

ОБ ИСПОЛЬЗОВАНИИ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ И РАСПАРАЛЛЕЛИВАНИЯ ВЫЧИСЛЕНИЙ ДЛЯ ОПТИМИЗАЦИИ ЗАКАЗОВ ПРЕДПРИЯТИЯ

В. В. МУРОМЦЕВ
А. А. СЛОБОДЮК

Белгородский
государственный
университет

e-mail: muromtsev@bsu.edu.ru

В статье рассматривается задача оптимизации заказов предприятия. Предложен метод решения с использованием генетических алгоритмов. Рассмотрена возможность распараллеливания вычислений при решении поставленной задачи.

Ключевые слова: оптимизация заказов, генетический алгоритм, распараллеливание вычислений, MPI.

Введение

Важной составляющей экономической деятельности любого предприятия является закупка продукции. Вне зависимости от вида предприятия и разновидности закупаемой продукции, перед руководителями возникает задача организовать процесс закупок таким образом, чтобы минимизировать материальные затраты при сохранении качества и количества закупаемой продукции и уровня предоставляемого сервиса. Такую задачу будем называть задачей оптимизации заказов предприятия.

Обычно данная задача сводится к выбору поставщиков, которые предлагают необходимый заказчику товар на оптимальных условиях – достаточно низкая закупочная цена, выгодные условия доставки, предоставление некоторых скидок и т.д.

Если ориентироваться только на уровень цен на тот или иной товар у различных поставщиков, то организация закупок не составляла бы труда и представляла бы собой лишь изучение предложений поставщиков и выбора из них того, который предлагает товар по наименьшей цене. Но в действительности помимо уровня цен на решение заказчика влияют дополнительные условия, как, например, предоставляемые поставщиком скидки.

Практика показала, что ни «ручной» анализ, проводимый экспертом, ни прямой перебор всех возможных вариантов закупок товаров с помощью компьютера, неприменимы, т.к. требуют огромных временных затрат. Так, при необходимости заказать n видов товаров и наличии на рынке m различных поставщиков существует m^n различных вариантов плана закупок. Разумеется, выполнить полный перебор всех вариантов невозможно.

Одним из путей решения задачи оптимизации заказов является разработка специальных эвристических алгоритмов, одним из видов которых являются генетические алгоритмы. В статье рассматривается генетический алгоритм оптимизации заказов, а также приводится краткое описание возможностей распараллеливания вычислений.

Применение генетического алгоритма

Рассмотрим задачу оптимизации заказа, когда поставщики предлагают различные скидки. Как правило, скидка устанавливается на цены товаров, и ее величина дискретно зависит от суммы поставки. В этом случае затраты на закупку товаров выражаются многомерной мультимодальной дискретной функцией:

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n K(i) Cc \left(x_i, i, \sum_{j=1; x_j = x_i}^n K(j) C(x_j, j) \right),$$

где n – число закупаемых товаров, пронумерованных натуральными числами; $x_i \in \{1, 2, \dots, b\}$, $i = 1, 2, \dots, n$ – номер поставщика, у которого закупается i -й товар; b – число поставщиков; $K(i)$ – количество i -го товара, которое требуется закупить; $C(x_j, j)$ – цена j -го товара у поставщика x_j без скидки; $Cc(x_i, i, \sum)$ – цена i -го товара у поставщика x_i при закупке у данного поставщика товаров на сумму большую или равную порогу скидки \sum (цена со скидкой).

Некоторые поставщики устанавливают скидку на сумму поставки, величина которой дискретно зависит от данной суммы. В этом случае затраты на закупку товаров также выражаются многомерной мультимодальной дискретной функцией:

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n K(i)C(x_i, i) - \sum_{l=1}^b C \left(l, \sum_{j=1; x_j=l}^n K(j)C(x_j, j) \right),$$

где $C(l, \Sigma)$ – скидка, предоставляемая поставщиком с номером l , при закупке у данного поставщика товаров на сумму большую или равную Σ .

Для решения задачи оптимизации заказов предприятия можно использовать одну из разновидностей генетического алгоритма [1]. В начале работы алгоритма формируется множество приближенных решений $M = \{(x_1, x_2, \dots, x_n) \mid x_i \in \{1, 2, \dots, b\}, i = 1, 2, \dots, n\}$. После того, как множество M сформировано, осуществляется уточнение решений с помощью генетического алгоритма. В предлагаемом алгоритме элементы множества M рассматриваются как хромосомы. Генетический код особи представляется одной хромосомой. Таким образом, исходное множество особей (начальная популяция) совпадает с множеством M .

