

ИССЛЕДОВАНИЕ МУЛЬТИАГЕНТНЫХ АЛГОРИТМОВ РЕШЕНИЯ ЗАДАЧ ДИНАМИЧЕСКОГО ПЛАНИРОВАНИЯ

Е.В. Бояркина

Физический институт им. П. Н. Лебедева РАН

Россия, 119991, Москва, Ленинский проспект, д.53

E-mail: liza-boyarkina@yandex.ru

Ключевые слова: мультиагентные системы, задача task allocation, агентные контракты.

Аннотация: Для решения задач динамического планирования, в которых требуется в режиме реального времени распределять некоторые задания между исполнителями, применяются разные методы. В самом общем виде есть два подхода к решению - централизованный с «всезнающим» планировщиком, и децентрализованный, где исполнители решают все сами между собой. В данной работе рассматривается задача распределения заданий task allocation в децентрализованном случае, в которой исполнители «индивидуально рациональны», то есть локально ориентированы на собственный успех. При этом в итоге решение стремится к оптимальной общей выгоде. Предлагается реализация одного из способов решения этой задачи с помощью мультиагентных систем, и исследование ее эффективности в статическом и динамическом случаях.

1. Постановка задачи

Задача распределения заданий между агентами (task allocation problem) состоит из:

- T - множества заданий
- A - множества агентов
- $c_i : 2^T \rightarrow R \cup \{\infty\}, i \in A$ - функций стоимости, задающих стоимость для агента i конкретного набора задач из T (затраты на выполнение этих задач)
- $(T_1^{init}, \dots, T_{|A|}^{init})$, где $\bigcup_{i \in A} T_i^{init} = T$ и $T_i^{init} \cap T_j^{init} = \emptyset \forall i, j \in A, i \neq j$ - начального распределение заданий

Агенты могут обмениваться задачами для достижения наилучшего распределения задач, то есть социального благосостояния, при котором общие затраты всех агентов минимальны. Таким образом, целью является минимизация суммы затрат всех агентов:

Требуется: найти распределение $(T_1, \dots, T_{|A|})$ такое, что

$$(1) \quad \sum_{i \in A} c_i(T_i) \rightarrow \min$$

Функция стоимости выбрана следующим образом: пусть задача t_i - выполнить объем работ y_i , тогда:

$$(2) \quad c(t_i) = y_i^{1/\alpha},$$

где $\alpha < 1$ - параметр агента.

2. Типы контрактов

Рассмотрим различные типы контрактов, с помощью которых агенты договариваются об обмене задачами. Эти контракты введены в статье [1].

Определение 1 (One task контракт). *O-контракт определяется как пара $(T_{i,j}, \rho_{i,j})$, $|T_{i,j}| = 1$, где $T_{i,j}$ - множество задач (в данном случае ровно одна задача), которое агент i передает на исполнение агенту j , $\rho_{i,j}$ - цена, которую он за это платит.*

Определение 2 (Cluster контракт). *C-контракт определяется как пара элементов $(T_{i,j}, \rho_{i,j})$, $|T_{i,j}| > 1$, где $T_{i,j}$ - множество задач, которое агент i передает на исполнение агенту j , $\rho_{i,j}$ - цена, которую он за это платит.*

Определение 3 (Swarp контракт). *S-контракт определяется как кортеж элементов $(T_{i,j}, T_{j,i}, \rho_{i,j}, \rho_{j,i})$, $|T_{i,j}| = |T_{j,i}| = 1$, где $T_{i,j}$ - множество задач, которое агент i передает на исполнение агенту j , $\rho_{i,j}$ - цена, которую он за это платит, $T_{j,i}$ - множество задач, которое агент j передает на исполнение агенту i , и $\rho_{j,i}$ - цена, которую он за это платит.*

Определение 4 (Multiagent контракт). *M-контракт определяется как пара (T, ρ) матриц размера $|A| \times |A|$, в которых хотя бы три недиагональных элемента не пусты, где $|T_{i,j}| \leq 1 \forall i, j$. Элемент $T_{i,j}$ - множество задач, которое агент i передает на исполнение агенту j , $\rho_{i,j}$ - цена, которую он за это платит.*

Другими словами, в первых трех типах контрактов обмен происходит между двумя агентами, причем в O и C только один из них передает задачи другому (в O ровно одну, в C хотя бы 2), а в S контракте два агента обмениваются парой задач. В M контракте участвуют уже как минимум 3 агента. Из этих определений очевидно, что O контракт самый простой и быстро осуществимый. Поэтому хочется посмотреть насколько эффективно можно решать задачу, используя только его.

