

АГЕНТНАЯ СИСТЕМА КООПЕРАЦИИ РЕСУРСОВ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ДЛЯ РЕШЕНИЯ ЗАДАЧ ВЫБОРА

Объект рассмотрения – организация решения разовых или эпизодически возникающих задач повышенной вычислительной сложности по кооперативной схеме использования ресурсов вычислительной сети. Кооперация ресурсов сети часто позволяет достичь нелинейного сокращения времени решения задачи без конструирования специальных вычислительных схем. Цель исследования – разработка шаблона платформенно-независимой открытой агентной системы решения задач выбора на конечных множествах вариантов и его реализация в рамках объектно-ориентированных технологий.

Задачи выбора на конечных множествах вариантов можно выделить практически во всех случаях дискретной оптимизации независимо от применяемой схемы поискового алгоритма (простого перебора, метода динамического программирования, ветвей и границ и др.). Конкретную решаемую задачу Z можно характеризовать тройкой $Z = \langle V, P, S \rangle$, где V – множество вариантов, подлежащих оценке по заданному критерию; P – процедура получения оценки качества для отдельного варианта из множества V ; S – процедура реализации вычислительной схемы решения Z , определяющая порядок применения P к элементам множества V .

Известно, что построение в приемлемые сроки строгих и, вместе с тем, изящных в математическом отношении схем решения практических задач дискретной оптимизации в большинстве случаев не представляется возможным.

Типичной является следующая ситуация:

- 1) имеется алгоритм и программа P оценки отдельного варианта из множества V ;
- 2) существует некоторый вариант схемы S , гарантирующий решение задачи Z ;
- 3) реализация схемы S возможна на параллельно функционирующих ЭВМ вычислительной сети путём декомпозиции тривиальным разбиением множества V .

Попытки разработки специальной схемы S , детально учитывающей особенности задачи Z , в ряде случаев не предпринимаются (принцип приоритета действия перед планом привлекателен для потребителя результатов). Очевидно, что прямолинейная реализация схемы S на N параллельно функционирующих ЭВМ приведёт к сокращению времени решения Z приблизительно в N раз. Реальные задачи часто позволяют использовать идею накопления опыта, возникающего при параллельном рассмотрении отдельных вариантов. Это приведёт к сокращению времени решения Z в первом приближении, как минимум, в N^2 раз [1], что особенно привлекательно в сетях с неравномерной загрузкой. Например, в системах управления группами оборудования в роботизированных производствах задачи оперативного планирования возникают редко, но требуют единовременно больших вычислительных ресурсов.

Анализ структуры задач выбора показывает, что реализация кооперативной схемы возможна в рамках системы, где основным активным элементом выступает агент, выявляющий доступность и готовность к решению задачи узла вычислительной сети. Известно, что “жадный” алгоритм распределения загрузки узлов сети является оптимальным по критерию минимума времени решения задачи Z [1]. В настоящее время существует ряд технологических возможностей реализации распределенных вычислений (RPC, CORBA, DCOM) [1], однако вопрос построения алгоритма решения задачи и его реализации остается в общем случае открытым.

Технология построения открытых систем взаимодействующих агентов в современных средах, включая Internet, базирующаяся на оформлении агента в виде

объекта класса [3], может быть основой решения задачи обеспечения коммуникации между узлами сети. Реализацию процедуры анализа варианта удобно представить классом на языках C++ или JAVA, где в случае необходимости может быть предусмотрен интерфейс вызова “родных” функций [3] вычислительной среды исполнения агента. Такие функции должны представлять, в частности, процедуру *P*. Остальные функции не являются проблемно-зависимыми и могут с точностью до протокола обмена соответствовать виртуальным операторам присваивания [2,3].