Основной шаг генетического алгоритма включает:

1. Выбор из текущей популяции M некоторого числа пар особей (родителей).
2. Выполнение генетических операций.
3. Отбор жизнеспособных особей.
4. Корректировка популяции M .

При выборе родителей предпочтение отдается особям соответствующим наименее затратным вариантам закупок требуемых товаров. Такой выбор организован путем вычисления для каждой особи $(m_1, m_2, \dots, m_n) \in M$ вероятности ее выбора в качестве родителя. Данная вероятность прямо пропорциональна значению $f(m_1, m_2, \dots, m_n)$.

Над выбранными парами родителей выполняются операции кроссинговера и репродукции. В результате выполнения таких операций два родителя $(m_1, \dots, m_{k-1}, m_k, \dots, m_n)$ и $(m'_1, \dots, m'_{k-1}, m'_k, \dots, m'_n)$ порождают двух особей: $(m_1, \dots, m_{k-1}, m'_k, \dots, m'_n)$ и $(m'_1, \dots, m'_{k-1}, m_k, \dots, m_n)$. Точка перекрестного обмена k выбирается случайным образом. Так же могут выполняться операции мутации. Мутации осуществляются над случайно выбранной особью. В результате мутации новая особь получается по одному из правил:

1. Случайно выбранный ген $m_i, i = 1, 2, \dots, n$ заменяется случайным числом в диапазоне от 1 до b .

2. Случайно выбранный ген m_i заменяется числом d в диапазоне от 1 до b . Здесь d – номер поставщика, которому заказаны товары на сумму, близкую к пороговому значению, после которого определена скидка у данного поставщика. Если таких поставщиков несколько, то при выборе d учитываются дополнительные критерии.

После выполнения генетических операций новые особи проверяются на жизнеспособность. Особь (m_1, m_2, \dots, m_n) считается жизнеспособной, если выполняется

$$\text{условие } f(m_1, m_2, \dots, m_n) < \sum_{i=1}^n K(i)C^*(i) + c_f, \text{ где } C^*(i) = \min_{x \in \{1, 2, \dots, b\}} \{C(x, i)\} - \text{минимальная цена на}$$

i -ый товар, предложенная одним из поставщиков, у которого может быть закуплен данный товар; $c_f \geq 0$ – некоторая заданная константа.

Жизнеспособные особи включаются в популяцию. Если размер популяции превышает заданный, то он корректируется. Корректировка размера популяции заключается в удалении наименее жизнеспособных особей. Из двух особей (m_1, m_2, \dots, m_n) и $(m'_1, m'_2, \dots, m'_n)$ наименее жизнеспособной считается особь (m_1, m_2, \dots, m_n) если $f(m_1, m_2, \dots, m_n) > f(m'_1, m'_2, \dots, m'_n)$.

Основной шаг генетического алгоритма повторяется заданное число раз (число поколений), также может быть задано время выполнения алгоритма.

Основные параметры генетического алгоритма: число поколений, размер популяции, вероятность мутации и др. являются настраиваемыми величинами и могут меняться в ходе работы алгоритма.

По завершении анализа заданного числа поколений в популяции отбирается наиболее жизнеспособная особь (m_1, m_2, \dots, m_n) . Если выполняется условие

$$f(m_1, m_2, \dots, m_n) < \sum_{i=1}^n K(i)C^*(i),$$

то закупка товаров осуществляется в соответствии с данной особью, т.е. первый товар закупается у поставщика m_1 , второй товар закупается у поставщика m_2 и т.д. В противном случае предложенным методом не удалось сократить затраты на закупку товаров за счет учета скидок, предоставляемых поставщиками.

Использование распараллеливания вычислений

При программной реализации генетических алгоритмов можно успешно использовать распараллеливание вычислений. Термин «распараллеливание» ассоциируется, прежде всего, с суперкомпьютерами и многопроцессорными системами. В силу ряда причин в России достаточное распространение получили только персональные компьютеры, и соответственно ситуация с освоением технологий работы с этими компьютерами более или менее благополучная. Компьютерный парк систем, относящихся к классу мультипроцессорных рабочих станций и, особенно, к классу суперкомпьютеров, чрезвычайно мал. Однако именно данная категория вычислительных машин заслуживает пристального внимания, так как параллельное выполнение множества программ для различных значений параметров позволяет существенно ускорить решение задачи. Также следует отметить, что использование многопроцессорных систем всегда более эффективно для обслуживания вычислительных потребностей большой группы пользователей, чем использование эквивалентного количества однопроцессорных рабочих станций, так как в этом случае с помощью некоторой системы управления заданиями легче обеспечить равномерную и более эффективную загрузку вычислительных ресурсов.