Ряд теорем из этой статьи говорит о том, что, в зависимости от начальных условий, могут потребоваться все типы контрактов для достижения глобального оптимума. Однако, как показано в работе, даже самый простой контракт на практике неплохо приближает решение к оптимальному.

3. Описание алгоритма

3.1. Простая версия

Сначала был реализован только алгоритм для самых простых контрактов - O-контрактов. Опишем его.

Допустим, что в начале задачи уже как-то распределены по агентам. В дальнейшей реализации первоначально распределяются просто поровну между всеми агентами. Опишем **этап подготовки алгоритма**.

Шаг 1. Сначала каждый агент определяет стоимость, которую он готов заплатить, отдавая задачу на исполнение. Эта стоимость определяется отдельно для каждой задачи, имеющейся у агента, и зависит как от сложности самой задачи, так и от сложности остальных имеющихся у агента задач.

Шаг 2. Теперь агенты начинают предлагать друг другу свои задачи.

Общение двух агентов происходит по следующему алгоритму: первый агент просматривает список задач и стоимостей второго агента, определяя самую выгодную для себя задачу, то есть такую, затраты на выполнение которой перекроются платой за ее выполнение. Выбранная задача помечается как возможная сделка.

Далее начинается **этап принятия решения**. Из всех возможных сделок, полученных на предыдущем этапе, выбирается та, которая принесет максимальную выгоду для всех агентов в целом. После чего выбранная сделка осуществляется, и предложения и возможные сделки пересчитываются.

Этот этап повторяется до тех пор, пока есть выгодные сделки.

3.2. Добавление более сложных контрактов

Далее к простому алгоритму были добавлены обменные S-контракты. Их реализация сходна с реализацией O-контрактов, с той разницей, что выгодной сделкой считается такой обмен двумя задачами между двумя агентами, который уменьшает их суммарные затраты, даже если затраты кого-то одного из них увеличились. Далее чередование O-S проводилось до тех пор, пока были возможности улучшения.

4. Результаты

В работе рассмотрены статический и динамический случаи задачи. В статическом случае исследована зависимость качества решения задачи простым O-контрактом от свойств задачи. Так, рассмотрены случаи одинаковых и разных по имеющимся ресурсам агентов, случаи больших и маленьких входных данных. В результате исследований, получены следующие результаты:

1. Решение предложенного алгоритма (только с простыми O-контрактами) не более, чем на 5% хуже оптимального в большинстве случаев. Более того почти в половине случаев алгоритм нашел оптимальное решение. При этом для некоторых начальных данных алгоритм справляется плохо, ошибаясь иногда на 40-60%.
2. Качество улучшается при увеличении меры разброса параметра α , то есть чем более разные агенты, тем лучше получается решение.

После добавления S-контрактов и чередования результаты улучшились, и максимальная ошибка снизилась до 7% на всех начальных данных.

В динамическом случае исследована способность алгоритма подстраиваться под случайные изменения задачи, такие как добавление новых заданий или удаление агентов. Для оценки качества работы алгоритма в разных ситуациях использовались разные метрики.

5. Выводы и дальнейшая работа

В результате различных экспериментов показано, что, хотя в теории простые контракты далеко не всегда приводят к оптимальному решению, на практике это решение получается близким к оптимальному. Дальнейшим этапом работы будет поиск критериев, по которым можно отличить сложные задачи (требующие более сложных контрактов) на начальном этапе.

Список литературы

1. Sandholm T.W. Contract Types for Satisficing Task Allocation: I Theoretical Results // AAAI Spring Symposium Series: Satisficing Models. 1998.