РАВНОВЕСИЯ ПО НЭШУ
В ИГРОВЫХ МОДЕЛЯХ РАЗМЕЩЕНИЯ[1]

Ю.А. Кочетов

Институт математики им. С.Л. Соболева СО РАН,
Новосибирский государственный университет, Новосибирск
e-mail: jkochet@math.nsc.ru

Аннотация. Рассматриваются дискретные задачи размещения в игровой постановке. Вводится понятие равновесного решения и показывается связь таких решений с локальными оптимумами. Установлено, что задача поиска равновесных решений принадлежит классу PLS и является плотно полной в этом классе.

Ключевые слова: равновесие по Нэшу, PLS-полные задачи, игровые модели размещения.

1. Введение

Многие оптимизационные задачи можно переформулировать в виде игры. В области дискретных задач размещения в последнее время активно исследуются игры Штакельберга. Игроки в таких играх неравноправны. Лидер делает ход первым, и его решение определяет систему ограничений для других игроков. Интерес к таким моделям связан как с многочисленными приложениями, например, исследованием иерархических систем управления, так и с вычислительной сложностью таких задач. В настоящей работе исследуется другой класс игр. Все игроки равноправны, не образуют коалиций и преследуют чисто эгоистические интересы. Они одновременно принимают решения, стараясь максимизировать суммарную прибыль от открытия предприятий и обслуживания потребителей. Тот факт, что игроки не образуют коалиций приводит к падению цен на рынке и перераспределению доходов между игроками и потребителями. Решение называют равновесным, если ни один из игроков не может увеличить свою прибыль при условии, что другие игроки не меняют свои решения. Понятие равновесного решения оказывается тесно связанным с понятием локального оптимума в соответствующей оптимизационной задаче. Таким образом, удается установить сложностной статус задачи поиска равновесных решений. В частности, для данной игры поиск равновесия не может быть NP-полной задачей, если NP$\neq$co-NP. Показано, что задача поиска равновесия принадлежит классу PLS и является плотно полной в нем. Отсюда в частности, следует, что доказательство существования полиномиального алгоритма вычисления равновесного решения было бы слишком сильным результатом. Все задачи из класса PLS решались бы в этом случае за полиномиальное время. Доказательство обратного утверждения приводит к выводу, что P$\neq$NP. Таким образом, вопрос о сложности нахождения равновесных решений в чистых стратегиях является интригующим и может иметь грандиозные последствия.

2. Игровая модель размещения производства

Рассмотрим ситуацию, когда p фирм одновременно открывают свои предприятия для обслуживания клиентов. Цель каждой фирмы (игрока) состоит в максимизации прибыли. Будем предполагать, что игроки принимают решения независимо друг от друга, не образуют коалиций и преследуют исключительно эгоистические интересы. Для простоты изложения будем считать,

[1]Работа выполнена при финансовой поддержке РФФИ (грант 06-01-00075)

что каждый игрок открывает одно предприятие, хотя все дальнейшие рассуждения можно обобщить и на случай произвольного числа предприятий.

Пусть величина r_j задает максимальную цену, которую клиент j согласен заплатить за предлагаемую продукцию, и матрица (c_{ij}) показывает затраты на обслуживание клиентов из открываемых предприятий. Если бы игроки объединились, стремясь получить максимальную прибыль, то их оптимальную стратегию можно найти из решения следующей оптимизационной задачи:

$$\max \sum_{j\in J}(r_j - \sum_{i\in I} c_{ij}x_{ij})$$

s.t.

$$\sum_{i\in I} x_{ij} \le 1, \quad j \in J;$$

$$x_i \ge x_{ij}, \quad i \in I, j \in J;$$

$$\sum_{i\in I} x_i = p;$$

$$x_i, x_{ij} \in \{0,1\}, \quad i \in I, j \in J,$$

где J — множество клиентов, I — множество предприятий, а переменные задачи имеют следующий смысл:

$$x_i = \begin{cases} 1, & \text{если открывается предприятие в пункте } i, \\ 0 & \text{в противном случае,} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{если клиент } j \text{ обслуживается из предприятия } i, \\ 0 & \text{в противном случае.} \end{cases}$$

Если первое ограничение выходит на равенство для всех $j \in J$, то каждый клиент заплатит максимальную приемлемую для него цену, и весь доход достанется игрокам.