Возможны следующие режимы использования n -процессорной системы:

1. Все ресурсы отдаются для выполнения одной программы, и тогда можно ожидать n -кратного ускорения работы программы по сравнению с однопроцессорной системой.

2. Одновременно выполняется n обычных однопроцессорных программ, при этом пользователь вправе рассчитывать, что на скорость выполнения его программы не будут оказывать влияния другие программы.

Для решения поставленной задачи логично было бы использовать первый из выше представленных режимов. Также предлагается воспользоваться средой параллельного программирования MPI [2]. MPI – это библиотека функций, обеспечивающая взаимодействие параллельных процессов с помощью механизма передачи сообщений. Она является общепризнанным стандартом в параллельном программировании с использованием механизма передачи сообщений.

MPI-программа представляет собой набор независимых процессов, каждый из которых выполняет свою собственную программу, не обязательно одну и ту же. Процессы MPI-программы взаимодействуют друг с другом посредством вызова коммуникационных процедур. Как правило, каждый процесс выполняется в своем собственном адресном пространстве, однако допускается и режим разделения памяти. MPI не специфицирует модель выполнения процесса – это может быть как последовательный процесс, так и многопоточковый. MPI не предоставляет никаких средств для распределения процессов по вычислительным узлам и для запуска их на исполнение. Эти функции возлагаются либо на операционную систему, либо на программиста.

Библиотека MPI состоит примерно из 130 функций, в число которых входят:

- функции инициализации и закрытия MPI-процессов;
- функции, реализующие коммуникационные операции типа точка-точка;
- функции, реализующие коллективные операции;

- функции для работы с группами процессов и коммутаторами;
- функции для работы со структурами данных;
- функции формирования топологии процессов.

Общий алгоритм решения задачи с помощью многопроцессорной системы строится по схеме «мастер-рабочий». Мастер отвечает за работу макроалгоритма поиска минимума и за раздачу заданий рабочим. Мастер осуществляет запрос к пользователю о вводе начальных данных. Далее следует обработка введенных данных и передача их рабочим. Рабочие выполняют все необходимые вычисления, т.е. осуществляют поиск минимального значения некоторой функции. По истечении лимита времени исполнения рабочие осуществляют передачу результатов вычислений мастеру. Следующим этапом является анализ массива данных, полученных мастером от рабочих. Среди предложенных решений выбирается минимальное, т.е. наиболее полно удовлетворяющее условию задачи. Выбранное решение выводится в качестве окончательного результата работы алгоритма.

С использованием распараллеливания вычислений, созданный генетический алгоритм несколько модифицируется. Так, запрос о вводе начальных данных и составление начальной популяции будет осуществляться мастером, а выполнение генетических операций и поиск в новом поколении наиболее жизнеспособных особей – уже рабочими. Каждый рабочий по истечении лимита времени передает информацию о найденной наиболее жизнеспособной особи мастеру. Мастер анализирует все полученные особи и выбирает среди них наиболее приспособленные. В настоящее время такой вариант алгоритма оптимизации заказов предприятия реализован на локальной вычислительной сети, состоящей из персональных компьютеров. Для реализации использовался компилятор языка Си и библиотека MPI.

Выводы

Предложенное решение задачи оптимизации заказов предприятия с использованием генетических алгоритмов позволяет получить существенный выигрыш во времени. В сочетании с технологиями распараллеливания вычислений эффективность данного метода существенно повышается. Таким образом, предложенный алгоритм может быть использован для решения практических задач.

Литература

1. Муромцев, В.В. Организация конкурсных закупок. [Текст] / В.В.Муромцев // Научные ведомости БелГУ. Серия «Информатика и прикладная математика». – 2006. – №1(21), вып. 2. – с.56-65.
2. MPI: Message Passing Interface – русскоязычная страница [Электронный ресурс]: сайт о суперкомпьютерах и параллельных вычислениях. – Режим доступа: http://www.parallel.ru/tech/tech_dev/mpi.html, свободный.

ABOUT USAGE OF GENETIC ALGORITHMS AND PARALLEL CALCULATIONS FOR OPTIMISATION OF ORDERS OF THE ENTERPRISE

V.V. MUROMTSEV

A.A. SLOBODYUK

Belgorod State University

e-mail: anna-slobodyuk@yandex.ru

The article considers the problem of optimisation of orders of the enterprise is considered. The method of the decision with use of genetic algorithms is offered. Possibility to use parallel calculations is considered at the task in view decision.

Keywords: optimisation of orders, genetic algorithm, parallel calculations, MPI.