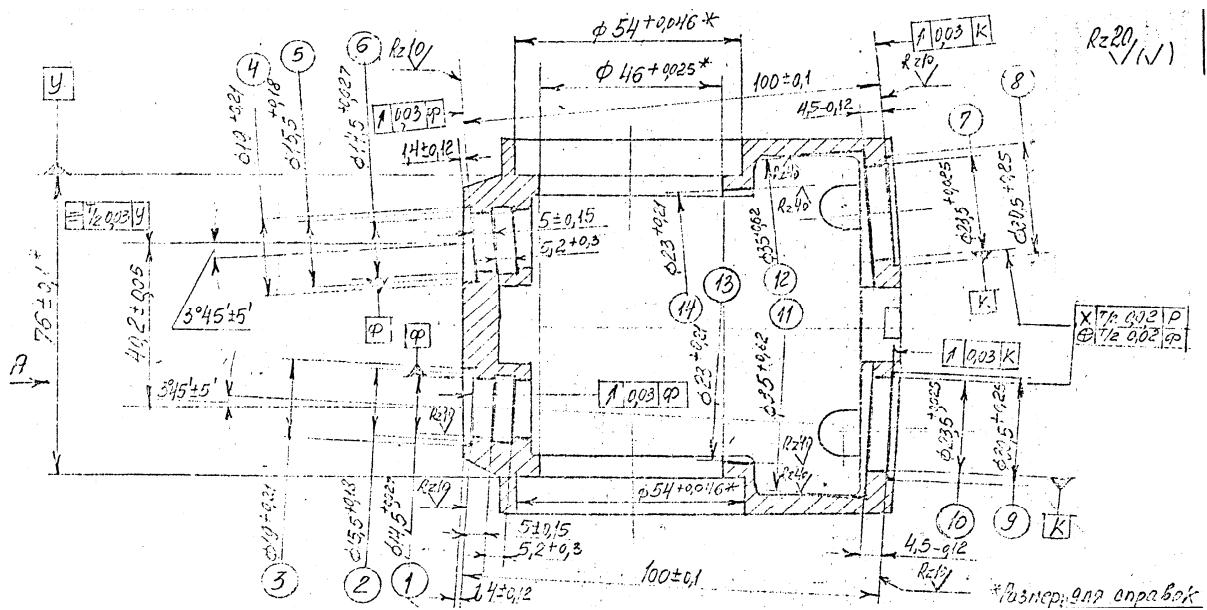


Рис. 4. Пример перегруженного эскиза

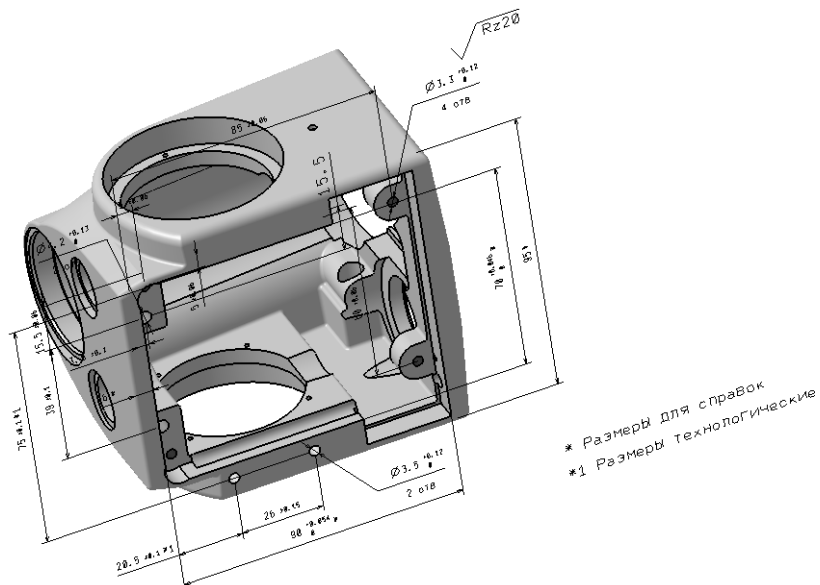


Рис. 5. Электронная модель с указанными на ней технологическими размерами

Метод твердотельного моделирования, используемый для построения, может влиять на множество аспектов, включая гибкость к изменениям, устойчивость в процессе изменений и требования к ресурсам для вычисления нового результата. Поэтому учет технических требований важен для достижения наибольшей эффективности твердотельной модели детали.

Современные изделия могут иметь очень сложную форму, множество сложных поверхностей, литейных скруглений и т.д. в связи с чем, часто возникают ситуации, когда операционный 2D эскиз перегружен размерами и проекциями сложных поверхностей (рис. 4). Для облегчения работы технолога в соответствии с [3] можно проставлять размеры, ценные указания и технологические требования к операции/переходу непосредственно на электронной модели, которые при необходимости могут быть автоматически перенесены на 2D эскиз (рис. 5). Для удобства отображения и перемещения по дереву ТП создаются специальные группы, позволяющие скрывать или отображать интересующую операцию с припуском и технологическими размерами.

Реализация метода «обратного проектирования» при разработке ТП с использованием электронных моделей детали и заготовки позволяет получить электронные модели заготовок, входящих на каждую формообразующую операцию, а также установить ассоциативные связи между электронной моделью заготовки и ее 2D эскизом. Это позволяет получить представление о степени обработки заготовки, для большей наглядности рекомендуется сделать модель цветной, где три разных цвета будут представлять конструкторскую модель, припуски и припуск, снимаемый на данной операции (рис. 6).

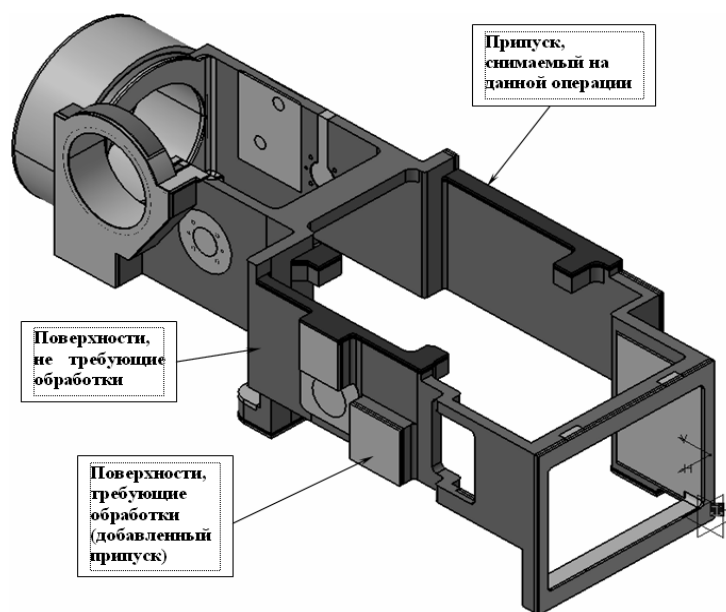


Рис. 6. Итоговая электронная модель

Таким образом, благодаря организованной среде можно получить технологическую модель изделия с описанием всего происходящего с ней на протяжении технологического процесса.

Параметрический подход, реализуемый в методах твердотельного и гибридного моделирования, дает возможность автоматического повторения метода построения геометрии при изменении одного или нескольких его аргументов, в пределах возможности самого метода. Разработчик изделия, таким образом, получает возможность варьировать различными параметрами с целью оптимизации целевого качества. В качестве параметров (аргументов) могут выступать геометрические элементы (точки, прямые, кривые, плоскости, поверхности), численные параметры с единицами измерения (расстояния, углы) и численные параметры без размерности.

Дальнейшим развитием параметрического подхода явилась алгебраическо-сценарийная параметризация. Она позволила обогатить историю построения простейшими ассоциативными связями между элементами и определяющими их размерами. Это было важное расширение «спецификации геометрического определения». Возможности модификации геометрии существенно повысились. Одним из проявлений алгебраическо-сценарийной параметризации стало моделирование на основе стандартизованных схем построения («Feature Based Design»). Оно дало возможность конструктору хранить образец метода геометрического построения для его последующего повторного применения.

Благодаря возможностям параметризации современных САПР можно реализовать аналогичные электронные модели заготовок для типового и группового ТП. Внутренняя структура модели может быть представлена в виде перечня элементов, а также свойств, присущих этим элементам. При этом для любого свойства есть возможность добавить формулу, которая преобразует это свойство в параметр, способный влиять как на элемент, так и на модель в целом. Благодаря этому можно добиться не только получения деталей с различными габаритами, любых геометрических элементов (т.е. различных вариантов исполнения деталей), но и полного изменения геометрии модели. Именно использование этой возможности позволяет создать параметрическую модель комплексной детали.

Описанные параметрические данные могут храниться в отдельном внешнем файле табличного процессора, что позволяет воспользоваться всеми функциями этого приложения для управления табличными данными. Таким образом, связь между различными параметрами может быть реализована не только в самой САПР, но и в таблице, хранящей значения параметров. Это может оказаться удобным в том случае, когда для различных конфигураций геометрии комплексной детали связь параметров в виде прямых зависимостей не является постоянной. Тогда, в одном случае, при наличии зависимости параметров в виде функции, зависимость реализуется формулой табличного процессора, а в другом – значения параметра могут задаваться в явном виде. Это может использоваться, например, для связи вспомогательной геометрии с реальной. В одном случае значения задаются явно, а в другом – вычисляются в табличном процессоре. Кроме того, в таблице могут храниться не только числовые значения параметров, но и текстовая информация, которая может быть как описанием модели, так и значением текстовых и логических переменных. Таким образом, графическая составляющая типовых и групповых проектов может содержаться в одном файле с параметрически описанной электронной моделью.

Электронные модели детали и заготовки могут использоваться для разработки технологического процесса, для накопления и использования, приобретенных ранее знаний и опыта проектирования – базы знаний, включающей прототипы изделий и технологические процессы их изготовления, нормативно-справочную информацию, правила проектирования и т.д. Наряду с общим сокращением временных затрат на разработку, модернизацию ТП, это позволяет точно контролировать весь технологический процесс, своевременно находить места сбоев и получения брака, а также сводит к минимуму риск повреждения технологического оборудования.

Литература

1. Зильбербург Л.И., Молочник В.И., Яблочников Е.И. Реинжиниринг и автоматизация технологической подготовки производства в машиностроении. – СПб: Политехника, 2004.
2. Голдовский П.С. Боткин Ю.А. Интегрированная САПР технологической подготовки производства // САПР и графика. – 2004. – № 9.
3. ГОСТ 2.052-2006. Электронная модель изделия. Общие положения.
4. Пелипенко А.Б., Яблочников Е.И. Современные тенденции в развитии CAD/CAM-технологий: ориентация на процессы // САПР и графика. – 2001. – № 9.

ПРИНЦИПЫ ОРГАНИЗАЦИИ И ИНТЕРФЕЙС УНИФИЦИРОВАННОГО МОДУЛЯ ИНТЕГРАЦИИ PDM- И CAD-СИСТЕМ

Т.Д. Голицына

Научный руководитель – к.т.н., доцент Т.А. Павловская

Статья посвящена вопросам интеграции систем управления PDM- и CAD-систем в машиностроении. Рассматривается существующий подход к решению этой задачи. Описывается унифицированный подход. В работе приведены принципы организации интегрирующего модуля и API системы.

Введение

Традиционный подход к построению систем, автоматизирующих взаимодействие систем управления данными об изделии (PDM) и систем автоматизированного проектирования (CAD) в машиностроении, представляет собой разработку частного решения для конкретного предприятия, т.е. интеграцию конкретной PDM-системы с конкретной CAD-системой. При этом проблемы, возникающие при интеграции, решаются в каждом случае различными, зачастую новыми, способами. Поскольку такие системы в подавляющем большинстве являются коммерческими, конкретная программная реализация обычно скрыта, поэтому провести анализ существующих решений с целью выбора лучшего или более универсального не представляется возможным. Кроме того, данный способ взаимодействия обладает перечисленными ниже недостатками.

1. Ограниченность применения разработанного программного обеспечения, поскольку это решение, как правило, специфично для решаемой задачи.
2. Для того чтобы заменить по каким-либо причинам одну из систем, потребуется полная переработка программного обеспечения.
3. Для использования нескольких систем одновременно необходима существенная модификация программного обеспечения.
4. Изменение API какой-либо из систем влечет за собой необходимость внесения изменений в прикладное программное обеспечение.

В то же время в направлении решения описанной проблемы сделаны серьезные шаги. Так, международный стандарт ISO 10303 (и его русскоязычная версия ГОСТ Р ИСО 10303 [1]) призван определить единый способ обмена информацией между всеми системами, содержащими данные об изделии [2]. Это позволит автоматизировать взаимодействие любых PDM- и CAD-систем, поддерживающих стандарт, в общем случае, без дополнительных интегрирующих программ. Стандарт определяет форму представления данных в виде текстового обменного файла, имеющего строго регламентированную структуру. Обменный файл используется для передачи данных между различными автоматизированными системами проектирования.

Однако в настоящее время существуют сложности в применении стандарта. Во-первых, несмотря на то, что прошло уже более 10 лет со дня выхода стандарта, еще не все CAD- и PDM-системы его поддерживают. Во-вторых, реализации стандарта для разных систем различаются (особенно это актуально для разных типов систем), и, как следствие, корректно из системы в систему переносится только информация о составе изделия. Так, например, при экспериментах с системами PDM Step Suite и CAD SolidWorks и PRO/E данные о размерах и параметрах детали не передавались, а попытка перенести данные из SolidWorks в PSS, а затем обратно приводила к полной потере информации о детали. Таким образом, реализация взаимодействия PDM- и CAD-систем существенно неполна и не позволяет оперировать всеми атрибутами изделия.

В рамках разработки унифицированного подхода к интеграции PDM- и CAD-систем [3] в настоящей статье предлагается решение этой проблемы.

Унифицированный подход

Альтернативой подходу к разработке программного решения для интеграции конкретной CAD-системы и конкретной PDM-системы является унифицированный подход, позволяющий сделать взаимодействие PDM-систем и CAD-систем единообразным и легко расширяемым. Общая схема предлагаемой архитектуры программного комплекса представлена на рис. 1.

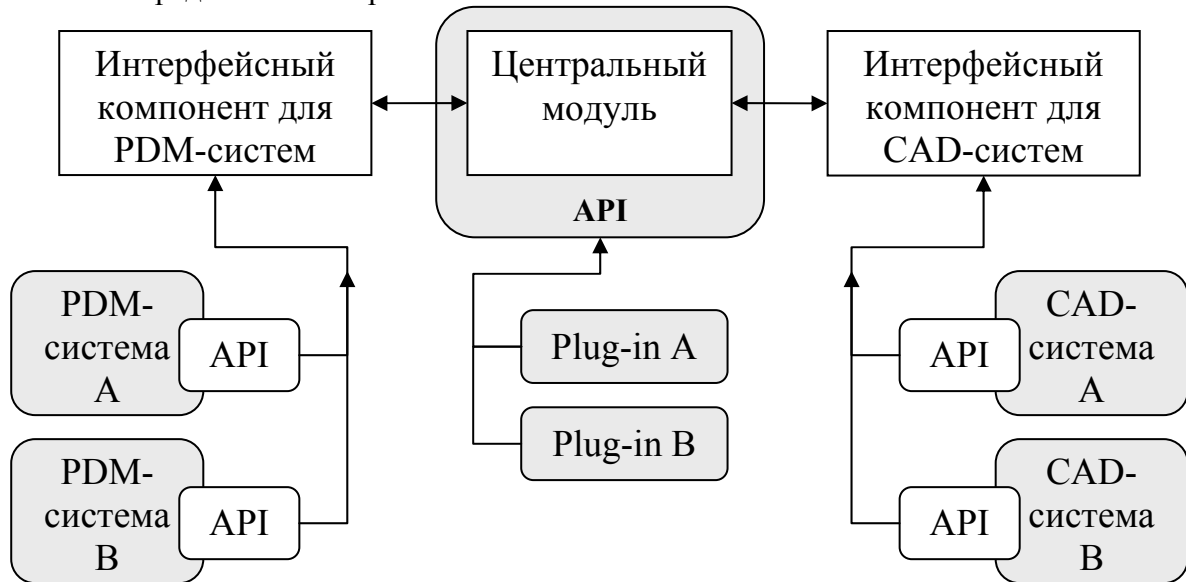


Рис. 1. Схема архитектуры программного комплекса

Как видно из рисунка, программный комплекс представляет собой взаимодействие нескольких CAD- и PDM-систем через интерфейсные компоненты и центральный модуль.

