

Программный интерфейс для SAT-решателей на основе технологии JNI

Гречишкина Д., Чухарев К.И., Университет ИТМО

**Научный руководитель – Чивилихин Д.С., кандидат технических наук,
научный сотрудник ФИТиП, Университет ИТМО**

Введение и постановка задачи

При решении оптимизационных NP-полных задач с помощью методов программирования в ограничениях, например, с помощью сведения исходной задачи к задаче выполнимости булевой формулы (*Boolean satisfiability* – SAT), зачастую применяется итеративный подход – к базовой задаче последовательно добавляются дополнительные ограничения, пока не будет найдено наиболее оптимальное решение. Современные SAT-решатели позволяют решать такие задачи *инкрементально*, не теряя прогресса решения базовой задачи. Инкрементальный подход значительно ускоряет процесс решения.

Существующие современные SAT-решатели в основном написаны на языках C/C++, что обеспечивает их высокую производительность. Однако, сложность языков C/C++ делает трудным практическое применение SAT-решателей через их программные интерфейсы, а стандартный формат записи задач SAT – DIMACS – не позволяет динамически добавлять ограничения в процессе решения, то есть решать задачу инкрементально. Решением этой проблемы является *jnisat* – программное средство для языка Java, основанное на технологии JNI (*Java Native Interface*), которое позволяет использовать возможности SAT-решателей *Minisat* и *Picosat*. Однако, данное решение обладает рядом недостатков. Во-первых, *jnisat* не имеет поддержки решения с предположениями (англ. *assumptions*). Во вторых, *Minisat* и *Picosat* были созданы более десяти лет назад – за это время появились более эффективные аналоги (например, *CaDiCaL*), что подтверждается ежегодными соревнованиями SAT Competition и SAT Race.

Целью данной работы является создание удобного программного интерфейса для использования современных SAT-решателей в приложениях на языках Java и Kotlin.

Описание предлагаемого решения

Предлагаемое решение представляет собой интерфейс для SAT-решателя *CaDiCaL*, написанный с использованием технологии JNI. Интерфейс реализует стандартный интерфейс для инкрементального решения SAT — *IPASIR (Reentrant Incremental Sat solver API)*, то есть поддерживает и инкрементальное решение, и решение с предположениями. В целях уменьшения времени работы (обращения к нативной библиотеке имеют значительные накладные расходы) в интерфейс были добавлены методы, позволяющие передать сразу весь дизъюнкт, весь список предположений и получить всю модель.

Для иллюстрации работы с интерфейсом было реализовано несколько примеров, демонстрирующих ключевые возможности SAT-решателя, такие как, например, инкрементальность и работа с предположениями.

Результаты. Предлагаемое решение реализовано на языке Java и доступно онлайн: <https://github.com/DespairedController/satwrapper>. Эффективность решения была оценена с помощью технологии JMH (*Java Microbenchmark Harness*) на нескольких типовых задачах: нахождение n -ого числа Белла, доказательство принципа Дирихле для данного n (простое и инкрементальное решения). Из-за использования SAT-решателя CaDiCaL эффективность решения на задачах, приводящих к UNSAT, значительно выше, чем у `jnisat`.

Авторы

Гречишкина Д. _____

Чухарев К.И. _____

Научный руководитель

Чивилихин Д.С. _____