Рассмотрим теперь ситуацию, когда игроки действуют независимо друг от друга. Каждый из них может открывать собственное предприятие в любом из пунктов множества I. Производственно-транспортные затраты k-го игрока зададим матрицей $c^k_{ij}, j \in J$. Без ограничения общности можно считать, что $c^k_{ij} \le r_j, j \in J$. В противном случае положим $c^k_{ij} = r_j$ и игрок не будет иметь прибыль от обслуживания этого клиента.

Пусть игроки открыли предприятия $i_1, i_2, \ldots, i_p$. Какую цену q_j заплатит j-й клиент за их продукцию? Обозначим через c_j минимальные производственно-транспортные затраты по обслуживанию этого клиента из открытых предприятий:

$$c_j = \min\{c^1_{i_1j}, c^2_{i_2j}, \ldots, c^p_{i_pj}\}, \quad j \in J.$$

Цена q_j не может быть меньше c_j, так как игрокам невыгодно поставлять продукцию по столь низкой цене. Предположим, что игроки выставили цену $q_j = r_j$. Тогда j-му клиенту все равно, какой из поставщиков будет его обслуживать. Так как каждый из игроков хочет оказаться поставщиком, то они начнут снижать цену. Пусть $i(j) = \arg\min\{c^1_{i_1j}, c^2_{i_2j}, \ldots, c^p_{i_pj}\}$ и c'_j — второй по величине минимальный элемент среди $c^1_{i_1j}, c^2_{i_2j}, \ldots, c^p_{i_pj}$. Процесс падения цены остановится на c'_j, когда игрок $i(j)$ останется единственным поставщиком, кому все еще выгодно обслуживание j-го клиента. Ниже опускать цену нецелесообразно. Выше поднимать цену нельзя, так как появляется конкурент, которому также становится выгодным обслуживание этого клиента. Получаем $q_j = c'_j$, $j \in J$. Если $c_j = c'_j$ для некоторого $j \in J$, то $q_j = c_j$, и обслуживание этого клиента не приносит прибыли ни одному из игроков.

Когда игроки действовали сообща, они держали цену q_j равной r_j и присваивали весь доход. Теперь $q_j = c'_j$ и прибыль $r_j - c_j$ делится между поставщиком $i(j)$, который получает $q_j - c_j$, и клиентом, который экономит величину $r_j - q_j$. Обозначим через Γ_k множество клиентов, обслуживаемых k-м игроком при заданном выборе $i_1, \dots, i_p$. Тогда прибыль k-го игрока равна

$$w_k = \sum_{j \in \Gamma_k} (q_j - c_j),$$

суммарная экономия клиентов определяется величиной

$$\nu(i_1, \dots, i_p) = \sum_{j \in J} (r_j - q_j),$$

суммарная прибыль игроков и экономия клиентов составляет величину

$$\mu(i_1, \dots, i_p) = \sum_{j \in J} (r_j - c_j) = \sum_{k=1}^{p} w_k + \nu(i_1, \dots, i_p).$$

Решение $(i_1, \dots, i_p)$ называют равновесием по Нэшу или равновесным решением, если ни один из игроков не может увеличить свою прибыль при условии, что другие игроки не меняют свой выбор. Оптимальным решением называют такое решение, при котором величина $\mu(i_1, \dots, i_p)$ достигает максимального значения. Такое решение действительно является наилучшим для общества, так как приносит наибольший «эффект» от размещения производства и выпуска продукции. Ниже будет показано, что оптимальное решение является равновесным, но не каждое равновесное решение является оптимальным. Для таких игр в [2] вводится понятие цены анархии. Эта величина равна отношению оптимального решения к наихудшему среди равновесий по Нэшу. Известно [3], что для данной задачи цена анархии не превосходит 2. Однако вопрос о трудоемкости нахождения равновесных решений до сих пор остается открытым. Неясно, можно ли найти хоть одно равновесное решение за полиномиальное время от длины записи исходных данных.