Центральный модуль. Центральный модуль является ответственным за взаимодействие систем. Например, модель изделия из CAD-системы должна преобразоваться в иерархическую структуру изделия, его составляющих и их параметров в PDM-системе. Модуль расширения должен иметь возможность доступа к параметрам и их преобразованию (например, перерасчет размеров). После этого структура изделия в PDM-системе, содержащая обновленную информацию о размерах каждой компоненты изделия, должна передаваться в CAD-систему с проведением соответствующих изменений. Для этих целей центральный модуль предоставляет единый программный интерфейс (API) для CAD-систем, PDM-систем, систем для расчета параметров изделий, а также различных модулей расширения.

Поскольку стандарт STEP создавался именно для унификации взаимодействия между системами автоматизации и поддержки производства, было принято решение положить его в основу унифицированного подхода. Стандарт содержит спецификацию стандартизованного интерфейса доступа к данным (Standard Data Access Interface – SDAI). Эта спецификация представляет собой набор функций для языков C и C++, обеспечивающих доступ к объектам в базе данных или текстовом файле. Этот интерфейс и определяет интерфейс центрального модуля.

Интерфейсные компоненты. Для работы с API центрального модуля CAD- и PDM-системам достаточно поддерживать стандарт STEP. В случаях, когда система полностью или частично не поддерживает стандарт, но имеет свой родной API, ее можно подключить через интерфейсный компонент.

Интерфейсные компоненты должны реализовывать следующие функции:

- (1) организация взаимодействия с одной или несколькими PDM- или CAD-системами;
- (2) локализация информации об особенностях конкретной системы;
- (3) поддержка API центрального модуля.

Интерфейсные компоненты могут разрабатываться или расширяться по необходимости подключения новых систем к программному комплексу на основании API CAD- или PDM-системы и API центрального модуля.

Модули расширения. Модулем расширения может быть любая вспомогательная программа, которой необходимо иметь доступ к данным, хранящимся в PDM- или CAD-системе. Для работы с этими данными такая программа должна поддерживать единый интерфейс центрального модуля, который позволит забрать для обработки необходимые данные и загрузить обратно результаты работы программы. Модули расширения могут добавляться в комплекс по мере необходимости.

Результаты реализации

Схема центральной части программного комплекса StepCAD приведена на рис. 2.

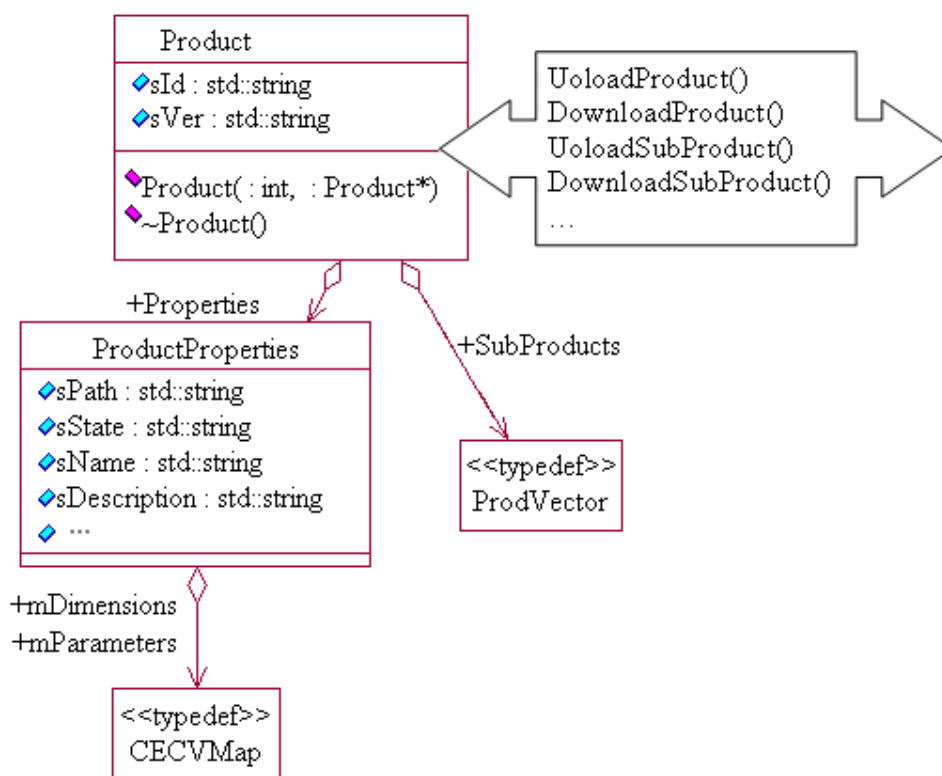


Рис. 2. Реализация программного комплекса

Программный комплекс StepCAD разрабатывается на основании приведенных выше принципов. Разработка ведется в среде Microsoft Visual C++ 6.0.

К настоящему моменту разработан интерфейсный компонент для PDM Step Suite [4]. В качестве основы интерфейсного компонента для CAD-систем в состав комплекса включена библиотека GSCADLink [5], поддерживающая работу с SolidWorks, Pro/E, SolidEdge, Unigraphics, Inventor, Компас 3D, AutoCAD, PCAD и CATIA. Ведется работа по преобразованию интерфейса GSCADLink в интерфейс центрального модуля.

Тестирование и отладка программного комплекса проводилось на примере взаимодействия PDM Step Suite, CAD SolidWorks и PRO/E.

Преимущества унифицированного подхода

Итоги опытной эксплуатации прототипа программного комплекса с предложенной архитектурой позволяет говорить о следующих достоинствах предлагаемого метода:

1. высокая степень универсальности предлагаемого решения, его применимость для решения многих задач;
2. простота расширения программного комплекса другими PDM- и CAD-системами, а также модулями расширения. При этом изменения, вносимые в центральный модуль для поддержки новой функциональности, минимальны;
3. унификация взаимодействия с PDM- системами и CAD-системами;
4. относительная независимость центрального модуля от изменений, вносимых в API PDM-систем или CAD-систем. Как результат, представленная архитектура является экономически выгодной: добавление новых систем производится по уже имеющемуся «образцу», а внедрение и поддержка программного комплекса – по аналогичным сценариям.

Кроме того, предложенное решение позволяет существенно расширить возможности стандарта по взаимодействию PDM- и CAD-систем. Например, оно позволяет определять ключевые характеристики изделий (масса, размеры, авторство) независимо от исходной системы и корректно их обрабатывать. Так, проводился тест для изделия, импортированного в PDM-систему, размеры которого затем изменялись, и проводился экспорт в CAD-систему. Размеры модели в CAD-системе соответственно изменялись. Имя автора изделия также корректно интерпретировалось при всех преобразованиях.

Поскольку предложенная архитектура позволяет не только экспортировать информацию об изделии, но и сопровождать ее всем необходимым, был предложен, реализован и опробован подход, при котором исходный файл модели в формате CAD-системы хранится в качестве документа в PDM-системе. При таком подходе изменения файла контролируются PDM-системой, а сам файл неразрывно связан с описанием изделия в PDM-системе. Такая функциональность оказалась очень удобной при работе и частом обращении к обеим системам.

Кроме того, такая архитектура комплекса позволяет оптимизировать работу PDM-системы. Так, например, файл в формате CAD-системы было решено привязать не к каждому изделию из числа одинаковых, а только к одному из них, в то время как остальные только ссылаются на него. Все это стало возможным благодаря тому, что центральный модуль обладает гораздо большей информацией о логике организации изделия, чем обменный файл.

Наряду с преимуществами, были выявлены проблемы унифицированного подхода.

1. Сложность сбора требований. Поскольку внедрение программного комплекса предполагается на нескольких предприятиях, то сбор и обработка требований представляет собой достаточно сложную задачу по сведению требований нескольких заказчиков в единую систему.
2. Сложность интерпретации данных, полученных в результате выгрузки или синхронизации информации: поскольку предлагаемая схема интеграции предполагает использование различных PDM-систем и различных CAD-систем, то необходимо обеспечить корректную передачу данных из одной системы в другую. Это достигается сложной системой конфигурации.

Заключение

В настоящей работе рассмотрены два подхода к интеграции PDM-систем и CAD-систем. Приведена архитектура программного комплекса, реализующего унифицированный подход, принципы организации API центрального модуля и результаты реализации подхода.

Литература

1. ГОСТ Р ИСО 10303 «Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными». – М.: Госстандарт России, 2000.
2. Mason H. ISO 10303 – STEP. A key standard for the global market. // ISO Bulletin. Апрель 2002. – С. 9–13.
3. Голицына Т.Д. Проблемы интеграции PDM и CAD систем. Унифицированный подход. // Научно-технический вестник СПбГУ ИТМО. – 2007. – Выпуск 39. Исследования в области информационных технологий. Труды молодых ученых. – С. 164–168.
4. PDM Step Suite. Техническое описание. http://pss.cals.ru/DOC/PSS_TD_2_1.pdf
5. GSCADLink. Описание. // Официальный сайт ООО «Глосис-Сервис». <http://glosys.ru/products/cad/GSCADLink.htm>

ЭВОЛЮЦИЯ МЕТОДОВ И ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

Е.Б. Боб, А.В. Латников, А.М. Мальцев, А.Е. Потехин
Научный руководитель – д.т.н., профессор А.А. Шалыто

В статье приводится анализ недостатков объектно-ориентированного метода программирования и соответствующих технологий. Предлагается использование технологии автоматного программирования в качестве альтернативы объектно-ориентированному программированию.

Введение

В данной статье рассматриваются основные этапы развития методов и технологий программирования, а также поднимается вопрос об основных причинах, повлекших описанный процесс. Анализ недостатков объектно-ориентированного программирования (ООП) как метода программирования, а также соответствующих технологий [1], позволяет поставить под сомнение возможность дальнейшего развития этого направления. Авторами предлагается использовать технологию автоматного программирования [2] в качестве альтернативы ООП, что позволит устранить существующие недостатки, а также ускорить технологический процесс, одновременно улучшив такие качества программного продукта, как надежность, масштабируемость и переносимость.

Эволюция методов программирования

У истоков всех языков программирования лежат машинные коды. Формально, возможно, даже неправильно называть их именно языком, так как программирование в чистом виде представляло собой не какую-нибудь абстракцию, а жесткое задание инструкций аппаратуре, которые должно было выполнить вполне конкретное компьютерное «железо». Важно отметить, что программа в то время полностью соответствовала своему поведению, иными словами, невозможна была ситуация, когда программа выполняла действия, которые не мог объяснить программист. Отчасти это следствие относительной простоты программ, отчасти – досконального знания архитектуры компьютера самим программистом. Процесс программирования был сложным, но вместе с тем прозрачным.

Первым шагом к абстрагированию программиста от компьютерного «железа» стал язык ассемблера. Впервые появилось некое подобие переносимости программ – для семейства машин со схожим набором инструкций возможно было написать одну ассемблерную программу, демонстрировавшую одинаковое поведение для всех представителей этого семейства. Однако наиболее значительный шаг с точки зрения эволюции – появление компилятора как промежуточного звена в цепочке «программист – компьютер». Отныне программист, с одной стороны, освобождался от необходимости досконально изучать архитектуру той машины, с которой работал, а с другой – целиком полагался на «интеллект» компилятора, своеобразного переводчика с языка программирования на язык, понятный машине.

Дальнейшее направление развития языков программирования вполне очевидно – раз появилась возможность абстрагировать программиста от платформы, необходима разработка как можно более универсальных средств такой абстракции. *Фортран*, *Кобол*, *Бейсик*, *Паскаль*, *Си* – все это отдельные точки на выбранном маршруте.

Отдельной вехой в развитии методов программирования стоит выделить появление объектно-ориентированных языков, первым из которых стал язык *Simula*, предложенный в 1967 г. [3]. Впоследствии объектно-ориентированные надстройки появились практически для всех используемых на тот момент языков программирования. Наиболее популярный пример – язык *Си++*.

Здесь стоит еще раз остановиться именно на методах программирования вне зависимости от используемых языков. От оперирования машинными кодами программисты постепенно уходили ко все более абстрактным операциям – инструкциям, процедурам, структурам, классам. Несложно заметить, что эволюция пошла по пути упрощения структуры языка – по пути упрощения программирования для самих программистов. Следствиями такого подхода являются:

- универсализация языков программирования с точки зрения решаемых задач и архитектуры компьютера;
- усложнение программ за счет введения все большего числа абстрактных понятий и структур;
- как следствие, тотальное усложнение компиляторов, обеспечивающих согласование «абстрактных структур», которыми оперирует программист, с тем, что реально работает «в железе».

Последнее – основная причина неэффективного использования компьютерных ресурсов. «Неэффективные программы – отнюдь не кощунство. Кощунство – это языки, которые заставляют программистов выполнять ненужную работу. Не расход машинного времени, а пустая трата времени программиста – вот истинная неэффективность» [4].

Как отмечалось выше, наиболее распространенным методом программирования в настоящее время является ООП. Конечно, оно обладает рядом существенных преимуществ по сравнению с теми методами, которые стоят ниже по эволюционной лестнице [5]:

- унификация принципов работы с типами данных, возможность определять новые типы совместно с методами их обработки;
- расширяемость и модульность кода;
- наглядность структуры программного кода;
- локализация данных и методов, объединенных определенными логическими отношениями.

Не стоит забывать при этом и про очевидные недостатки ООП, среди которых стоит особо выделить трудности при описании поведения объектов и программ в целом. Иногда под термином «поведение» ошибочно понимают набор методов, которые предоставляет класс. Однако набор методов – это всего лишь список тех действий, которые объект может выполнить над другими объектами. Этот набор лишен логической последовательности, которая и определяет поведение. Поэтому появляется необходимость в сложной и запутанной документации, которая призвана хоть как-то улучшить понимание программ.

К недостаткам ООП также можно отнести и неэффективное использование компьютерных ресурсов, о котором уже упоминалось выше. Использование сторонних библиотек со скрытой реализацией и недостаточно подробной документацией может отрицательно сказаться на производительности продукта. При этом ООП не позволяет добиться значительного улучшения продуктивности разработки по сравнению с процедурным программированием [6].

Одним из способов устранить или хотя бы сгладить недостатки методов программирования является изменение, а зачастую и усложнение, технологического процесса разработки.

Современный технологический процесс

В любой организации, занимающейся разработкой программного обеспечения, существует некоторый набор правил и методик, согласно которому сотрудники выпускают программный продукт. Эти правила могут быть тщательно задокументированы или же, наоборот, хаотичны, но именно они и называются процессом разработки.

Одной из наиболее известных моделей процесса разработки является «модель водопада» (*Waterfall*), предложенная еще в 70-х гг. XX века. После нее появилось множество самых разнообразных процессных моделей: итеративная, спиральная, *Rational Unified Process*, *eXtreme Programming* и т.д. Существуют даже международные стандарты на процессы разработки (*ISO 9000*, *CMM/CMMI*) [7, 8].

Основной причиной такого многообразия является желание уменьшить производственный цикл, не ухудшив конечный результат – сам программный продукт. В качестве оптимального решения предлагается использовать ту или иную модель в зависимости от размеров проекта: сложности продукта, количества программистов в команде и т.д. Тем не менее, ни одна модель пока не принесла действительно значительных улучшений [9].

Следует отметить, что фазы технологического процесса практически не изменяются. В зависимости от процессной модели может изменяться порядок или взаимодействие фаз друг с другом. Некоторые модели предлагают методики и инструментальные средства для использования на каждой фазе.

В таблице приведены общепринятые фазы процесса разработки [10], а также примерное процентное соотношение временных затрат на каждую фазу для моделей *Waterfall* и *eXtreme Programming* (таблица).

Фаза	Описание	<i>Waterfall</i>	<i>eXtreme Programming</i>
Анализ требований к продукту	Сбор требований к системе, их систематизация, документирование и анализ	20%	20%
Разработка архитектуры продукта	Создание функциональной модели системы, удовлетворяющей требованиям	20%	10%
Кодирование	Непосредственно программирование: перевод функциональной модели в программный код	10%	55%
Тестирование	Проверка программы на соответствие требованиям и наличие внутренних ошибок	35%	
Документирование	Создание документации пользователя	15%	15%
Сопровождение	Внедрение и поддержка готового продукта	N/A	N/A

Таблица. Фазы технологического процесса

Из таблицы видно, что в теории наименее длительными являются фазы документирования и кодирования (для модели *Waterfall*). Действительно, это вполне возможно, но только при тщательной и детальной разработке архитектуры программы, что, к сожалению, крайне редко. На практике усилия на кодирование значительно увеличиваются за счет ошибок и неточностей при сборе требований и разработке архитектуры.

Для компаний по производству программного обеспечения обычно наивысший приоритет имеет срок сдачи продукта заказчику, а отнюдь не качество программы с точки зрения программирования. В ситуации, когда поставка продукта уже не за горами, а программа еще не готова, попытка уменьшить любую из фаз разработки приводит к ухудшению качества конечного продукта:

- при сокращении затрат на тестирование естественным образом уменьшается надежность системы;
- уменьшение затрат на разработку архитектуры приведет в первую очередь к более длительному циклу кодирования;
- сокращение фазы документирования отражается на качестве пользовательской документации. При этом не факт, что вообще кто-то поймет, как пользоваться программой;
- при сокращении фазы сбора и анализа требований появляется риск, что готовый продукт окажется совершенно не тем, что требовалось заказчику.

