

УДК 519.63

## Оптимизация проектных решений на основе параллельного программирования

Придухо В. Т., Пилипенко Д. В., Багдеев П. А.  
Белорусский национальный технический университет

### *Введение*

Современные методы оптимизации далеко не всегда справляются с решением реальных задач без помощи человека. Нет, пока такой теории, которая учла бы любые особенности функций, описывающих постановку задачи. Следует отдавать предпочтение таким методам, которыми проще управлять в процессе решения задачи.

При решении конкретной задачи оптимизации исследователь прежде всего должен выбрать математический метод, который приводил бы к конечным результатам с наименьшими затратами на вычисления или же давал возможность получить наибольший объем информации об искомом решении.

Как правило, нельзя рекомендовать какой-либо один метод, который можно использовать для решения всех без исключения задач, возникающих на практике. Одни методы в этом отношении являются более общими, другие - менее общими. Наконец, целую группу методов нелинейного программирования на определенных этапах решения оптимальной задачи можно применять в сочетании с другими методами, например динамическим программированием или принципом максимума.

Пожалуй, наилучшим путем при выборе метода оптимизации, наиболее пригодного для решения соответствующей задачи, следует признать исследование возможностей и опыта применения различных методов оптимизации.

### *Постановка задачи*

В связи с вышесказанным было решено создать систему, которая бы проводила оптимизацию, используя одновременно несколько методов. Данная система должна отвечать следующим требованиям:

1. Реализация неструктурной оптимизации однокритериальных оптимизационных моделей

2. Использование при этом параллельного программирования, т.е. одновременное использование различных методов оптимизации в параллельном режиме
3. Расширяемость базы моделей
4. Расширяемость базы методов

### *Структура алгоритма*

Учитывая, что алгоритмы оптимизации существенно отличаются друг от друга по ряду показателей: области и скорости сходимости, трудоемкости поиска и т. д. Некоторые методы оптимизации (МО) хорошо сходятся вблизи оптимальной точки и плохо вдали, а некоторые, наоборот, быстро сходятся вдали от окрестности оптимальной точки и медленно в самой окрестности. Все сказанное приводит к выводу о необходимости комплексного применения МО при решении задач оптимального проектирования и выбора стратегии поиска.

*Первая стратегия решения задач* в подсистеме оптимального проектирования предусматривает на начальных этапах поиска применение глобальных процедур оптимизации для получения оценки глобального экстремума, а уточнение решения — быстросходящимися в окрестности оптимума локальными алгоритмами. Такая стратегия позволяет, во-первых, с достаточной надежностью и точностью определить значение глобального экстремума, а, во-вторых, снизить затраты на поиск. Снижение затрат обусловлено тем, что этапы глобального поиска могут выполняться с невысокой точностью и требовать поэтому минимального числа вычислений функции качества, а этапы локального уточнения производятся из области притяжения глобального экстремума.

*Второй стратегией решения задач* оптимизации является параллельное решение сформулированной задачи одновременно двумя и более алгоритмами оптимизации. В этом случае управляющая программа подсистемы осуществляет одновременный запуск программ, реализующих выбранные МО, и контролирующего блока. МО решают поставленную пользователем задачу оптимизации, в функции же контролирующего блока входит наблюдение за ходом отдельных процессов и корректировка их траекторий на основе оценки складывающейся в процессе поиска ситуации.

Разработано соответствующее программное обеспечение для реализации описанных выше стратегий по следующим принципам (рис. 1).



Рис.1 Схема оптимизационного алгоритма

При старте решатель получает стартовые параметры, после чего передает стартовые параметры в методы и запускает цикл поиска. Методы работают

параллельно до тех пор, пока не будет выполнено условие перезапуска алгоритмов.

При выполнении данного условия решатель

- 1 получает от методов найденные ими лучшие решения
- 2 выбирает из них лучшее
- 3 устанавливает его во все методы как стартовую точку
- 4 предлагает выбрать методы для следующего круга цикла
- 5 стартует следующий круг цикла

#### Описание программы

С точки зрения программной реализации данный программный продукт реализован в среде C#, методы реализованы в сборке Methods.dll, модели реализованы в сборке Model.dll. Трехмерная графика реализована с применением DirectX. Среде C# выбрана в связи с тем, что она представляет достаточно гибкий и удобный инструмент для разработчика, а также, в связи с тем, что в Dot.NET реализована достаточно стабильная работа многопоточных приложений.

Интерфейс пользователя состоит из одного окна, в котором производятся все действия связанные с процессом оптимизации и ряда вспомогательных диалоговых окон (рис. 2).

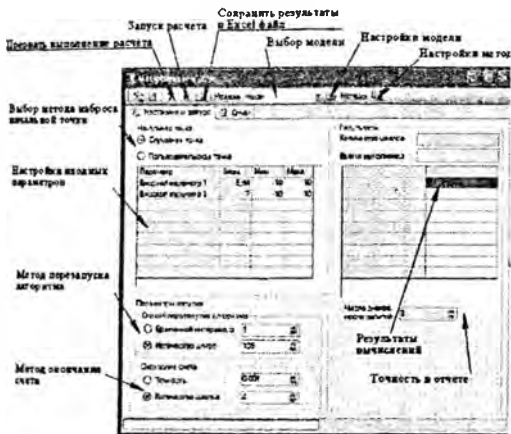


Рис.2 Интерфейс пользователя

По результатам вычислений отображается окно, в котором отображается двухмерное сечение по заданным осям оптимизируемой функции и траектория поиска алгоритма оптимизации.

Было проведено тестирование программного

продукта на ряде тестовых функций, таких как функция Розенброка, «Асимметричная долина», функция Пауэлла, функция Зангвилла. По результатам тестов оказалось, что алгоритмы оптимизации обладают хорошей сходимостью, что позволяет предложить данный оптимизационный модуль для решения широкого круга задач.

### Выводы

Применение данного программного продукта целесообразно по нескольким причинам.

- Независимость программы от других программ. Следовательно, увеличена скорость работы методов
- Запуск расчета возможен и неподготовленным в области методов оптимизации пользователем.
- Визуализация работы методов реализована на графике (построение поверхности, траектории методов) и в списке результатов.
- Возможность расширения библиотеки методов и моделей.