3. Связь с локальными оптимумами

Рассмотрим следующую оптимизационную задачу:

$$\min \sum_{j \in J} \sum_{k=1}^{p} \sum_{i \in I} c_{ij}^k x_{ij}^k$$

$$\sum_{k=1}^{p} \sum_{i \in I} x_{ij}^k = 1, \quad j \in J;$$

$$x_i^k \geq x_{ij}^k, \quad i \in I, j \in J, k = 1, \dots, p;$$

$$\sum_{i \in I} x_i^k = 1, \quad k = 1, \dots, p.$$

где переменные x_i^k и x_{ij}^k имеют следующий смысл:

$$x_i^k = \begin{cases} 1, & \text{если игрок } k \text{ открывает предприятие в пункте } i, \\ 0 & \text{в противном случае,} \end{cases}$$

$$x_{ij}^k = \begin{cases} 1, & \text{если клиент } j \text{ обслуживается игроком } k \text{ из предприятия } i, \\ 0 & \text{в противном случае.} \end{cases}$$

Целевая функция задачи задает производственно–транспортные затраты игроков. Первое ограничение требует обслужить всех клиентов. Второе ограничение позволяет обслуживать клиентов только из открытых предприятий. Последнее ограничение позволяет каждому игроку открывать только одно предприятие.

Заметим, что в этой задаче несколько игроков могут открывать предприятия в одном месте. Поэтому такая постановка близка по смыслу к задаче о p-медиане [1], но не идентична ей. Будем обозначать ее через *p-median game* или PMG. Пусть (x_i^k, x_{ij}^k) — допустимое решение этой задачи и при данных значениях x_i^k, величины x_{ij}^k задают оптимальное распределение клиентов между открытыми предприятиями. Другими словами, будем предполагать, что x_{ij}^k определяются по переменным x_i^k и решение можно задавать только этими переменными. Тогда задачу можно переписать следующим образом: найти

$$\min \sum_{j \in J} \min_{k=1,\ldots,p} \min_{i \in I} c_{ij}^k x_i^k$$

$$\sum_{i \in I} x_i^k = 1, \quad k = 1, \ldots, p,$$

$$x_i^k \in \{0, 1\}, \quad k = 1, \ldots, p, i \in I.$$

Пусть (x_i^k) — допустимое решение задачи. Под окрестностью $Swap(x)$ будем понимать множество допустимых решений, которые могут быть получены из решения (x_i^k) выбором некоторого k и заменой предприятия этого игрока на любое другое. Локальным минимумом для данной окрестности называется такое решение, для которого любое соседнее решение имеет не меньшие значение целевой функции.

Теорема 1 Между локальными минимумами задачи PMG и равновесными решениями существует взаимно–однозначное соответствие.

Таким образом, вопрос о нахождении равновесных решений тесно связан с поиском локальных минимумов.

4. Сложность нахождения равновесных решений

Рассмотрим вопрос о сложности получения локальных оптимумов.

Определение 1. *Оптимизационная задача* OP определяется следующим набором объектов $< \mathcal{I}, Sol, F, goal >$, где

1. $\mathcal{I}$ – множество входов задачи OP;
2. Sol – функция, которая каждому входу $x \in \mathcal{I}$ сопоставляет множество допустимых решений $Sol(x)$;
3. F – функция, которая задаёт вес $F(s, x)$ допустимого решения s на входе x;
4. величина $goal \in \{\min, \max\}$ уточняет является ли задача OP задачей на максимум или минимум.

В оптимизационной задаче необходимо найти оптимальное решение для заданного входа x.