Наиболее затратной оказывается фаза сопровождения системы, которая может тянуться годами (конечно, если заказчик это оплачивает). Именно на этой фазе всплывают основные недостатки уже готового продукта, и именно тогда разработчики начинают ощущать последствия непродуманной архитектуры и непрозрачного кода. Ошибочная функциональная модель продукта, незадокументированный код увеличивают время исправления каждой ошибки в десятки раз.

Попробуем более детально разобраться в причинах, которые зачастую не позволяют реализовать приведенную «идеальную» модель в реальных проектах.

Вспомним три постулата ООП: наследственность, полиморфизм и инкапсуляция данных. Это не что иное, как структурные требования. Как уже отмечалось выше, средствами объектно-ориентированных языков очень сложно описать поведение программы, в то время как именно к поведению выдвигаются основные требования. Поэтому несколько искусственно выглядят модели поведения программ на этапе проектирования – гарантировать точное соответствие поведения реальной программы и ее модели могут только программисты. Даже если архитектура программы составлена корректно, на этапе кодирования она может быть неточно перенесена в программу из-за того, что программисты используют средства ООП. В этом случае возникают дополнительные временные затраты на тестирование – проверку соответствия программы той схеме, что была спроектирована. Таким образом, помимо проверки соответствия программного продукта требованиям, приходится проводить дополнительное тестирование для проверки соответствия кода спроектированной модели, что на самом деле является еще одним недостатком используемой технологии ООП.

Кроме дополнительных затрат на тестирование, возникает необходимость дополнительного документирования кода. Ведь разработанную архитектуру нельзя считать полноценной документацией, так как она отражает лишь желаемую модель, которой *должен* соответствовать итоговый продукт. Важно подчеркнуть, что соответствие продукта архитектурной модели является негласным требованием и также нуждается в тестировании и документировании.

Здесь может быть скрыта еще одна проблема. В теоретической модели процесса создания документации происходит лишь после того, как написан код и проведено тестирование. Однако документация необходима не только заказчику продукта, но и программистам, которые впоследствии будут поддерживать выпущенный продукт. Соответственно, необходимо, чтобы документация охватывала все стадии разработки от сбора требований до тестирования. Поэтому на практике проектная документация обычно накапливается в течение всего цикла разработки. Таким образом, помимо необходимости обеспечить соответствие модели программы и ее реального кода, необходимо также поддерживать единство документации, создаваемой в ходе разработки.

Для этого разработчики на каждом этапе стараются переиспользовать ту часть документации, что была создана ранее. Например, на этапе кодирования необходимо создавать лишь ту часть документов, которые описывают существенные аспекты реализации, но не затрагивают поведенческую модель, разработанную и задокументированную на этапе проектирования.

Это опять же, является новым требованием, которое по своей сути является очередным недостатком технологии ООП. Невыполнение этого требования обычно приводит:

- либо к полному несоответствию документов на архитектуру и на код, что, в свою очередь, выливается в удвоенные затраты на документирование модели поведения программы (первый раз – на этапе проектирования, второй раз – на этапе кодирования, чтобы обеспечить полноту документации);
- либо к необходимости реструктуризации кода с точки зрения требований к документации (выглядит опять же несколько искусственно);
- либо к дополнительным, часто даже не планируемым, затратам на этапе поддержки.

Таким образом, сокращение общего цикла разработки программного продукта невозможно за счет сокращения хотя бы одной из его фаз – обязательно пострадает сам продукт. С другой стороны, из-за недостатков используемой технологии ООП появляются дополнительные временные затраты, которых можно избежать, выбрав в качестве основы процесса разработки технологию автоматного программирования [11].

Автоматное программирование – хорошо забытое старое

Традиционное место использования автоматов в классическом процессе разработки программного обеспечения – этап проектирования, когда создается графическое представление модели программы в виде конечного автомата. Это позволяет, во-первых, явно определить все состояния, в которых может находиться программа, во-вторых, описать входные и выходные воздействия, в-третьих, при необходимости провести формальную верификацию модели программы, что, в свою очередь, позволит сократить время тестирования конечного продукта, а также повысить надежность всей системы. Впервые явно выделять состояния программы было предложено в языке *SDL (Specification and Description Language, ITU-T Recommendation Z.100)*, который до сих пор является признанным стандартом описания телекоммуникационных протоколов [12].

Возможно, именно из-за того, что автоматный подход используется в основном для описания модели поведения программы, большинство программистов ассоциирует термин «конечный автомат» именно с «картинкой», представлением, которым следует руководствоваться при написании кода программы, используя современные (объектно-ориентированные) языки программирования. Именно из-за этого возникают сложности с сокращением цикла разработки, о которых упоминалось выше.

Для того чтобы та или иная технология программирования нашла свое отражение в виде признанного промышленного процесса, необходимо как минимум три вещи: строгая математическая модель, наглядное (желательно графическое) представление этой модели – нотация и инструментальное средство разработки программ в заданной нотации [13]. При этом необходимо обеспечить согласованность всех трех составляющих: если уж в качестве математической модели широко используется конечный автомат, разработаны и стандартизованы графические средства представления программ с использованием автоматов, то остается выполнить последний шаг – перенести автоматы в программы, или, точнее, писать программы «в автоматах».

Отметим, что существуют инструментальные средства, позволяющие из описания программы (с использованием диаграмм состояний), получить программный код на выбранном целевом языке программирования. Чаще всего недостатком таких средств является их ориентированность на использование в рамках сложившегося процесса разработки, а следовательно привязанность к определенным языкам программирования [14].

В [15] полностью описан технологический процесс создания программного продукта при использовании *технологии автоматного программирования* совместно с существующими языками программирования. Преимущества использования автоматного подхода при программировании очевидны:

- единство поведенческой модели программы, разработанной на этапе проектирования, и самой программы фактически избавляет от необходимости тестирования кода на соответствие архитектуре. Более того, архитектура программы уже является не просто представлением – это управляющий автомат, целиком и полностью задающий поведение программы;
- исчезает необходимость документирования кода в части поведения: если код по сути является управляющим автоматом, исчерпывающей документацией является сам автомат;
- появляется возможность повысить надежность конечного продукта за счет верификации автомата, задающего логику работы системы.

Таким образом, использование технологии автоматного программирования позволяет устранить существующие недостатки, описанные выше. Однако остается открытым вопрос использования ООП совместно с автоматным подходом. В [15], например, предлагается применять автоматный подход в качестве *дополнения* к существующей технологии ООП. В таком случае необходимо, чтобы автомат стал основной базовой конструкцией языка программирования, чтобы обеспечить *изоморфность* программы и диаграммы состояний на протяжении всего цикла жизни программного продукта, особенно в процессе поддержки, когда код может быть неоднократно модифицирован.

Использование автоматов на этапе программирования фактически упраздняет многие языковые конструкции современных языков программирования. Более того, в [14] отмечается, что модель программы в виде конечного автомата является исполнимой, что, по сути, избавляет от необходимости построения кода, так как программа является *интерпретацией* автоматной модели. Функциональная нагрузка на язык программирования, таким образом, сводится к описанию входных и выходных воздействий на автомат. Для этого программистам совершенно необязательно иметь весь тот инструментарий, который предоставляют современные языки программирования.

Упрощение языка программирования и превращение его в *язык описания входных и выходных воздействий* позволит также избавиться еще от одного существенного недостатка объектно-ориентированных языков – неэффективного использования компьютерных ресурсов. Компилятор программ превращается в таком случае в интерпретатор автоматной модели, снабженный *сопроцессором*, позволяющим выполнять выходные действия, которыми помечен автомат, а единственным представлением программы остается конечный автомат.

Описанная технология программирования, также как и технология автоматного программирования, базируется на понятии конечного автомата как основного элемента, явно определяемого на всех этапах разработки программного продукта. Но существенным отличием является отказ от процедуры кодогенерации на этапе программирования, а также сведение программирования как такового к описанию входных и выходных действий автомата на специальном языке программирования. Вместе с тем предлагаемая технология, вследствие отсутствия явного определения понятий «объект» и «отношение», является также альтернативой технологии ООП.

Заключение

Современные методы и технологии программирования являются вершиной долгого эволюционного процесса, в основе которого лежит желание сократить сложность программирования, внедряя новые уровни абстракции. Программисты тем самым освободились от необходимости детального изучения архитектуры платформы, для которой они пишут код, появились мощные средства разработки, сформировались четкие технологические процессы. Вместе с тем на первый план выходят требования к качеству итогового продукта, а также ко времени его разработки. И в свете этих требований

всплывают неочевидные до сих пор недостатки используемых методов и технологий программирования.

В данной работе предпринята попытка анализа этих недостатков, а также способов выхода из сложившейся ситуации, которые могли бы существенно изменить структуру современного технологического процесса.

Предлагается использовать концепцию автоматного программирования на всех этапах разработки программного продукта, что позволит выдвинуть на первый план поведение разрабатываемого приложения. Таким образом, единственным представлением программы будет являться конечный автомат, а задача программиста сведется к описанию входных и выходных воздействий на специальном, сильно упрощенном, языке программирования.

В отличие от существующей технологии автоматного программирования [2], предлагается также отказаться от процедуры генерации программного кода из автоматного описания, введя понятие интерпретатора автоматных программ. Таким образом, поведенческая модель программы становится исполнимой и требует лишь описания входных и выходных воздействий.

Представление программы в виде конечного автомата будет использоваться на этапе разработки архитектуры, когда будет разрабатываться основная логика поведения программы, определяться входные и выходные воздействия, проводится формальная верификация модели. Это же описание будет использоваться на этапе кодирования и оно же в итоге будет являться исполнимой программой.

Предлагаемая технология обладает всеми преимуществами технологии автоматного программирования и вместе с тем устраняет многие недостатки ООП, такие как, например, неэффективное использование компьютерных ресурсов.

Литература

1. Грэхем И. Объектно-ориентированные методы. Принципы и практика. 3-е изд. – М.: Издательский дом «Вильямс», 2004. – 880 с.
2. Шалыто А.А. Технология автоматного программирования / Труды Всероссийской научной конференции «Методы и средства обработки информации» – М.: МГУ. 2003. – с.528 – 535.
3. Sklenar J. Introduction to OOP in Simula / Proceedings of 30 Years of Object Oriented Programming. – University of Malta. 1997.
4. Грэм П. Языки программирования через сто лет. // Компьютерра-онлайн, 08.2004. – Режим доступа: <http://www.computerra.ru/hitech/35042/>
5. Moessenboeck H. Object-Oriented Programming in Oberon-2 – Springer-Verlag. 1993. – Режим доступа: <http://www.uni-vologda.ac.ru/oberon/infoart/plus&min.htm>
6. Thomas E. Potok, Mladen Vouk, and Andy Rindos. Productivity Analysis of Object-Oriented Software Developed in a Commercial Environment. //Software – Practice and Experience. – 1999. – № 10. – pp. 833–847.
7. M. Paulk, C. Weber, B. Curtis, M. Chrissis, et al. The Capability Maturity Model. – Addison-Wesley. 1995.
8. CMMI for Development. Version 1.2 / Carnegie Mellon Software Engineering Institute. – August 2006. ESC-TR-2006-008. – Режим доступа: <http://chrGUIbert.free.fr/cmmi12/cmmi-dev/text/index.php>
9. Askar Rahimberdiev «Современные процессы разработки программного обеспечения». // RSDN Magazine. – 2006. – №4.
10. Макконнелл С. Профессиональная разработка программного обеспечения. – СПб: Символ-Плюс. 2007. – 240 с.

11. Шалыто А.А., Тукель Н.И. SWITCH-технология – автоматный подход к созданию программного обеспечения «реактивных» систем. //Программирование. – 2001. – №5. – С. 45–62.
12. Гольдштейн Б.С. Сигнализация в сетях связи. 3-е издание. – М.: Радио и связь, 2001. – 446 с.
13. Тукель Н.И., Шалыто А.А. Программирование с явным выделением состояний. // Мир ПК. – 2001. – №8. – С. 116–121, №9. – С. 132–138.
14. Гуров В., Нарвский А., Шалыто А. Исполняемый UML из России // PC Week, Russian Edition. – 2005. – № 26. – С. 18–19.
15. Шалыто А.А. SWITCH-технология. Алгоритмизация и программирование задач логического управления. – СПб: Наука, 1998. – 628 с.

ПРОГРАММИСТЫ, КОМПИЛЯТОРЫ, ПРОЦЕССОРЫ – ПОИСК ЕДИНОГО ВЕКТОРА

**Е.Б. Боб, А.В. Латников, А.М. Мальцев, А.Е. Потехин
Научный руководитель – д.т.н., профессор А.А. Шалыто**

В статье приводится анализ недостатков современных подходов к программированию с точки зрения повышения производительности. Предлагается использование технологии автоматного программирования с поддержкой этой технологии на уровне «железа».

Введение

В данной статье рассматриваются современные подходы к программированию и созданию микропроцессоров. Порочная практика создания программного обеспечения привела к тому, что ресурсы производительности микропроцессоров каждого нового поколения сводятся на нет уже к моменту их появления. Каким бы быстрым не было «железо», программисты всегда находят способы растратить новые мощности практически впустую, за редким исключением узкоспециализированных программ (обработка видео, аудио и т.п.). Ситуация усугубляется с появлением многоядерных систем. Авторами предлагается использовать технологию автоматного программирования [1] с поддержкой автоматов на уровне «железа», что позволит повысить производительность систем и упростить распараллеливание программ.

Программное обеспечение и производительность компьютерных систем

Бурное развитие микропроцессорной техники в последнее десятилетие воспринимается обывателем как гигантский скачок в мире высоких технологий. Каждое новое поколение микропроцессоров удивляет нас своими возможностями. Не отстают и средства разработки программного обеспечения. Сегодня рядовым программистам уже необязательно знать тонкости «железа», что делает их труд более универсальным. Однако у каждой медали есть и обратная сторона. Подобное отстранение не может не сказаться на производительности программного обеспечения. На протяжении долгого времени считалось, что нововведения в микропроцессорах должны с лихвой компенсировать этот недостаток. А действительно ли современные микропроцессоры настолько уникальны?

«Мы гордимся современной аппаратурой и пренебрежительно относимся к достижениям двадцати-тридцатилетней давности, между тем это ослиная гордость. Чтобы там ни говорила реклама, невозможно не признать, что за последнее время ничего принципиально нового не придумано. Эксплуатируется сравнительно небольшое количество весьма древних идей и, если что и совершенствуется, – так это проектные нормы. То, что раньше занимало целый шкаф, теперь свободно умещается на ладони. Нет ничего такого, что в том или ином виде не присутствовало бы в «динозаврах» первых поколений» [2].

Миф о всемогущих микропроцессорах сыграл с нами злую шутку. За несколько десятилетий не было создано ничего принципиально нового, а программистов продолжали учить по-старому. Все новшества средств разработки программного обеспечения – это не что иное, как надстройки над старыми технологиями. Авторы статьи уверены, что настало время обратиться к истокам и пересмотреть свой взгляд на очень многие вещи.

Возможен ли вечный рост?

Выведенный более сорока лет назад закон Мура работает, и по сей день. Одна из его формулировок гласит, что производительность микропроцессоров удваивается ка-

ждые 18 месяцев. Фактически наблюдается экспоненциальный рост производительности микропроцессоров на протяжении почти что полувека! Возникает вопрос: а может ли такой рост продолжаться вечно? Профессор экономики Кеннет Болдинг однажды так высказался по поводу подобного положения вещей: «тот, кто верит, что экспоненциальный рост на земле может продолжаться вечно, либо сумасшедший, либо экономист» [3]. Хотя это высказывание относилось к экономическому росту, оно легко может быть применено и к росту производительности компьютерных систем. Достаточно только заменить слово «экономист» на «программист». Сегодня разработчики программного обеспечения мало задумываются о производительности как таковой, свято веря в то, что в будущем их программы будут работать быстрее за счет более производительного «железа». Производителям же микропроцессоров с каждым годом становится все труднее преодолевать технические барьеры и увеличивать рост производительности в соответствии с законом Мура. Если посмотреть на сам термин «производительность», то можно заметить, что его смысл существенно изменился за последние годы. Многие модные новшества в микропроцессорах имеют значение при обработке потоковых данных (звук, видео), но мало сказываются на обычных «поведенческих» программах. Более того, скорость работы последних, как правило, уменьшается. Только за счет повышения тактовой частоты это незаметно [4]. Дальнейшее повышение тактовой частоты становится дорогим и, следовательно, нецелесообразным с экономической точки зрения. Да и технический порог уже не за горами. Появившиеся в последнее время многоядерные архитектуры – это, по логике вещей, шаг в верном направлении. Однако и здесь все не так просто: увеличение числа ядер в N раз не увеличивает производительность во столько же раз. Более того, прирост производительности падает, а сложность системы растет при добавлении в систему каждого нового ядра, и оба эти процесса являются экспоненциальными [5]. В основном это происходит из-за необходимости организации обмена и синхронизации данных при выполнении программы. Современные средства программирования не позволяют решать подобные задачи, оставляя их на откуп производителям «железа». При этом программисты живут в своем «линейном» мире и стараются абстрагироваться от такого рода проблем. Сегодня предпринимаются попытки ввести в языки программирования специальные директивы для сближения программистов и аппаратчиков [6]. Авторы статьи уверены, что эти попытки обречены на провал – слишком уж большая пропасть образовалась между разработчиками программного обеспечения и производителями микропроцессоров. Подходы к разработке программного обеспечения не пересматривались уже несколько десятков лет, и попытки их подстройки неизбежно приведут к неудаче. Здесь требуются качественные изменения, главным из которых станет отход от программирования с использованием флагов. Иначе получается все как в известной басне И.А. Крылова «Лебедь, Рак и Щука», где в роли главных героев выступают программисты, компиляторы и микропроцессоры. А можно ли направить их усилия в одном направлении?