Идея объектно-ориентированного представления функциональной части агента (для краткости представлена шаблоном класса на языке C++ [2]):

```
template <class Criterial, class Variant> class Agent {
    Criterial F;
    static volatile Criterial R; // Глобальный объект рекордной оценки
protected:
    virtual Criterial FirstStep();
    virtual bool NextStep(Criterial &);
public:
    virtual void P(Variant V) {
        F = V.FirstStep(); // Предварительная оценка варианта
        if (R>F) { // Организация анализа варианта
            while (V.NextStep(F)) if (R<F) return;
            if (R>F) R = F; // Установление нового рекорда
        }
    }
};
```

Здесь предполагается решение задачи минимизации критериальной функции *F*. Кроме того, используемая для оценки отдельного варианта процедура *P* обладает возможностью получения в процессе работы нижних оценок значения функции *F*, а *R* – текущее значение рекордной оценки, является общедоступным для всех агентов.

В общем случае отказ от анализа варианта из-за установления его бесперспективности может потребовать нетривиальной реализации оператора сравнения *R* и *F*. Например, пусть методом статистических испытаний решается задача выбора оптимального по критерию *F* варианта системы из конечного множества конкурирующих вариантов.

Характерные черты метода статистических испытаний, существенные для организации корпоративной схемы вычислительного процесса:

- последовательная схема анализа вариантов системы, допускающая произвольный порядок их перечислений;
- итерационная схема анализа отдельного варианта, обусловленная необходимостью накопления статистики для достижения требуемой точности и достоверности результатов моделирования.

Пусть оцениваемое значение *F* для любого варианта системы распределено по нормальному закону с неизвестной дисперсией. Правомочность такого предположения следует из закона больших чисел и предельных теорем теории вероятности, а также из-за отсутствия обоснованных гипотез о значении дисперсии оценки *F*.

Доверительный интервал $\varepsilon(n)$ оценки *F* после *n* итераций статистических испытаний определяется при указанных предположениях выражением:

$$\varepsilon(n) = t_{\alpha/2, n-1} \sqrt{\frac{d}{n}},$$

где $t_{\alpha/2, n-1}$ – квантиль распределения Стьюдента с $n-1$ степенью свободы, соответствующий доверительной вероятности $1-\alpha$; d – оценка дисперсии величины F по результатам n испытаний.

Истинное значение F с вероятностью $1-\alpha$ находится в интервале $[F - \varepsilon(n), F + \varepsilon(n)]$, где F – оценка математического ожидания величины F по результатам n испытаний.

Пусть R – текущее значение оценки лучшего из проанализированных вариантов. Очевидно, что при решении задачи минимизации F условием прекращения анализа текущего варианта из-за его бесперспективности является $F - \varepsilon(n) > R$. При выявлении ситуации $F > R$, при некотором значении n можно определить требуемое число испытаний из уравнения $F - \varepsilon(m) = R$, где m – прогнозируемый номер испытания для вынесения, возможно, окончательного суждения об анализируемом варианте.

Можно заметить, что значение

$$m = \frac{F - R}{t_{\alpha/2, n-1} \cdot d}.$$

пригодно и для случая решения задачи максимизации значения F .

Полиморфизм функций сравнения позволяет считать агентный класс проблемно независимым. Таким образом, система взаимодействующих агентов обеспечивает объединение ресурсов вычислительной сети с целью декомпозиции исходной задачи. На примере комбинаторной задачи размещения производственных предприятий подтверждены результаты аналитических оценок эффективности кооперации ЭВМ [1].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Тихомирова Е.В. Характеристики потоков анализируемых вариантов в кооперативных схемах//Моделирование и информационные технологии проектирования: Сб. научн. тр., вып. 4. - Мн.: Объединенный институт проблем информатики НАН Беларуси, 2002. - С.110-116.

2. Гамма Э., Хелм Р., Джонсон Р., Влссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб.: Питер, 2001. – 386 с.

3. Родли Д. Создание JAVA – апплетов/Пер. с англ. – К.: НИПФ “ДиаСофт Лтд.”, 1996. – 386 с.

Ревотюк Михаил Павлович

Доцент кафедры информационных технологий автоматизированных систем, канд.техн.наук
Белорусский государственный университет информатики и радиоэлектроники
Тел.: +375(17)239-84-62
E-mail: rmp@bsuir.unibel.by

Кузнецова Наталья Владимировна

Студентка факультета информационных технологий и управления
Белорусский государственный университет информатики и радиоэлектроники
Тел.: +375(17)239-88-23
E-mail: kafitas@bsuir.unibel.by