Определение 2. *Задача локального поиска* определяется парой $\Pi = (OP, N)$, где OP — оптимизационная задача, а N — функция окрестности, которая каждому допустимому решению s на входе x ставит в соответствие множество $N(s, x) \subseteq Sol(x)$ соседних решений. Задача локального поиска заключается в отыскании локального минимума для заданного входа x.

Определение 3. Задача локального поиска Π принадлежит классу PLS, если существует три полиномиальных алгоритма A, B, C и полином q такие, что
1. Алгоритм A определяет является ли любое слово x входом задачи. Если $x \in \mathcal{I}$, то алгоритм

находит допустимое решение задачи OP;

2. Для любого входа задачи $x \in \mathcal{I}$ и любого слова s алгоритм B определяет является ли s допустимым решением. Если $s \in Sol(x)$, то за полиномиальное время алгоритм находит значение целевой функции $F(s, x)$;

3. Для любого входа $x \in \mathcal{I}$ и любого решения $s \in Sol(x)$ алгоритм C определяет является ли s локальным оптимумом. Если нет, то алгоритм находит соседа $s' \in N(s, x)$ с меньшим значением целевой функции;

4) Для любого входа $x \in \mathcal{I}$ длина любого допустимого решения $s \in Sol(x)$ полиномиально ограничена длиной входа задачи, т.е. $|s| \leq q(|x|)$.

Легко проверить, что задача поиска локального минимума для задачи PMG с окрестностью *Swap* принадлежит классу PLS.

Определение 4. Пусть Π_1 и Π_2 — две задачи локального поиска. Задача Π_1 PLS – сводится к задаче Π_2, если существуют такие полиномиально вычислимые функции h и g, что
1) по произвольному входу x задачи Π_1 функция h вычисляет некоторый вход $h(x)$ задачи Π_2;

2) по произвольному решению s входа $h(x)$ функция g находит некоторое решение $g(s, x)$ для входа x;

3) если $x \in \Pi_1$ и s — локальный оптимум для входа $h(x) \in \Pi_2$, то $g(s, x)$ — локальный оптимум для входа x.

Задачу Π из класса PLS называют PLS–полной, если любая задача из этого класса может быть сведена к ней. Примеры PLS-полных задач можно найти в [5]. Ниже будет показано, что поиск локального минимума в задаче PMG с окрестностью *Swap* является PLS-полной задачей, т.е. самой трудной задачей в этом классе. Более того, будет получена экспоненциальная нижняя оценка на число шагов в худшем случае для алгоритмов локального улучшения. Эта оценка не зависит от правила замещения, т.е. от способа выбора лучшего соседа в окрестности, но не закрывает возможности получения локального оптимума (равновесия по Нэшу) за полиномиальное время другими алгоритмами.

Определение 5. *Графом переходов* $TG_\Pi(x)$ для входа x задачи Π называется ориентированный граф, вершинами которого являются все допустимые решения задачи. Дуга (s, s') принадлежит графу, если s' —соседнее решение для s и $F(s, x) > F(s', x)$. Высота вершины s есть длина кратчайшего пути в графе $TG_\Pi(x)$ из вершины s в сток, т. е. в локальный минимум. Высота графа $TG_\Pi(x)$ равна максимальной высоте его вершин.

Высота вершины есть, по сути, оценка снизу на число шагов алгоритма локального улучшения независимо от применяемого правила замещения. Таким образом, алгоритм требует экспоненциального числа шагов тогда и только тогда, когда найдутся исходные данные задачи, для которых высота графа переходов является экспоненциальной функцией от длины записи исходных данных.