Программисты

Программисты – кто они такие? На первый взгляд, вопрос кажется бессмысленным – это люди, пишущие код. К большому сожалению, подавляющее большинство людей в мире информационных технологий именно так и считает. Любой человек, выучивший синтаксис какого-либо языка программирования, сегодня гордо называет себя программистом. А программист-профессионал, по мнению многих, это *человек, написавший код так, что его никто, кроме него, не понимает*. На самом деле, это не программисты, а кодировщики, в первом случае, и плохие кодировщики – во втором. Задача программистов – это разработка поведенческой модели. Сегодня же их усилия в основном тратятся на борьбу с неправильными указателями, утечками памяти и т.п. На-

стоящему программисту, по большому счету, абсолютно неважно, на каком языке разрабатывать программу, так как его задача – разработать алгоритм в соответствии с требованиями, руководствуясь которыми кодировщики будут «кодировать».

Существует процесс разработки программного обеспечения, который и предусматривает подобный подход: технические эксперты разрабатывают дизайн на основе технического задания, а «программисты» пишут код в соответствии с дизайном. Все абсолютно верно – дизайн соответствует требованиям, код – дизайну. Однако есть нюанс, ломающий эту идеалистическую модель – человеческий фактор. Технический эксперт, как правило, разрабатывает дизайн на «бумаге», а «программист» переносит его в код, руководствуясь своими *домыслами*. Причем основной целью дизайна является описание поведения системы. Даже если предположить, что дизайн полностью соответствует требованиям, нельзя утверждать, что код также будет им соответствовать. Просто потому, что «программист» – это человек, а человеку свойственно ошибаться. Исправить подобное положение вещей призвано дорогостоящее «двойное» тестирование. Термин «двойное» означает, что одновременно тестируются и дизайн, и код, что не лучшим образом сказывается на конечном качестве продукта – как на его поведении, так и на отказоустойчивости.

Компиляторы и микропроцессоры

Компиляторы облегчают жизнь разработчикам программного обеспечения, так как изолируют последних от аппаратных особенностей той или иной системы, делая программистов независимыми от «железа». В последнее время компиляторы стали достаточно «умными» – они учитывают специфические особенности микропроцессоров при оптимизации размера кода, быстродействия и т. п. Но столь ли они всемогущи? Нет – компиляторы не в состоянии исправлять такие упущения программистов, как неоптимальное использование ресурсов, неверную реализацию алгоритмов и т.п. Таким образом, можно утверждать, что разработчики компиляторов в основном ориентируются на особенности микропроцессоров. Однако и здесь все не так идеально, как хотелось бы. Ведь код, генерируемый компилятором, и код, исполняемый ядром микропроцессора – зачастую абсолютно разные сущности. Дело в том, что многие микропроцессоры транслируют коды команд программы в свое внутреннее представление.

Теперь начинается самое интересное – подходим к сердцу системы. Что знают рядовые программисты о современных микропроцессорах? Микропроцессор выполняет программный код, обращаясь к памяти и устройствам ввода/вывода. Причем память линейная, скорее всего виртуальная, и ее объем практически неограничен. И сегодня этих сведений оказывается вполне достаточно для написания программы. Это отчасти так, если создается некое приложение, которое не критично ко времени исполнения. Однако, в конце концов, рядового пользователя начинают раздражать «тормозящие» программы, которые съедают к тому же львиную долю дискового пространства. Что уж говорить о серьезных встроенных системах!

Можно было бы пожалеть бедный микропроцессор – ведь ему приходится перерабатывать столько неоптимального кода, используя столь хитроумные и сверхсложные механизмы: кэширование, предсказание переходов, переупорядочивание команд на конвейере и т. п. Однако сам микропроцессор и его окружение (чипсет, память) – это часть проблемы. Архитектура микропроцессора рассчитана на одиночное плавание – процессор не принимает никаких управляющих директив программиста, кроме непосредственно кода. И даже о том, что программисту абсолютно очевидно, микропроцессор не имеет ни малейшего представления.

Что оказывает наибольшее влияние на производительность современного микропроцессора? Количество ядер? Тактовая частота? Нет! Самое существенное влияние оказыва-

ют программные переходы и обращения к памяти, которая, несмотря на все ухищрения, работает гораздо медленнее ядра микропроцессора. Представим себе конвейерное производство – четкий и отлаженный механизм. Ни одного лишнего движения – каждая деталь доставляется со склада и подается на конвейер в то самое время, когда она необходима. Рабочие (роботы) не простаивают ни секунды – каждый хорошо проинструктирован (запрограммирован) на выполнение определенной задачи. Вдруг, ни с того, ни с сего, со склада начинают доставлять абсолютно другие детали. Рабочие не знают, что с ними делать – конвейер останавливается. То, что находилось на конвейере в этот момент, идет в утиль. Примерно то же самое происходит и в микропроцессоре. При любом программном переходе конвейер команд должен быть очищен и перезагружен новыми командами. Другая ситуация – когда закрывают один из складов, и детали к конвейеру приходится везти из соседнего города. Это похоже на обращение к новой области памяти. Если прочитать, например, пособие по оптимизации обращений к памяти для конкретного микропроцессора/чипсета и попытаться написать программу в соответствии с рекомендациями, то рядовой программист просто не сможет ее понять. Смешно, но изощренные алгоритмы, которыми забивают головы сегодняшним студентам, проигрывают по быстрдействию простым реализациям, учитывающим особенности микропроцессора/чипсета. И проигрывают в разы, а всемогущие компиляторы и здесь оказываются бессильными [2].

Три кита

Как уже говорилось выше, для серьезного технологического прорыва требуются качественные и согласованные изменения, затрагивающие все стадии процесса разработки программного обеспечения и аппаратную архитектуру в целом. Как превратить героев басни И. А. Крылова в трех китов, которые смогут стать основой компьютерных систем будущего?

Во-первых, необходимо избавиться от таких архаизмов, как указатели, ручное выделение и освобождение памяти, переходов и т.п. Можно спросить, а как же тогда писать программы? При тех подходах, которые существуют сегодня – никак. Но в том-то и дело, что сами эти подходы требуется изменить. Предположим, что программа разбита на две части: логическую (поведенческую) и исполняемую (простейшие обработчики, из которых вынесена вся логика) [7]. «Высокоуровневые» программисты должны сосредоточиться на логической, а «низкоуровневые» – на исполняемой части. Первая часть будет фактически дизайном – его можно проверить, так как поведение системы задается требованиями. Во второй же части логика вообще отсутствует, а соответственно нет и переходов – такой код очень легко тестировать, особенно если ручное выделение и освобождение памяти запрещено, а указатели отсутствуют.

Во-вторых, функции компилятора – собрать обе части программы. Вдобавок компилятор должен проанализировать, какие ресурсы требуются тому или иному обработчику, и добавить соответствующее описание в программу. И все – никаких ухищрений (оптимизаций и т.п.).

В-третьих, аппаратная часть должна полностью управляться логической частью программы – в этом случае переходы будут известны заранее, и можно исключить эвристический предсказатель из системы. Сердцем такой системы станет «поведенческий» процессор – он будет реагировать на внешние (внутренние) воздействия и, работая в соответствии с логической частью программы, выбирать и передавать код исполняемых модулей (обработчиков) различным исполнительным устройствам. Причем выбор конкретного исполнительного устройства будет осуществляться в соответствии с описаниями, подготовленными компилятором. При таком подходе работа конвейера команд станет управляемой, а выборка памяти для конечных обработчиков будет делаться заранее. *Фактически код программы будет формироваться на лету.*

То, что было описано выше, является ничем иным, как *автоматным программированием*. Позвольте, скажете вы, но автоматное программирование применяется программистами уже сегодня! И да, и нет. Дело в том, что сегодня автоматы являются частью программы. Предлагаемый же подход представляет собой не программно-аппаратный комплекс, а автоматнo-аппаратный комплекс, в котором автоматы являются единственным возможным методом создания программ. Автоматы должны заменить операционные системы: драйвера уйдут в прошлое – внешние устройства будут напрямую воздействовать на автоматы. Сервисы операционной системы должны быть стандартизованы и написаны на автоматах, равно как и все другие сервисы. В этом случае необходимую конфигурацию операционной системы (да и всей системы в целом) можно будет просто собирать как из кубиков. Если вдобавок к этому унифицировать исполняемый код обработчиков, то коэффициент переиспользования будет очень высоким.

Автоматное программирование дает неоспоримое преимущество при написании поведенческих программ: упрощается как понимание алгоритма, так и сопровождение программного продукта. Сегодня дизайнеры сложных систем все равно в той или иной степени используют автоматный подход при проектировании. Однако при компиляции все преимущества автоматов растворяются: микропроцессор уже не может оперировать логикой состояний для предсказания переходов. Предлагаемая концепция автоматнo-аппаратного комплекса позволяет донести логику работы программы до «железа» в неизменном виде. При таком подходе эвристические механизмы не будут требоваться. Новое «железо» сможет работать в полном соответствии с логикой программы, заранее выбирая те области памяти, которые ему необходимы.

В настоящее время авторы статьи работают над синтаксисом языка описания автоматов и прототипом автоматного компилятора, способного разделять «логическую» и «исполняемую» части программы. На первом этапе на выходе компилятора предполагается получать программный код, в котором логическая часть будет интегрирована с исполняемой. Этот код будет использоваться для проработки технологии и устранения выявленных недостатков. В дальнейшем планируется создать программные модели «логического» и «исполнительного» процессоров – прототип автоматного «железа». На этом прототипе описанная технология будет проверена в полном объеме.

Заключение

Современные микропроцессоры и средства разработки программного обеспечения в результате своего развития оказались оторванными друг от друга. Компиляторы, призванные объединить эти два полюса, справляются со своей задачей недостаточно эффективно. С появлением многоядерных архитектур эффективность существующего подхода снижается еще больше. В данной работе предпринята попытка анализа противоречий, а также способов выхода из сложившейся ситуации. Предлагается использовать концепцию автоматного программирования с поддержкой автоматов на уровне «железа». Несмотря на то, что эта концепция требует полного пересмотра процесса разработки программного обеспечения и микропроцессоров, такой подход оказывается наиболее оптимальным с точки зрения увеличения производительности и распараллеливания вычислений.

Литература

1. Шалыто А.А. Технология автоматного программирования. / Труды Всероссийской научной конференции «Методы и средства обработки информации» – М.: МГУ. 2003. – С. 528–535.
2. Касперский К. Техника оптимизации программ. Эффективное использование памяти. – СПб.: БХВ-Петербург. 2003. – 464 с.

3. Википедия [Электронный ресурс], Режим доступа: http://en.wikipedia.org/wiki/Kenneth_Boulding, свободный. – Загл. с экрана. – Яз. англ.
4. The Microarchitecture of the Pentium 4 Processor / Intel Technology Journal Q1, 2001
5. Amdahl G. Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities. / AFIPS Conference Proceedings. – 1967. – P. 483–485.
6. OpenMP [Электронный ресурс], Режим доступа: <http://www.openmp.org/blog/>, свободный. – Загл. с экрана. – Яз. англ.
7. Шалыто А.А. Логическое управление. Методы аппаратной и программной реализации. – СПб: Наука, 2000. – 784 с.

ПОСТРОЕНИЕ УПРАВЛЯЮЩЕГО ГРАФА ПРОГРАММЫ, СОДЕРЖАЩЕЙ НЕЯВНЫЙ ПОТОК УПРАВЛЕНИЯ

А.М. Карпушинский

Научный руководитель – к.т.н., доцент Т.А. Павловская

Рассмотрены проблемы тестирования программ, написанных на объектно-ориентированном языке и содержащих неявный поток управления, обусловленный присутствием конструкций обработки исключений. Предложены алгоритмы построения управляющего графа таких программ. Приведено пошаговое описание алгоритма построения внутрипроцедурного и межпроцедурного управляющего графа.

Введение

Современные технологии производства программного обеспечения предусматривают повсеместное использование объектно-ориентированных языков программирования (C++, C#, Java). Наряду с большими преимуществами таких языков, некоторые их возможности сопряжены с определенными трудностями, возникающими у разработчиков. Так, основная проблема, связанная с использованием полиморфизма и механизма обработки исключений, состоит в том, что они вводят (*implicit control flow*) – поток управления, не очевидный в исходном тексте [1, 2]. Из-за сложности, привносимой неявным потоком управления в объектно-ориентированный код, разработчики могут упустить важные взаимодействия в программах. Например, в процессе продвижения исключения из вызванного метода до вызывающего через несколько промежуточных методов, цепочка вызовов может создать непреднамеренные зависимости данных в программе. В результате, если не учитывать наличие неявного потока управления, эффективность разработки и сопровождения программного кода может снизиться.

Привнесение обработки исключений в исходный код также увеличивает трудоемкость процесса тестирования. Структурное тестирование проводится по принципу «белого ящика», т.е. основано на знании внутренней структуры программы, и характеризуется степенью, в которой тесты покрывают логику программы. При структурном тестировании программы необходимо определить критерий тестирования. Затем представить исходный код в виде некоторой структуры. Логике программы принято представлять в виде графовой модели – *управляющего графа (УГ)*. Следующим этапом структурного тестирования является выделение путей в графе согласно выбранному критерию с целью обеспечения полного покрытия выделенных элементов (операторов, ветвей, путей).

Теоретическая часть

Рассмотрим вкратце механизм обработки исключений в языке Java. Для обработки исключений используются *конструкции try*, состоящие из обязательного блока *try* и необязательных блоков *catch* и *finally*. Блок *try* содержит исполняемый код, в котором могут выбрасываться исключения с помощью оператора *throw*, за ним следует один или несколько блоков *catch* или блок *finally*, либо и блоки *catch*, и *finally*. Блок *try* выполняется либо полностью, либо пока не выбросится исключение. Если в процессе выполнения команд блока *try* выбрасывается исключение определенного типа, управление передается в соответствующий этому типу блок *catch*, затем оператору, следующему за блоком *try*. Если для соответствующего исключения не было найдено ни одного блока *catch*, оно выбрасывается на блок выше и далее согласно стеку вызовов, пока не найдется соответствующий этому исключению обработчик. Если определен блок *finally*, он выполняется после блока *try*, с которым связан. При этом передача управления в блок *finally* осуществляется как в случае нормального завершения выполнения блока *try* (нормальный контекст), так и в случае возникновения исключения (исключительный

контекст). На рис. 1 представлены различные варианты передачи управления при обработке исключений.

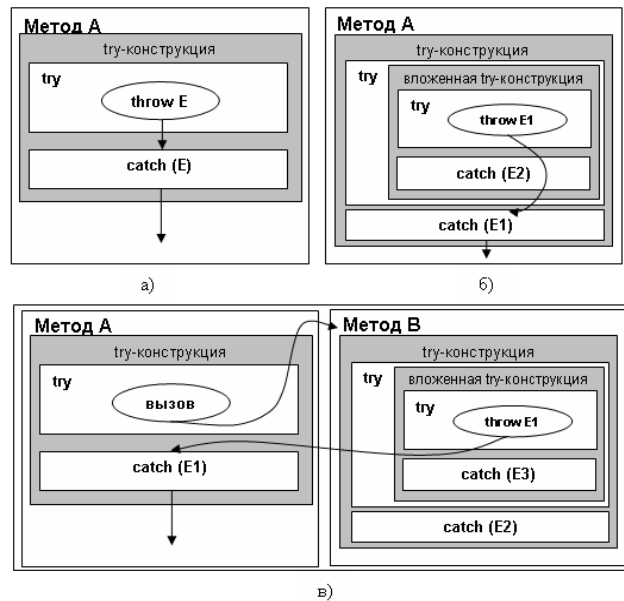


Рис. 1. Передача управления при обработке исключений: в случае одиночной try-конструкции (а), в случае вложенной try-конструкции (б), в случае продвижения исключения (в)

Проблему получения информации о потоке управления можно разделить на два уровня: анализ *внутрипроцедурного* и *межпроцедурного* потоков управления. На уровне локального анализа извлекается информация о потоке управления для каждого метода в отдельности (принимаются во внимание локальные передачи управления в рамках одного метода, такие как continue). На уровне глобального анализа извлекается информация о потоке управления между взаимодействующими методами (учитываются такие передачи управления, как выход, возврат). Также необходимо учитывать неявный поток управления, обусловленный механизмом обработки исключений. В соответствии с этим разработаны следующие алгоритмы: алгоритм построения УГ метода, алгоритм вывода типов для определения типов исключений, выбрасываемых оператором, и алгоритм построения межпроцедурного УГ программы.

Алгоритм построения УГ метода

Определение. Управляющий граф G метода M – это направленный граф (N, E, s, e) , где N – множество вершин, E – множество дуг, s, e – начальная и конечная вершины. Вершины УГ соответствуют операторам либо блокам операторов, а дуги – всевозможным передачам управления между операторами (блоками).

С учетом конструкций механизма обработки исключений УГ должен содержать вершины, представляющие оператор throw и блоки catch и finally, а также дуги, представляющие нормальную и исключительную передачу управления. В таком УГ вершина throw может иметь несколько вершин-детей. Количество вершин определяется числом типов исключений, которые могут быть выброшены соответствующим оператором throw. Для определения всех возможных типов исключений используется метод вывода типов и каждому типу сопоставляется одна выходящая из throw-вершины дуга. В УГ вершина, представляющая блок catch, который обрабатывает явно выброшенные исключения, имеет входящие дуги для этих исключений. Однако вершина, представляющая блок catch, который обрабатывает только неявно

выброшенные исключения, не имеет входящих дуг. Для такой вершины входящие дуги определяются на уровне анализа межпроцедурного потока управления. На рис. 2 представлен пример УГ метода.

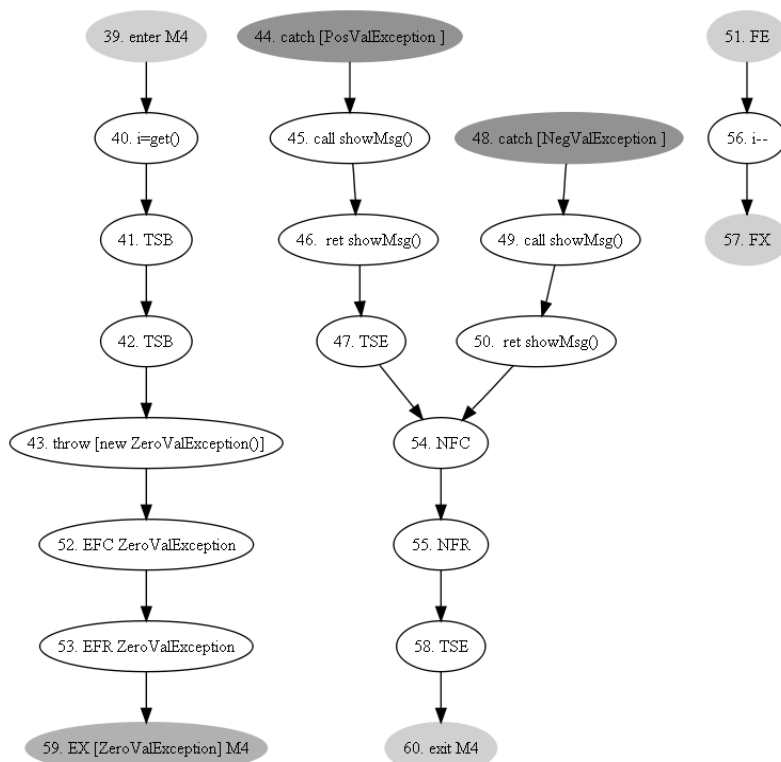


Рис. 2. Пример управляющего графа метода

Алгоритм построения УГ для метода М предназначен для построения УГ из абстрактного синтаксического дерева (АСД) этого метода и сбора информации о методе М, необходимой для построения межпроцедурного представления программы, содержащей М. Входными данными для алгоритма является АСД метода, выходными – УГ метода. В описании алгоритма не приводится этап построения вершин для общих конструкций языка (ветвления, циклы, и т.п.).

В УГ программы, содержащей обработку исключений, будем соединять вершину throw с каждым обработчиком catch, который может обработать выброшенное исключение. Чтобы определить условия перехода к конкретному обработчику, необходимо:

- 1) определить все возможные типы исключений, которые могут быть выброшены при выполнении каждого оператора;
- 2) так как исключение может быть обработано вне метода, в котором оно было выброшено, вершина throw может не иметь выходящих дуг; такие дуги появляются, когда УГ для всех методов соединяются в один межпроцедурный УГ.

Алгоритм построения УГ для метода предусматривает создание специальных вершин для каждого оператора try, throw и каждого блока catch. Для каждого блока finally, встречающегося в методе, строится отдельный УГ со своим входом и выходом, а также создаются вершины-вызовы этого блока в УГ метода. Каждая вершина throw соединяется с вершиной catch, если типы выбрасываемого и обрабатываемого исключений являются подходящими [1].

Присутствие в коде блоков finally увеличивает сложность путей управления, так как он может выполняться в двух разных контекстах: в нормальном и исключительном.

Для каждого блока finally строится отдельный УГ, и УГ метода дополняется вершинами - вызовами для обоих контекстов выполнения (см. рис. 2).

Алгоритм состоит в рекурсивном обходе вершин АСД. Действия, выполняемые алгоритмом, основаны на типе очередной вершины АСД. Обобщенная блок-схема алгоритма представлена на рис. 3.

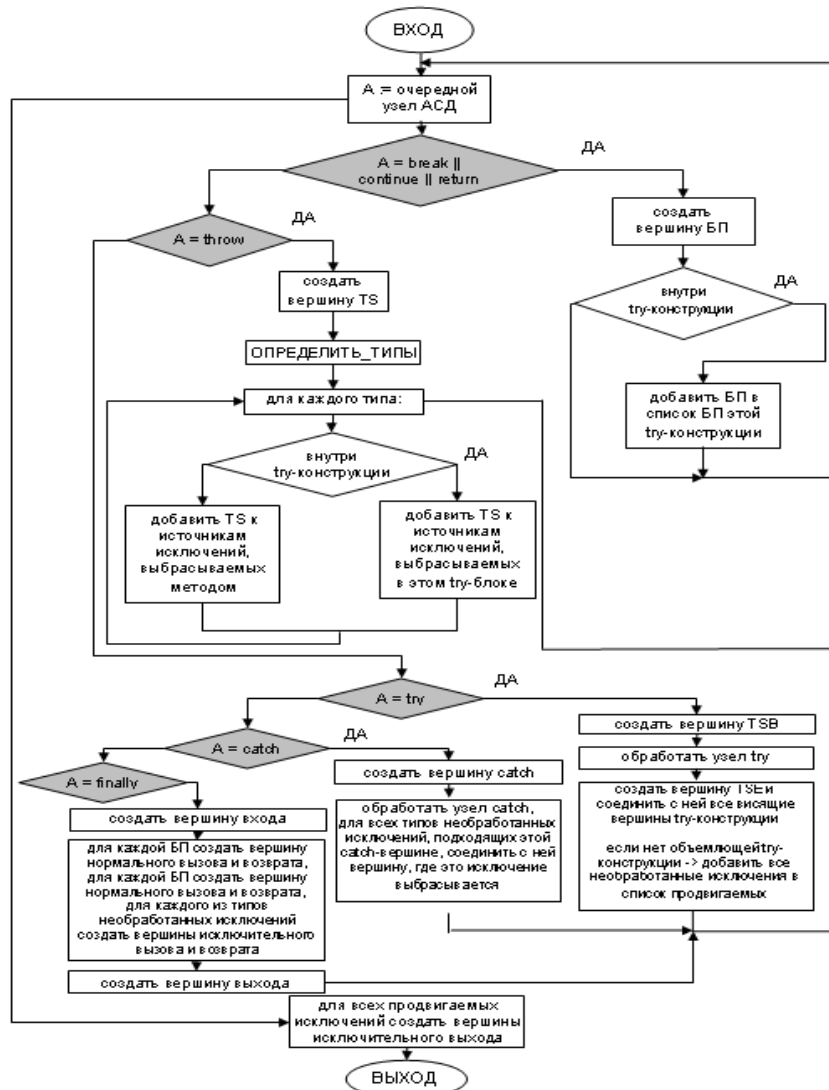


Рис. 3. Общая схема алгоритма построения УГ метода

Алгоритм построения межпроцедурного УГ

Межпроцедурный УГ программы P содержит модифицированные УГ для каждого метода из P, которые соединены в каждой точке вызова при помощи дуг вызова и возврата: *дуга вызова* соединяет вершину вызова с вершиной входа вызванного метода, а *дуга возврата* соединяет вершину выхода вызванного метода с соответствующей вершиной возврата. Продвижение исключения в стеке вызовов из вызванного метода в вызывающий обуславливает появление дополнительных межпроцедурных дуг между УГ методов, отличных от обычных дуг вызова и возврата.

Межпроцедурные дуги, соединяющие вершину исключительного выхода вызванного метода и catch-вершину, вершину вызова finally-блока или вершину исключительного выхода из вызывающего метода, назовем *дугами исключительного возврата*.

Другим типом межпроцедурных дуг, которые встречаются в межпроцедурном УГ, являются дуги, обусловленные выделением отдельного УГ для каждого finally-блока: если блок содержит оператор безусловного перехода, то управление передается вовне finally-блока. *Дуга безусловного перехода* соединяет вершину, представляющую безусловный переход, с вершиной, представляющей оператор, которому осуществляется передача управления.

Алгоритм построения межпроцедурного УГ состоит из следующих действий.

1. Создать список всех методов программы.
2. Для каждого метода N найти все методы, вызывающие этот метод. Для каждого такого метода M выполнить п.п. 3, 4, 5, 6.
3. Создать дуги вызова N, возврата и исключительного возврата из N.
4. Создать вершины исключительного входа и выхода в finally-блоки для каждого типа исключений, продвигаемых из N.
5. Создать вершины и дуги исключительного выхода.
6. Создать дуги безусловного перехода из finally-блоков.

Алгоритм состоит в итеративной обработке всех методов в программе с целью определения типов исключений, неявно продвигаемых каждым методом. В процессе определения таких типов для метода M, происходит добавление вершин исключительного вызова, возврата и выхода в УГ метода M, соединение всех УГ, используя четыре перечисленных выше типа межпроцедурных дуг.

Алгоритм вывода типов исключений, выбрасываемых оператором

Правильность приведенных алгоритмов построения УГ зависит от точности определения типов исключений, которые могут быть выброшены в различных точках программы. В приведенной выше графовой модели каждая вершина throw имеет столько выходящих дуг, сколько типов исключений может быть выброшено соответствующим оператором throw. Для определения типов исключений используется алгоритм вывода типов, основанный на подходах, описанных в [3, 4].

В ходе построения УГ, если текущий узел АСД оказался throw-оператором, происходит определение типов исключений, выбрасываемых этим оператором с помощью алгоритма ОПРЕДЕЛИТЬ_ТИПЫ, блок-схема которого изображена на рис. 4.

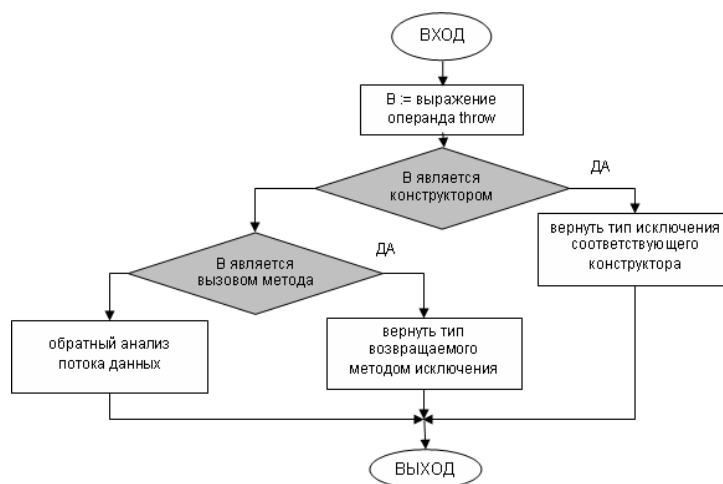


Рис. 4. Общая схема алгоритма вывода типов исключений

Этот алгоритм предполагает обратный анализ потока данных, начиная с вершины throw УГ.

ОПРЕДЕЛИТЬ_ТИПЫ (THROW, УГ).

THROW – узел throw, для которого определяются типы, УГ – неполный УГ.

Шаг 1. Если выражение, представленное в throw, является вызовом конструктора, вернуть тип исключения, соответствующего этому конструктору. Если выражение является вызовом метода, вернуть тип возвращаемого методом исключения и все его подтипы. Иначе в throw указана переменная, перейти к шагу 2.

Шаг 2. Обратный анализ потока данных, начиная с текущей вершины throw. Рассматриваются все родители текущей вершины. Для каждой такой вершины: если это вершина catch, перейти к шагу 3, если это вершина присваивания, перейти к шагу 4, если это вершина-вход УГ, перейти к шагу 5. Иначе перейти к шагу 2 для этой вершины.

Шаг 3. Добавить к возвращаемым типам тип, заданный в обработчике, и все его подтипы, перейти к шагу 2 для следующего родителя.

Шаг 4. Если выражение в правой части присваивания является вызовом конструктора, вернуть тип исключения, соответствующего этому конструктору. Если выражение является вызовом метода, вернуть тип возвращаемого методом исключения и все его подтипы. Иначе в throw указана переменная, перейти к шагу 2 для текущей вершины.

Шаг 5. Вернуть тип переменной и все его подтипы.

Реализация

Описанные алгоритмы реализованы программно в рамках системы построения управляющего графа программы. Входными данными для системы являются файлы, содержащие исходный текст программы, написанной на языке Java. Текст программы должен соответствовать грамматике языка Java и не должен содержать синтаксических ошибок. Система автоматически строит управляющие графы каждого метода и межпроцедурный управляющий граф в графическом и текстовом виде.

Заключение

Исследована проблема тестирования программ, содержащих обработку исключений. Разработаны алгоритмы построения управляющего графа для метода, алгоритм выведения типов выбрасываемых оператором исключений, а также алгоритм построения межпроцедурного управляющего графа. Для демонстрации разработанных алгоритмов реализована программная система, позволяющая получить из исходного текста программы ее управляющий граф. Выходные данные системы могут быть использованы при структурном тестировании.

Литература

1. Sinha S., Orso A., Harrold M.J. Automated support for development, maintenance, and testing in the presence of implicit control flow. / Proceedings of the 26th International Conference on Software Engineering (ICSE'04), 2004.
2. Shujuan Jiang, Yuanpeng Jiang. An analysis approach for testing exception handling programs. // SIGPLAN Notices. «ACM» – NY, USA. – 2007. – Vol. 42.
3. Palsberg J., Schwartzbach M. Object-oriented type inference / Proceedings of Object-Oriented Programming Systems, Languages and Applications. – 1991. – P. 146–161.
4. Plevyak J., Chien A. Precise concrete type inference for object-oriented languages. // Proceedings of Object-Oriented Programming Systems, Languages and Applications. – 1994. – P. 324–340.

ИЗГОТОВЛЕНИЕ ПРОТОТИПОВ ПЕЧАТНЫХ ПЛАТ В УСЛОВИЯХ ЕДИНИЧНОГО ПРОИЗВОДСТВА

Ю.В. Федосов

Научный руководитель – к.т.н., доцент С.Ф. Соколев

Рассматриваются возможности имеющегося оборудования для изготовления прототипов печатных плат, и предлагается изготовление прототипов печатных плат с использованием проводящих полимеров.

Введение

В настоящее время на рынке автоматизированных устройств предлагаются различные установки для производства печатных плат (ПП). Как правило, эти приборы либо являются фотоплоттерами, т.е. конечным результатом их работы является фотошаблон для печатной платы, либо устройство наносит пасту для пайки, устанавливает компоненты на уже готовую плату.