Определение 6. Пусть задача Π_1 PLS-сводится к задаче Π_2 и h, g — соответствующие функции. Говорят, что сводимость является *плотной*, если для любого входа x задачи Π_1 можно указать такое подмножество R допустимых решений входа $y = h(x)$ задачи Π_2, что
1) в R содержатся все локальные минимумы для входа y

2) существует полиномиальный алгоритм, который для каждого решения p входа x позволяет находить решение $q \in R$ для входа y такое, что $g(q, x) = p$;

3) пусть граф переходов $TG_{\Pi_2}(y)$ содержит ориентированный путь из вершины $q \in R$ в вершину $q' \in R$ такой, что в нем нет промежуточных вершин из R, и пусть $p = g(q,x)$ $p' = g(q',x)$ — соответствующие решения для входа x. Тогда $p = p'$ или граф переходов $TG_{\Pi_1}(x)$ содержит дугу из вершины p в вершину p'.

Задачу Π из класса PLS называют плотно полной, если все задачи из этого класса плотно к ней сводятся. Известно [4], что следующая задача о раскраске вершин графа в два цвета является плотно полной. Задан неориентированный граф $G = (V, E)$ с весами $w_e, e \in E$ на ребрах. Допустимым решением задачи является любая раскраска вершин в два цвета, т.е. функция вида $c : V \to \{0, 1\}$. В качестве целевой функции используется суммарный вес ребер, вершины которых окрашены в один цвет. Окрестность *Flip* состоит из всех раскрасок, отличающихся от заданной цветом одной вершины. Задача состоит в поиске раскраски, являющейся локальным минимумом для окрестности *Flip*.

Теорема 2 Задача поиска *Flip*-минимальной раскраски плотно сводится к задаче локального поиска (PMG, *Swap*).

Рассмотрим теперь вопрос о трудоемкости получения равновесных решений с помощью следующего итерационного алгоритма. Пусть некоторое решение (x_i^k) не является равновесным, т.е. существует игрок, быть может не один, который может увеличить свою прибыль, выбрав другое предприятие. Шаг алгоритма состоит в нахождении такого игрока и выборе для него нового предприятия с увеличением прибыли. Процедура выбора нового предприятия может быть любой. Наша цель будет состоять в оценке числа шагов такого итерационного процесса в худшем случае при любой процедуре выбора.

Теорема 3. Итерационный алгоритм в худшем случае требует экспоненциального числа шагов для нахождения равновесного решения при любой процедуре выбора игрока и лучшего предприятия для него.

Справедливость утверждения следует из плотной PLS полноты задачи локального поиска (PMG, *Swap*), и существования задачи экспоненциальной высоты в классе PLS [5]. Оценка остаётся экспоненциальной при использовании любого правила выбора игрока и лучшего предприятия для него: детерминированного, вероятностного или любого другого произвольной сложности.

В задаче поиска равновесных решений нужно найти хотя бы одно (любое) равновесное решение. Однако, если решения (стратегии) игроков уже известны, и они готовы их менять только при увеличении прибыли, то возникает другая задача, более важная с практической точки зрения. Найти равновесное решение, достижимое итерационным алгоритмом локального улучшения из заданной начальной стратегии. Другими словами, в графе переходов задана стартовая вершина. Требуется найти сток, достижимый из стартовой вершины вдоль некоторого ориентированного пути. Оказывается, что такая задача является значительно более сложной, чем предшествующая. Из следующей теоремы получаем, что если для этой задачи удастся построить полиномиальный алгоритм, то полиномиально разрешимой будет любая задача из класса PSPACE и, в частности, все задачи из класса NP.

Теорема 4. Нахождение равновесного решения при заданной начальной стратегии игроков является PSPACE-полной задачей.

Известно [5], что в классе PLS есть задачи локального поиска с фиксированной стартовой точкой, которые являются PSPACE-полными. Задача о раскраске вершин графа с окрестностью *Flip* — одна из них. Плотная сводимость, установленная в теореме 2, влечет полиномиальную сводимость соответствующих задач с фиксированными стартовыми решениями [5]. Следовательно, нахождение равновесного решения, достижимого из заданной начальной стратегии игроков является PSPACE-полной задачей.