Как правило, эти установки производятся различными фирмами, которые не стремятся к объединению самого технологического процесса в единое целое (подготовка стеклотекстолита, нанесение фоторезиста, производство снимка, проявка, промывка, сверление отверстий, лужение дорожек, нанесение пасты), а предлагают реализацию его части. Если предприятие обладает достаточными средствами для покупки и сопровождения такого парка станков, возникает ряд вопросов, связанных с их наладкой, ремонтом, подготовкой кадров, а также с совместимостью оборудования. Известная ситуация, когда изделие (в данной ситуации ПП), будучи неоднократно извлеченным, из одного станка и переустановленным в другой, теряет точность изготовления из-за смены схем базирования, что неизбежно сказывается на качестве выпускаемой продукции. Так, например, некоторыми фирмами производятся компактные станки для установки DIP-компонентов на плату, однако вследствие небольших размеров станка и вибраций, производимых сервоприводами, компоненты смещаются по направлению, перпендикулярному движению платы, даже тогда, когда они приклеены к ее поверхности.

В наши дни активно развиваются САЕ-системы, позволяющие как воспроизвести изделие, так и в сжатые сроки создать прототип нового устройства. Известность установок, воспроизводящих твердотельные модели, заставляет задуматься о возможности быстрого воспроизведения различных электронных узлов для последующей наладки и доработки. Макетные платы, предлагаемые рядом фирм, обладают рядом существенных недостатков, одним из которых является вынужденное использование навесного монтажа, и, как следствие, возможные наводки в схеме. Кроме того, наладка ПП, выполненная навесным монтажом, представляет довольно сложную задачу.

Различные компании, в частности, Everprecision Tech или LPKF, специализирующиеся на оборудовании для изготовления прототипов плат, предлагают изготовление печатной платы с использованием фольгированного диэлектрика в качестве заготовки [1, 2]. Так, установки ProtoMat 95s/II и EP2002 сфрезеровывают определенные участки фольги, оставляя проводники толщиной до 0,1 мм на изоляционной подложке. Если требуется многослойная печатная плата, производителю предлагается спрессовывать несколько изготовленных плат в блок.

Описание решения

С подобной задачей (изготовление прототипов многослойных печатных плат) может справиться установка, которая выполняет послойное наращивание проводников и диэлектрика, используя в качестве токоведущего материала проводящий полимер [3, 4]. В этом случае не требуется мощного силового привода установки, поскольку процесс

изготовления платы подразумевает только перемещение каретки по двум взаимно перпендикулярным осям для осуществления процесса изготовления платы и, может быть, опускание платы на глубину в несколько десятков микрон для наращивания нового слоя. Также можно отметить, что такая установка будет создавать весьма малый уровень шума, который может быть с успехом нейтрализован с использованием достаточно простых схем.

В данной установке в качестве токоведущего материала может использоваться токопроводящий быстросохнущий клей (аналог современного токопроводящего клея), заключающий в своем составе как фуллерены, так и серебряные гранулы [5], отверждаемый тепловым излучением ИК-диапазона, а в качестве диэлектрика – полимерный порошок, подобный используемому в аппаратах быстрого прототипирования для нужд машиностроения.

Весь цикл изготовления прототипа будет экологически чистым и, за счет использования необходимого материала с достаточной точностью, безотходным. Рис. 1 отражает перечисленное.

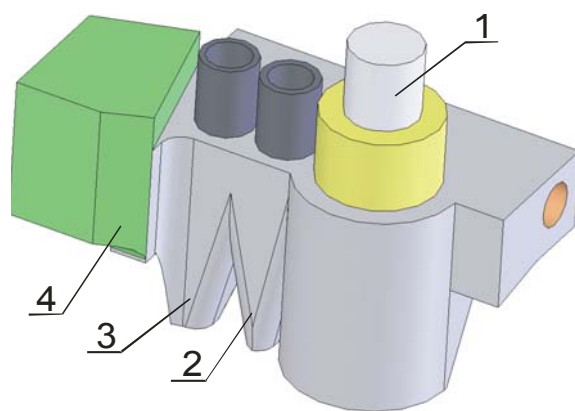


Рис. 1. Возможная конструкция каретки прототипирующего устройства: 1– источник ИК-излучения, 2 – устройство подачи проводящего полимера, 3 – устройство подачи изолирующего полимера, 4 – дозатор

В силу того, что технология изготовления изделия будет подразумевать экранизацию проводящего рисунка, изделие будет достаточно влагозащищенным [6]. Также стоит отметить, что в технологический процесс изготовления можно включать установку SMT-компонентов, что позволит получить на выходе уже не только саму плату как подложку для дальнейшей работы, а всю плату, собранную целиком. Также можно отметить, что полученный прототип печатной платы (без предустановленных компонентов) существенно облегчает последующий процесс монтажа, позволяя применять в качестве токоведущего соединителя обычный электропроводящий клей с составом, аналогичным предыдущему. Таким образом, переходное сопротивление будет лежать в диапазоне 1,6–2,5 Ом при толщине проводников до 0,1 мм. Диапазон рабочих температур полученного изделия будет лежать от $-60\text{ }^{\circ}\text{C}$ до $+65\text{ }^{\circ}\text{C}$, что соответствует стандартному климатическому исполнению УХЛ2.

Исходя из вышеописанного, можно определить функции и системные компоненты установки для создания прототипов печатных плат.

Поскольку устройство вряд ли будет использоваться для создания прототипов с габаритами большими, чем $500\times 500\text{ мм}$, и с количеством слоев, большим 30, то его вес едва ли будет превосходить 0,7 кг. Исходя из этого соображения, в качестве привода для перемещения платы может выступить низковольтный двигатель постоянного тока мощностью порядка нескольких ватт.

В качестве источника теплового излучения может выступить лазер ИК-диапазона. Поскольку толщина каждого слоя вкупе с толщиной предыдущего не будет превышать десятков микрон, с этой задачей может справиться лазер мощностью порядка нескольких сотен милливатт. Для привода каретки, которая будет включать в себя лазер, устройство нанесения порошка и емкость с проводящим составом, также можно применить два низковольтных двигателя постоянного тока мощностью порядка нескольких ватт.

Поскольку предлагаемое устройство должно обеспечивать довольно высокую точность изготовления изделия, то в данной конструкции следует применять не ременные передачи, а винтовые – это позволит более точно позиционировать каретку и облегчит управление ее перемещениями. Рис. 2 отражает основные моменты кинематической схемы устройства.

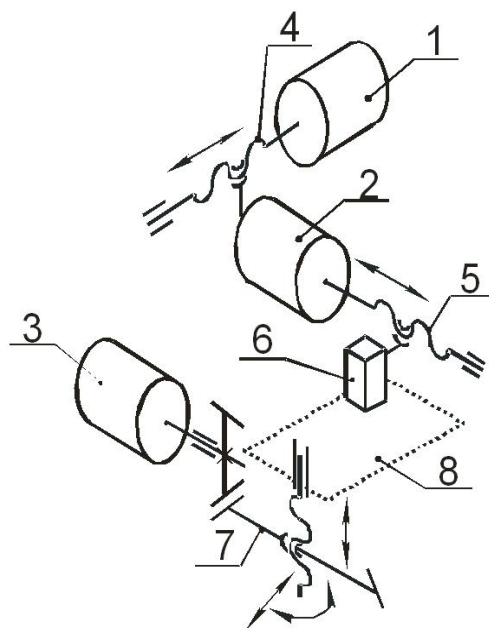


Рис. 2. Кинематическая схема: 1–3 – двигатели постоянного тока, 4, 5 – приводы каретки, 6 – каретка, 7 – привод стола с заготовкой, 8 – плата

Исходя из того, что в настоящий момент форматом де-факто проектирования печатных плат является спецификация Gerber RS-274X, можно сказать, что управление устройством должно осуществляться от встроенного контроллера, плата которого должна подключаться к компьютеру стандарта IBM PC через разъем одного из портов и программного обеспечения, переводящего векторную графику Gerber RS построчно в растровое изображение и, соответственно, в координаты для позиционирования головок.

Цикл работы устройства можно сформулировать следующим образом:

- формирование рисунка проводников;
- сушка;
- нанесение слоя порошка с попутным формированием проводящих межслойных взаимосвязей;
- оплавление;
- повтор вышеперечисленного по необходимости.

Заключение

В статье было рассмотрено состояние технологического оборудования для изготовления печатных плат и прототипов печатных плат. Был предложен способ изготовле-

ния прототипов печатных плат с использованием проводящего полимера в качестве токопроводящего проводника. Была рассмотрена возможная конструкция предлагаемого устройства, описаны принципы его работы и предложена его схема.

Литература

1. Черкасов С. Оборудование компании Everprecision Tech для прототипов печатных плат // Электроника: Наука, Технология, Бизнес. – 2005. – № 7. – С. 78–79.
2. Rapid PCB Prototyping Solutions, LPKF ProtoMat 95s/II, проспект фирмы LPKF, 2004.
3. Plastic Electronics, D de Leeuw // Physics World. – D31. – March 1999.
4. R.E. Gleason. How far will curcuits shrink // Science Spectra. – 2000. – 20. – С. 32.
5. Аналитический портал химической промышленности – Режим доступа: <http://www.polymer.ru/>, свободный. – Загл. с экрана. – Яз. рус., англ.
6. Уразаев В.Г. Влагозащита печатных узлов. – М.: Техносфера, 2006.

ITSM В ITIL – СТРУКТУРНО-ОБРАЗУЮЩИЙ ПОДХОД К ПРОЕКТИРОВАНИЮ, ВНЕДРЕНИЮ И УПРАВЛЕНИЮ ИТ-СИСТЕМАМИ КЛАССА HELP (SERVICE) DESK

Р.В. Иванов, А.Е. Михайленко

Научный руководитель – к.т.н., доцент Н.Ф. Гусарова

Представлен анализ применения идеологии ITIL для информационного сопровождения обслуживания заявок в подразделениях технической поддержки.

Введение

В работе [1] рассматривалась задача моделирования процесса обработки заявок в службе технической поддержки (СТП) сложных технических систем. В ней было показано, что подавляющее большинство предприятий, входящих в сегмент малого и среднего бизнеса (SMB), не имеют четкого регламента работы службы технической поддержки. Такая ситуация связана с тем, что немногие компании готовы инвестировать средства в управление ИТ-инфраструктурой, а свои убытки, связанные с нарушением функционирования сервисов, считают допустимо-приемлемыми.

Большие компании, напротив, понимают важность ИТ-процессов и влияние их на бизнес в целом, поэтому вкладывают немалые средства в поддержание ИТ-инфраструктуры. Успешный опыт таких компаний нашел обобщение в библиотеке инфраструктуры информационных технологий ITIL [2]. В настоящей статье представлен анализ применения идеологии ITIL для разработки системы информационного сопровождения (ИТ-системы) обслуживания заявок в подразделениях технической поддержки.

Проблемно-ориентированный анализ структуры ITIL

ITIL (англ. Information Technology Infrastructure Library) – библиотека, описывающая лучшие из применяемых на практике способов организации работы подразделений или компаний, занимающихся предоставлением услуг в области информационных технологий [3], являющаяся де-факто общепризнанным стандартом управления информационными системами.

Использованный в библиотеке процессный подход полностью соответствует стандартам серии ISO 9000 (ГОСТ Р ИСО 9000) [4, 5]. Он не имеет себе равных по обеспечению измеримости и управляемости деятельностью предприятия, акцентирует внимание предприятия на достижении поставленных целей, а также на ресурсах, затраченных на достижение этих целей. Со временем из библиотеки ITIL выросла значительная часть принятых во всем мире стандартов построения ИТ-процессов.

ITIL (версия 2) состоит из 7 томов (рис. 1), в которых изложена методика управления ИТ-инфраструктурой предприятия [6]. Целями этой методики являются предоставление и обеспечение высокого качества ИТ-услуг, которые должны соответствовать бизнес-требованиям организации и повысить степень удовлетворенности пользователей. Каждый том библиотеки ITIL охватывает отдельную область управления ИТ-инфраструктурой.

В мае 2007 года Агентство Office of Government Commerce [7] (Великобритания) выпустило стандарт ITIL v.3 [8], включающий в себя только пять книг: Service Strategy, Service Design, Service Transition, Service Operation, Continual Service Improvement. На сегодняшний день не существует общепризнанного русскоязычного глоссария, поэтому использование методик этой версии стандарта только начинается.

Наиболее известная часть ITIL v.2 — десять базовых процессов (рис. 2), обеспечивающих поддержку и предоставление ИТ-сервисов – Information Technology Service Management или ITSM [9, 10, 11, 12].

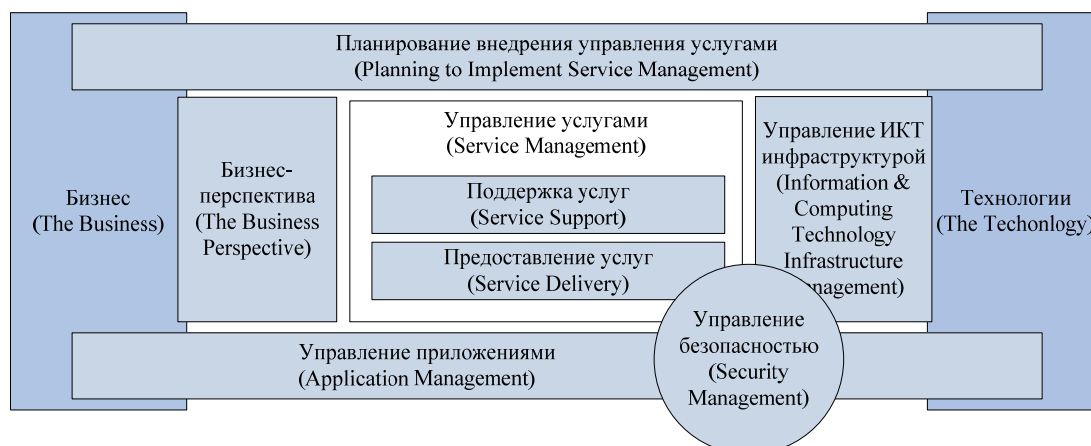


Рис. 1. Структура ITIL v.2

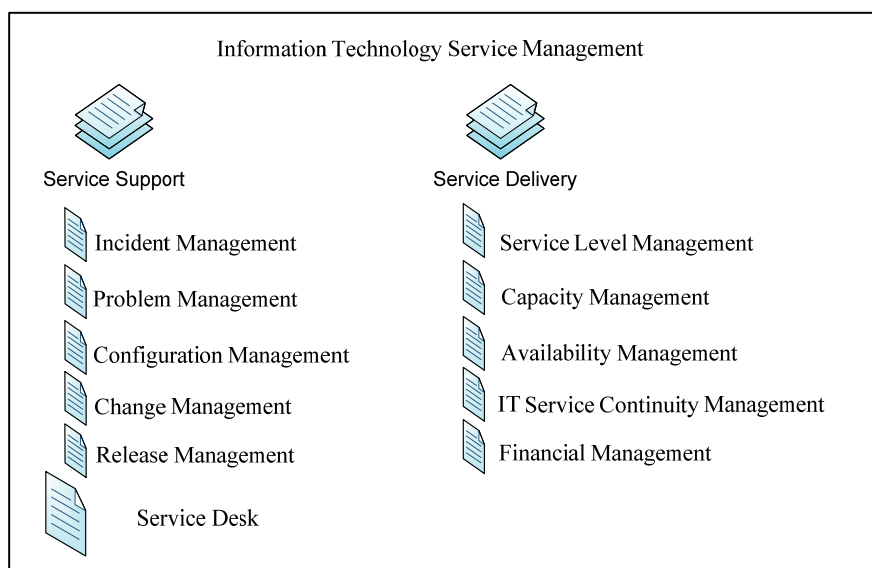


Рис. 2. Структура ITSM

Кроме того, в структуре процессов ITSM важную роль играет служба поддержки пользователей «Service Desk».

В аспекте задачи моделирования процесса обработки заявок в СТП наибольший интерес представляют такие части книги «Service Support» (поддержка услуг) библиотеки ITIL, как «Service Desk» (служба поддержки пользователей), «Incident Management» (процесс управления инцидентами) и «Problem Management» (процесс управления проблемами).

Анализ подходов к парированию неопределенностей бизнес-процесса на базе концепции ITIL

Многие компании предложили свои подходы к реализации идей, заложенных в ITIL. Наибольшую известность получили модель Hewlett-Packard – ITSM HP Reference Model [13] и методология Microsoft Operations Framework (MOF) [14, 15]. Внесенные в

MOF, по сравнению с ITIL, дополнения и изменения позволяют использовать ее в гетерогенных средах (рис. 3). Отметим, что все модификации, вносимые конкретными фирмами, отражают специфику рассматриваемых ими конкретных бизнес-процессов.