Рассмотрим более общую ситуацию, когда игроки вынуждены платить за открытие предприятий. Обозначим через $f_i^k \geq 0$ затраты k-го игрока на открытие i-го предприятия. Теперь любой игрок может оказаться в ситуации, когда ему невыгодно открывать ни одного предприятия. Для такой игры соответствующая оптимизационная задача имеет вид:

$$\min \sum_{j \in J} \sum_{k=1}^{p} \sum_{i \in I} c_{ijk} x_{ij}^k - \sum_{k=1}^{p} \sum_{i \in I} f_i^k x_i^k$$

$$\sum_{k=1}^{p} \sum_{i \in I} x_{ij}^k = 1, \quad j \in J,$$

$$x_i^k \geq x_{ij}^k, \quad i \in I, j \in J, k = 1, \ldots, p,$$

$$\sum_{i \in I} x_i^k \leq 1, \quad k = 1, \ldots, p,$$

$$x_i^k, x_{ij}^k \in \{0, 1\}, \quad i \in I, j \in J, k = 1, \ldots, p.$$

Как и прежде решение будем задавать переменными x_i^k, восстанавливая по ним оптимальные значения x_{ij}^k. Соседним решением для решения x_i^k будем называть любое решение, которое может быть получено из данного выбором некоторого игрока k и заменой его решения на любое другое допустимое решение задачи. Если игрок k не имел открытого предприятия, то он может открыть любое. В противном случае он может закрыть его и либо открыть новое, либо не открывать никакого предприятия. Так определенная окрестность является, очевидно, более широкой, чем окрестность *Swap*.

Теорема 5. Нахождение равновесного решения при наличии платы за открытие предприятий является плотно PLS-полной задачей.

Предположим теперь, что каждый игрок может отрывать несколько предприятий. Для такого случая легко выписать соответствующую оптимизационную задачу и определить подходящую окрестность. Эта окрестность будет включать в себя окрестность *Swap* и, следовательно, соответствующая задача локального поиска оказывается PLS–полной. Плотная полнота такой задачи со всеми вытекающими отсюда последствиями является пока открытым вопросом.

5. Заключение

В работе рассмотрены игровые модели размещения производства. Показана связь равновесных решений с локальными оптимумами и установлена плотная PLS-полнота задачи нахождения таких решений. Как следствие получаем, что стандартный алгоритм локального улучшения с любым правилом замещения требует в худшем случае экспоненциального числа шагов. Нахождение равновесного решения при заданной начальной стратегии игроков оказывается PSPACE-полной задачей. Тем не менее, вопрос о трудоемкости нахождения равновесных решений, так же как и локальных оптимумом до сих пор остается открытым.

Список литературы

[1] Кочетов Ю.А., Пащенко М.Г., Плясунов А.В. О сложности локального поиска в задаче о p-медиане // Дискрет. анализ и исслед. операций. Серия 2. 2005. Т. 12, № 2. С. 44–71.

[2] Koutsoupias E., Papadimitriou C. Worst-case equilibria // Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science. Trier, Germany. 1999. 404-413.

[3] Vetta A. Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions // Proceedings of 43rd Annual IEEE symposiun on foundations of computer science, Vancouver, Canada. 2002. 416–425.

[4] Vredeveld T., Lenstra J. K. On local search for the generalized graph coloring problem // Oper. Res. Letters. 2003. V. 31, N. 4. P. 28–34.

[5] Yannakakis M. Computational complexity // Local search in combinatorial optimization. Chichester: Wiley, 1997. P. 19–55.

NASH EQUILIBRIA IN THE FACILITY LOCATION GAMES

Yu. Kochetov

Sobolev Institute of Mathematics,
Novosibirsk State University, Novosibirsk
e-mail: jkochet@math.nsc.ru

Abstract. We consider the discrete facility location games. The Nash equilibria are introduced and relationship with local optima are presented. We show that the equilibrium search problem belongs to the class PLS and is tightly complete in it.

Key words: Nash equilibrium, PLS complete problems, facility location games.