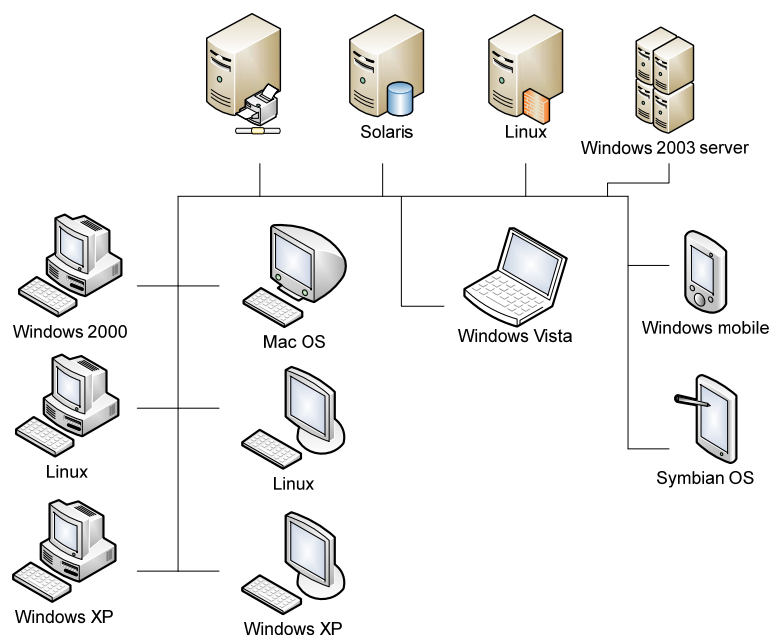


Рис. 3. Пример гетерогенной среды

Как показано в работе [1], специфика бизнес-процесса СТП состоит в наличии в нем неопределенностей различного происхождения (в частности, совмещение ролей акторами СТП, семантические неопределенности формулировок и др.). Рассмотрим возможности вышеописанных функций управления «Service Desk», «Incident Management» и «Problem Management» для парирования неопределенностей бизнес-процесса.

Service Desk – Служба поддержки

Проактивное взаимодействие. При управлении запросами клиентов важную роль играет понятие единой точки доступа. HelpDesk, являясь единой точкой доступа, должен представлять собой последовательный интерфейс между пользователями и техническими специалистами ИТ-отдела, реализуемый, например, посредством традиционных видов связи (телефон, e-mail, факс, веб-форма и др.).

Независимо от конкретной технологии реализации интерфейса, ITIL рекомендует предусмотреть в ИТ-системе тактику «проактивных взаимодействий», являющуюся предупредительным действием. Служба поддержки должна заранее (проактивно) обеспечивать клиентов информацией, например, о запланированной остановке сервиса, профилактических или аварийных работах. Эта информация может включать в себя известные проблемы, которые, вероятно, вызовут трудности или прерывания сервиса в будущем.

Как показал анализ, в разрабатываемой ИТ-системе наиболее эффективным является реализация проактивного взаимодействия в виде информирования группы пользователей, на работу которых могут быть наложены ограничения, связанные с недоступностью сервисов, или работа которых может быть полностью парализована, посредством e-mail. Кроме того, целесообразно предусмотреть информационную полосу на главной странице портала службы поддержки (рис. 4) для публикации информации о предстоящих мероприятиях и событиях, по которым в данный момент идет оператив-

ное реагирование, что уменьшит или исключит вовсе поступление заявок, на которые службе поддержки пришлось бы реагировать ответом о проведении работ.

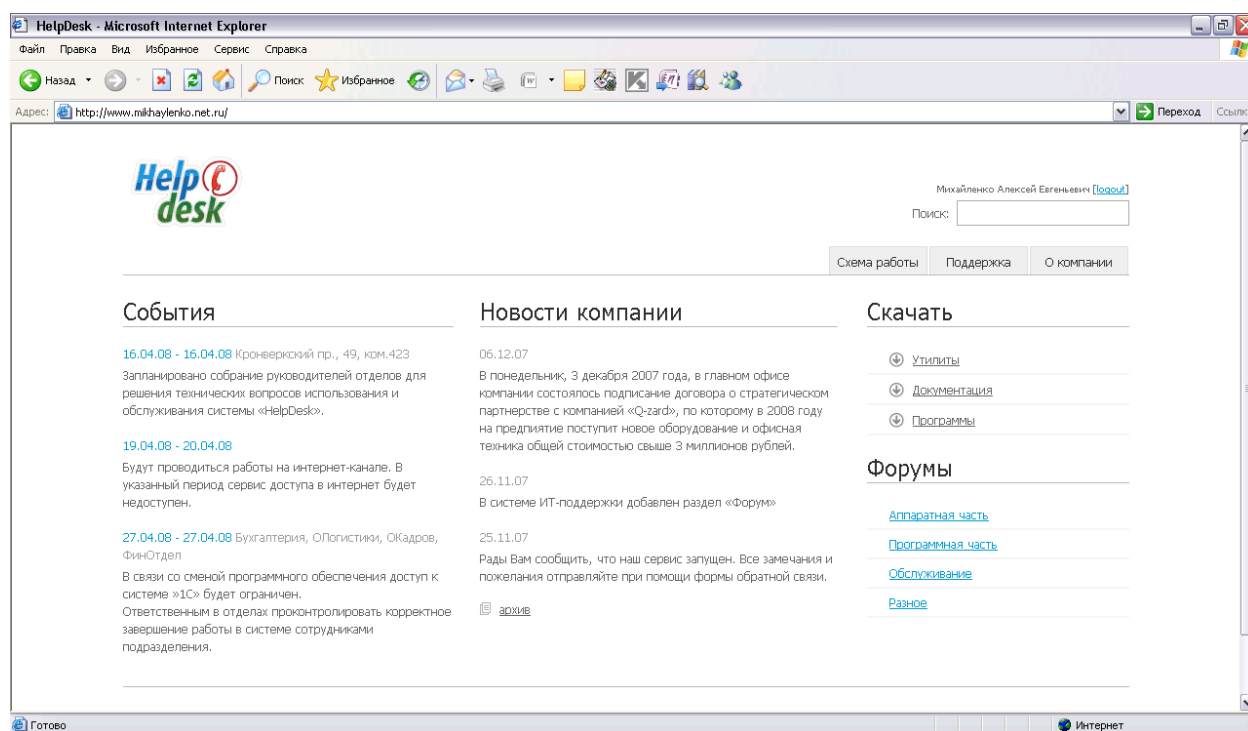


Рис. 4. Пример информационной полосы на главной странице портала службы поддержки

Организация системы ролевых функций. Как особо отмечено в ИТЛ, успешная работа службы поддержки возможна только при четком понимании каждым ее актором своей роли¹ и ее важности в общем деле. При создании ролей на этапе проектирования системы необходимо исходить из размеров самого предприятия, ИТ-инфраструктуры и ИТ-отдела. На этой основе следует определить вид структуры службы поддержки (СП):

- 1) централизованная СП, для которой не имеет значения географическое положение пользователя, использующая децентрализованные группы реагирования и разрешения заявок на местах;
- 2) децентрализованная СП, состоящая из нескольких служб в различных географических местах, например, оказывающая поддержку на разных языках в разных временных зонах;
- 3) виртуальная СП, объединяющая элементы централизованной и децентрализованной СП, основанная на развитии сетей и телекоммуникаций, использующая общий язык и стандарт записи запросов.

Однако все эти варианты подходят для организаций с достаточно развитой ИТ-инфраструктурой. Небольшие и малые организации, на которые, в первую очередь, ориентирована данная статья, используют схему «одна поддержка (операторы) + одна группа реагирования (в данном случае – технические специалисты)».

Для устранения неопределенностей, связанных с отсутствием закрепления должностного функционала за конкретными сотрудниками, в разрабатываемой ИТ-системе предлагается организовать систему ролевых функций, которая позволит создать конеч-

¹ Подробнее в [6] раздел «Service Desk» / Пункт «Service Desk responsibilities, functions, staffing levels etc».

ную матрицу должностных обязанностей в СТП и поставить ее в соответствие каждой роли (набору функций) конкретного сотрудника СТП.

Кроме того, для совершенствования модели обслуживания в СТП целесообразно организовать динамическое разделение и совмещение ролей в зависимости от состояния базы знаний ИТ-системы и других факторов. Формами такой организации могут являться совмещение ролей и введение второй линии поддержки пользователей.

Совмещение ролей. Такая форма организации может встречаться, например, на начальных этапах внедрения ИТ-системы, когда происходит начальное накопление базы знаний и поток входящих заявок является небольшим. В этом случае технические специалисты (класса «Эксперт») выполняют функции по приему, регистрации и разрешению заявок пользователей, т.е. выполняют функции оператора ИТ-системы и технического специалиста как такового (рис. 5).

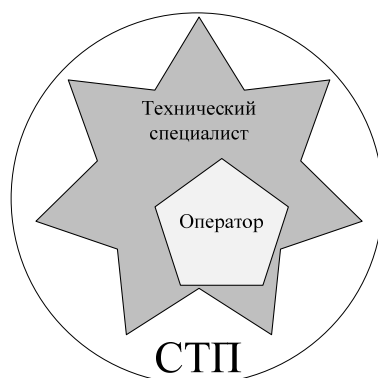


Рис. 5. Пример совмещения ролей

При росте количества сотрудников в компании увеличивается поток заявок, направляемых в СТП. Это может вызвать задержки в обслуживании вновь поступающих заявок. В такой ситуации становится очевидным, что введение роли оператора разгрузит технических специалистов от выполнения рутинных операций и позволит им заниматься техническими вопросами заявок.

Кроме того, при достаточном накоплении информации в базе знаний, как и в ситуации, описанной выше, целесообразно переложить часть функций на сотрудника с ролью «оператор». Даже не имея высокой квалификационной подготовки в предметной области, он, используя сформированную базу знаний, сможет разрешать входящие заявки и, тем самым, отсеивать поток заявок, решения по которым уже имеются.

Именно эти показатели говорят о необходимости перехода в стадию эксплуатации по схеме «оператор + технические специалисты». Однако при эксплуатации ИТ-системы происходит усложнение входящих запросов. Эти запросы могут быть не в компетенции оператора, но могут быть разрешены, с большой долей вероятности, более опытными операторами, поэтому, при наличии расширенного штата сотрудников СТП, рекомендуется введение второй линии поддержки.

Введение второй линии поддержки пользователей. Разделение функций и многоступенчатая обработка входящей заявки предполагает, что задачей команды первой линии является попытка разрешить поступивший запрос либо при первом контакте, либо используя известные обходные решения¹, а также собственные опыт и знания. Если запрос не будет разрешен описанными способами, то будет сформирована заявка и передана в группу реагирования – вторую линию поддержки, которая попытается решить

¹ обходное решение – временное решение, не устраняющее причины инцидента или проблемы.

вопрос, а при невозможности это сделать присвоит заявке соответствующие классификаторы и отправит ее в работу к техническим специалистам. Структуры группы реагирования могут различаться в зависимости от конкретного предприятия и быть организованы, например, по платформам и приложениям (группа серверов, десктопов, сетей или баз данных).

Таким образом, в ИТ-отделе динамично развивающейся компании можно будет выделить 3 ступени поддержки (рис. 6): первая линия поддержки, вторая (группа реагирования) и технические специалисты (заметим, что рекомендации ITIL не запрещают введения и большего числа уровней поддержки).

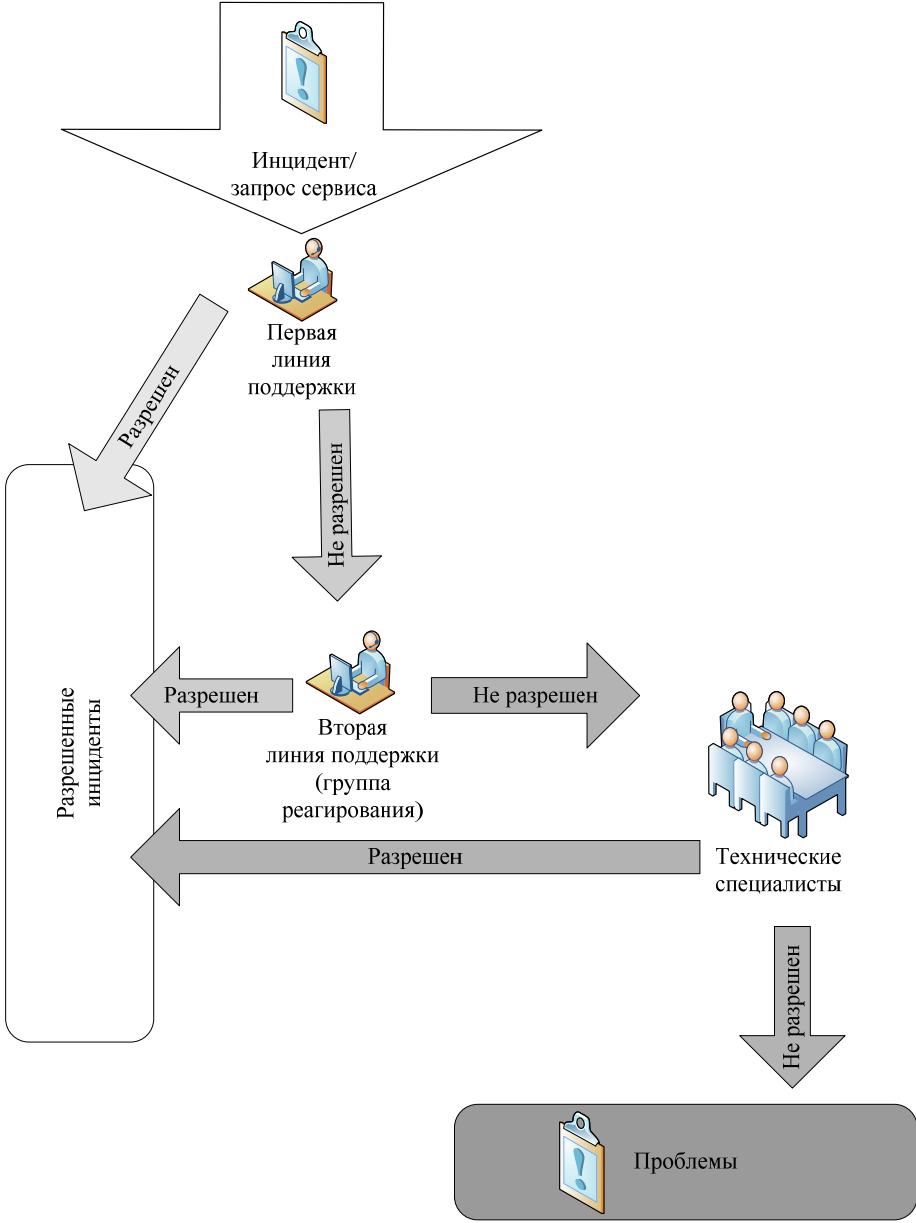


Рис. 6. Рекомендуемая структура службы поддержки

Следует отметить, что могут возникать ситуации, когда операции по каким-либо функциям могут быть временно переподчинены, переданы или упразднены, а, следовательно, и роли, в чью ответственность входили соответствующие работы. В первом случае снова может возникать совмещение ролей.

Для реализации перечисленных выше форм организации СТП в разрабатываемой ИТ-системе должны быть предусмотрены механизмы динамического управления ролями:

- механизм для создания и изменения структуры и иерархии ролей в службе поддержки в целях её более эффективной организации;
- механизм динамического изменения роли, который будет использоваться, например, в случае совмещения ролей (роль более высокого уровня иерархии может быть изменена на разрешенную роль более низкого или смежного, для совершения необходимых функций поддержки).

Работа механизма отслеживания позволит контролировать выполнение заявки в разрезе информации о том, кем она была выполнена, и в какой роли был исполнитель, а также даст возможность динамически регулировать количество персонала в той или иной роли, выделять новые роли, исходя из потребностей ИТ-инфраструктуры.

Использование рекомендаций раздела Service Desk применительно к разрабатываемой ИТ-системе помогает наиболее целесообразно организовать работу в СТП, распределить работы по ролям и создать четко формализованную матрицу ответственности в привязке к конкретным сотрудникам СТП.

Incident Management & Problem Management – Управление инцидентами и проблемами

Переходя к предоставлению ИТ-услуг, введем понятия инцидента, запроса сервиса и проблемы в нотации библиотеки ITIL [6, 16]:

- инцидентом называется любое событие, снижающее качество ИТ-услуги или ограничивающее полноту ее предоставления;
- проблемой называется группировка одного или более инцидентов, имеющих схожие характеристики и неизвестную причину (заметим, что при поступлении новой заявки невозможно сразу определить, является ли она заявкой по инциденту или по проблеме);
- запросом сервиса называется любое событие, не относящееся к выше перечисленным, в т.ч. запрос на смену пароля, подключение рабочего места нового сотрудника и т.п.

Известно, что на практике 20% проблем, которые приходится решать ИТ-специалистам, являются причиной 80% инцидентов, поэтому заблаговременное выявление проблем не менее важно, чем своевременное устранение инцидентов. В связи с этим управление инцидентами является важным процессом. Цель этого процесса – как можно скорее устранить любой инцидент.

Схема организации этого процесса, согласно рекомендациям раздела Incident Management (управление инцидентами), в котором освещаются проблемы восстановления работоспособности системы или отдельных ее компонентов при минимальных потерях для бизнеса, представлена на рис. 7.

Многие организации, действуя в соответствии с этой схемой, достигают высоких показателей в разрешении инцидентов, снижающих производительность, и уменьшении перерывов в работе, разрабатывая в высшей степени действенные процессы и процедуры управления инцидентами. Но при таком варианте развития событий в компаниях поддержка клиентов часто концентрируется на восстановлении после инцидентов, а не на устранении их основных первопричин. Это ведет к повторению инцидентов, что снижает как степень удовлетворенности клиентов, так и экономическую эффективность бизнес-процесса.

Для решения этой проблемы в разрабатываемой ИТ-системе предлагается после обработки заявки по схеме рис. 7 автоматически заносить максимально возможный

объем информации об инциденте и методах его разрешения в базу знаний для последующего ее использования. Это в дальнейшем приведет к тому, что многие сервисные запросы и инциденты будут обработаны и решены без выхода за пределы деятельности поддержки, т.е. без обращения к инженерам ИТ-отдела.



Рис. 7. Блок-схема управления инцидентами

Для выявления проблем (причин инцидентов) предлагается использовать ресурс линий поддержки. Выполняя повседневное координирование процесса управления инцидентами, они выявляют многократно повторяющиеся инциденты, которые указывают на основную проблему, т.е. линии поддержки являются важным источником информации для групп реагирования и ИТ-специалистов, которые занимаются управлением проблемами. В случае небольшого штата службы поддержки эти функции реализуются конкретными исполнителями в ходе совмещения ролей.

Выявленные таким образом проблемы будут фиксироваться в специальном разделе базы знаний, который будет доступен для использования как всем линиям поддержки, так и ИТ-специалистам. Первым – для установления соответствия новых инцидентов существующим проблемам, вторым – для возможного объединения нескольких проблем в одну или решения одних проблем на основе знаний о других.

Типовая схема управления проблемами, согласно разделу Problem Management, представлена на рис. 8. Управление проблемами направлено на поиск и фиксирование обходных путей и решений, которые могут заключаться в изменении ИТ-инфраструктуры (рис. 9). Эти решения и изменения должны применяться службой поддержки при первоначальной обработке новых инцидентов.

Управление инцидентами и проблемами тесно связаны, но имеют совершенно разную направленность в рамках организации поддержки. Если управление инцидентами направлено на как можно более быстрое восстановление нормальной работы сервиса, то управление проблемами концентрируется на выявлении и устранении основных проблем и их первопричин.



Рис. 8. Блок-схема управления проблемами

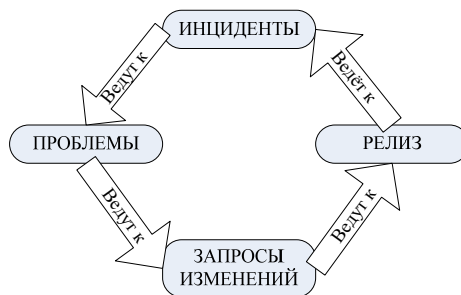


Рис. 9. Цикл событий проблемы

Заключение

В статье рассмотрены рекомендации библиотеки ITIL, являющиеся базовыми в аспекте разработки систем класса Help (Service) Desk. Сформулированы функциональные требования для эффективного управления инцидентами, проблемами и СТП в целом, что позволяет наиболее эффективно использовать потенциал СТП. Кроме того, выявлено, что накопленная информация по инцидентам и проблемам может являться источником знаний для самостоятельного решения пользователями возникающих проблем; использовать эту информацию также можно для упреждающих советов, тем самым, реализуя концепцию проактивности.

В долгосрочной перспективе использование разработанной ИТ-системы позволит повысить эффективность работ по обслуживанию заявок, сократить простои оборудования и, следовательно, связанные с ними расходы.

Литература

1. Иванов Р.В., Маятин А.В., Михайленко А.Е. Моделирование процесса обработки заявок в службе технической поддержки сложных технических систем // Научно-технический вестник СПбГУ ИТМО. – 2007. – Вып. 44. Современные технологии. – С. 268–274.
2. The Official ITIL ® Website [Electronic resource]. – Electronic data. – APM Group Ltd., cop. 2007-2008. – Mode access: <http://www.itsmportal.ru/>
3. Многоязычная общедоступная универсальная энциклопедия Википедия // ITIL [Электронный ресурс]. – Электрон. дан. – Режим доступа: <http://ru.wikipedia.org/wiki/ITIL>
4. ГОСТ Р ИСО 9001-2001. Системы менеджмента качества. Требования. Введ. 31.08.2001. – М.: Госстандарт России: Изд-во стандартов, 2001. – V, 22 с.
5. ГОСТ Р ИСО 9004-2001. Системы менеджмента качества. Рекомендации по улучшению деятельности. Введ. 31.08.2001. – М.: Госстандарт России: Изд-во стандартов, 2001. – VI, 48 с.
6. Service Management – ITIL® Version 2. – London.: Office of Government Commerce (OGC): TSO (The Stationery Office), 2000. – 312 pages
7. Office of Government Commerce [Electronic resource]. – Electronic data. – Crown, cop. 2008. – Mode access : http://www.ogc.gov.uk/guidance_itsm.asp
8. An Introductory Overview of ITIL ® v.3 [Electronic resource]. – Electronic data. – itSMF Ltd., cop. 2007. – Mode access : http://www.best-management-practice.com/gempdf/itsmf_An_Introductory_Overview_of_ITIL_V3.pdf
9. IT Service Management - ITIL® [Electronic resource]. – Electronic data. – TSO. – Mode access : <http://www.best-management-practice.com/IT-Service-Management-ITIL/>
10. Информационный портал по управлению ИТ [Электронный ресурс]. – Электрон. дан. – М.: Компания IT Expert. – Режим доступа: <http://www.itsmportal.ru/>
11. Независимый ITSM портал [Электронный ресурс]. – Электрон. дан. – М.: Бизнес-сеть "Kinetics", cop. 2006-2008 – Режим доступа: <http://www.itsmonline.ru/>
12. Многоязычная общедоступная универсальная энциклопедия Википедия // ITSM [Электронный ресурс]. – Электрон. дан. – Режим доступа: <http://ru.wikipedia.org/wiki/ITSM>
13. HP Services // ITSM HP Reference Model [Electronic resource]. – Electronic data. – Hewlett-Packard Development Company, L.P., cop. 2008. – Mode access: <http://h20219.www2.hp.com/services/cache/78360-0-0-0-121.html>
14. Microsoft TechNet // Microsoft Operations Framework (MOF) [Electronic resource]. – Electronic data. – Microsoft Corporation, cop. 2008. – Mode access: <http://www.microsoft.com/mof>
15. MOF Service Management Functions [Electronic resource]. – Electronic data. – Microsoft Corporation, cop. 2008. – Mode access: <http://www.microsoft.com/technet/solutionaccelerators/cits/mo/smf/>
16. Таблица соответствия терминов ITIL v.2 и их русских аналогов [Электронный ресурс]. – Электрон. дан. – М.: Компания IT Expert. – Режим доступа: <http://www.itsmportal.ru/articles/itil/2003-09-26%2000:00:00-18sprintable.html>

НАШИ АВТОРЫ

Urbanski Michael – Технический университет Ильменау, Германия
Абдурахманов Игорь Алиевич – аспирант кафедры мехатроники
Абрамчук Михаил Владимирович – ассистент кафедры мехатроники
Амвросьева Анна Владимировна – аспирант кафедры измерительных технологий и компьютерной томографии
Антонов Антон Валентинович – соискатель кафедры мехатроники
Ашмарин Юрий Александрович – магистр кафедры мехатроники
Боб Евгений Борисович – аспирант кафедры технологии программирования
Богданов Всеволод Викторович – аспирант кафедры технологии приборостроения
Бондаренко Роман Владимирович – аспирант кафедры технологии приборостроения
Бузунов Денис Александрович – студент кафедры мехатроники
Бузунов Роман Александрович – студент кафедры мехатроники
Виноградова Алла Алексеевна – студент кафедры мехатроники
Гаврилов Максим Иванович – аспирант кафедры высшей математики
Гладышев Геннадий Алексеевич – соискатель кафедры мехатроники
Голицына Татьяна Дмитриевна – аспирант кафедры информатики и прикладной математики
Гусельников Владимир Сергеевич – аспирант кафедры технологии приборостроения
Иванов Роман Владимирович – ассистент кафедры технологий профессионального обучения
Ильин Борис Александрович – аспирант кафедры технологии приборостроения
Исламова Анастасия Дамировна – студент кафедры компьютерной теплофизики и энергофизического мониторинга
Калитеевский Дмитрий Николаевич – студент кафедры мехатроники
Карпушинский Антон Михайлович – аспирант кафедры информатики и прикладной математики
Киселев Филипп Владимирович – аспирант кафедры технологии приборостроения
Коваленко Павел Павлович – аспирант кафедры мехатроники
Котельникова Юлия Евгеньевна – аспирант кафедры технологии приборостроения
Ларин Михаил Сергеевич – студент кафедры мехатроники
Латников Андрей Викторович – аспирант кафедры технологии программирования
Мальцев Александр Михайлович – аспирант кафедры технологии программирования
Мелих Светлана Андреевна – аспирант кафедры технологии приборостроения
Михайленко Александр Евгеньевич – студент кафедры физики
Недоцука Галина Александровна – аспирант кафедры мехатроники
Новиков Борис Юрьевич – аспирант кафедры лазерных технологий и экологического приборостроения
Окуловский Юрий Сергеевич – аспирант Уральского государственного университета им. А.М. Горького
Панова Ольга Игоревна – студент Московского государственного технического университета им. Н.Э. Баумана
Перечесова Анна Дмитриевна – студент кафедры мехатроники
Пестов Андрей Александрович – аспирант кафедры высшей математики
Потехин Андрей Евгеньевич – аспирант кафедры технологии программирования
Родинков Сергей Александрович – аспирант кафедры мехатроники
Саенко Алексей Петрович – студент кафедры измерительных технологий и компьютерной томографии

Саломатина Анна Алексеевна – аспирант кафедры технологии приборостроения
Свистунов Виталий Андреевич – аспирант кафедры технологии приборостроения
Сергушин Павел Анатольевич – аспирант кафедры мехатроники
Сизова Алина Александровна – студент кафедры мехатроники
Смагин Алексей Владимирович – аспирант кафедры технологии приборостроения
Соколов Дмитрий Александрович – аспирант кафедры информационно-навигационных систем
Соловьев Петр Сергеевич – аспирант кафедры компьютерных технологий
Суясов Дмитрий Игоревич – аспирант кафедры компьютерных технологий
Терещенко Александр Валерьевич – аспирант кафедры технологии приборостроения
Ткачёв Алексей Леонидович – аспирант кафедры мехатроники
Федосов Юрий Валерьевич – студент кафедры технологии приборостроения
Фомина Юлия Николаевна – аспирант кафедры технологии приборостроения
Шека Андрей Сергеевич – студент Уральского государственного университета им. А.М. Горького
Шемплинер Владимир Владимирович – аспирант кафедры измерительных технологий и компьютерной томографии
Шувал-Сергеев Никита Александрович – аспирант кафедры технологии приборостроения
Юльметова Ольга Сергеевна – аспирант кафедры технологии приборостроения

УПРАВЛЕНИЕ ФИЗИЧЕСКИМИ ПРОЦЕССАМИ	3
Панова О.И. Исследование теплообмена и гидродинамики на поверхностях со сферическими углублениями при различных числах Рейнольдса	3
Исламова А.Д. Метод измерения теплопроводности анизотропных высокотеплопроводных тел в виде стержней и пластин	10
Суясов Д.И. Выделение структурных признаков распознаваемых символов на основе набора клеточных автоматов	13
МЕХАТРОННЫЕ СИСТЕМЫ	20
Родинков С.А. Сравнение переходных характеристик стандартных сигналов и полиномов Уолша	20
Гаврилов М.И., Пестов А.А., Соловьев П.С. Звуковая локация и её цифровой анализ	25
Перечесова А.Д., Сергушин П.А. Закон движения механизма плетения САН	29
Соколов Д.А. Конструктивная реализация диаметральной схемы гироинклинометра с дополнительным разворотом корпуса скважинного прибора	36
Ткачѐв А.Л. Влияние температуры на частотные характеристики микроакустических датчиков на поверхностных и объемных акустических волнах	42
Коваленко П.П. Методика информационной оценки восприятия изображений	49
Абдурахманов И.А. Сравнение деформаций алюминиевых и кремниевых балок и пластин в тепловых полях	54
Шемплинер В.В. Восстановление дефокусированных изображений методом двумерного преобразования Фурье и регуляризации Тихонова	60
Новиков Б.Ю. Оптические характеристики микролинз, полученных методом лазерно-индуцированной аморфизации стеклокерамик	71
Амвросьева А.В. Расчет усилия в клеммных соединениях	81
Саенко А.П. Методы и средства оценки динамической остроты зрения	87
Ашмарин Ю.А., Виноградова А.А., Коваленко П.П., Недоцука Г.А., Urbanski M. Бесконтактный способ измерения геометрических размеров тел	90
Ларин М.С. Работа с пакетом программ Open Computer Vision	95
Виноградова А.А., Калитеевский Д.Н. Фрактальный генератор Коха для ковра Серпинского	100
Бузунов Д.А., Бузунов Р.А., Сизова А.А. Модернизация трибометрической системы «Трибал»	105
МЕХАНИКА, ТЕХНОЛОГИИ, САПР	108
Терещенко А.В. Влияние микрогеометрии поверхности на ее износостойкость-планирование эксперимента	108
Шувал-Сергеев Н.А. Интеграция конструкторских и технологических САПР на основе формата 3DXML	113
Бондаренко Р.В., Киселев Ф.В. Формирование базы знаний для ТПО	118
Богданов В.В. Проектирование технологических процессов с использованием удаленной САПР ТП	126
Ильин Б.А., Мелих С.А. Проектирование технологических процессов в PDM-системе ЛОЦМАН: PLM	131
Смагин А.В. Построение базы данных режущего инструмента в АСТПП	134

Юльметова О.С. Оптимизация микрогеометрии функциональных поверхностей деталей приборов для повышения их качества	140
Окуловский Ю.С., Шека А.С. Об архитектуре роботов и интеллектуальном управлении ими	143
Саломатина А.А., Фомина Ю.Н. Оптимизация конфигурирования и распределения заказов виртуального предприятия	151
Котельникова Ю.Е. Разработка учебно-методического взаимодействия учебно-методического комплекса с Internet	156
Антонов А.В., Гладышев Г.А. Расчет изгибных и контактных напряжений зубчатых колес с несимметричным профилем зуба методом конечных элементов	160
Свиштунов В.А. Табличный процессор для решения технологических задач	168
Абрамчук М.В. Формирование отечественных стандартов качества зубчатых колёс и передач, не противоречащих рекомендациям ISO	172
Гусельников В.С. Использование электронной модели детали при разработке технологического процесса	179
Голицына Т.Д. Принципы организации и интерфейс унифицированного модуля интеграции PDM и CAD систем	185
Боб Е.Б., Латников А.В., Мальцев А.М., Потехин А.Е. Эволюция методов и технологий программирования	191
Боб Е.Б., Латников А.В., Мальцев А.М., Потехин А.Е. Программисты, компиляторы, процессоры – поиск единого вектора	199
Карпушинский А.М. Построение управляющего графа программ, содержащих неявный поток управления	205
Федосов Ю.В. Изготовление прототипов печатных плат в условиях единичного производства	211
Иванов Р.В., Михайленко А.Е. ITSM в ITIL – структурно-образующий подход к проектированию, внедрению и управлению ИТ-системами класса HELP (SERVICE) DESK	215
НАШИ АВТОРЫ	225

**Научно-технический вестник СПбГУ ИТМО. Выпуск 48.
МЕХАТРОНИКА, ТЕХНОЛОГИИ, СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ.** / Главный редактор д.т.н., проф. В.О. Никифоров. – СПб: СПбГУ ИТМО, 2008. – 229 с.

**НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК СПбГУ ИТМО
Выпуск 48**

**МЕХАТРОНИКА, ТЕХНОЛОГИИ, СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ**

Главный редактор
доктор технических наук, профессор
В.О. Никифоров

Дизайн обложки В.А. Петров, А.А. Колокольников
Редакционно-издательский отдел СПбГУ ИТМО
Зав. РИО Н.Ф. Гусарова

Лицензия ИД № 00408 от 05.11.99.

Подписано в печать 10.04.08.

Заказ 1188. Тираж 100 экз.