

## МЕТАЭВРИСТИЧЕСКИЕ МЕТОДЫ ОПТИМИЗАЦИИ В ЭКОНОМИКЕ

Метаэвристические методы набирают большую популярность в задачах оптимизации как методы, способные за приемлемый отрезок времени найти глобальный оптимум в пространстве поиска решений. В статье рассматриваются такие метаэвристические методы, как генетический алгоритм, кукушкин поиск, метод роя частиц, алгоритм светлячков, гармонический поиск, метод имитации отжига, поиск с запретами, жадный адаптивный метод случайного поиска. Приводится их обзор и сравнительный анализ. Делается вывод на основании проведенного обзора и сравнительного анализа о применимости данных методов в экономических задачах оптимизации.

*Ключевые слова:* метаэвристические методы, задача оптимизации, генетический алгоритм, кукушкин поиск, метод роя частиц, алгоритм светлячков, гармонический поиск, метод имитации отжига, поиск с запретами, жадный адаптивный метод случайного поиска.

Методы оптимизации направлены на нахождение такого способа распределения ресурсов, при котором обеспечивается минимальное или максимальное значение интересующего показателя. Они нашли широкое применение в экономике, поскольку множество экономических задач требуют оптимальное в рамках заданных критериев распределение имеющихся ресурсов для достижения результата. Следовательно, возникает задача оптимизации.

Однако большинство реальных задач оптимизации в экономике – нелинейные и мультимодальные, с различными сложными ограничениями, а их цели могут противоречить друг другу. При этом даже когда цель одна, оптимального решения может не существовать. То есть, нахождение оптимального решения или даже субоптимальных решений становится нелегким делом, и традиционные методы оптимизации зачастую для этого не подходят.

Одним из наиболее перспективных направлений для решения таких сложных задач, с которыми традиционные методы не справляются, являются так называемые метаэвристические методы.

Все метаэвристические методы используют определенный компромисс между рандомизацией и локальным поиском. И если качественные решения сложных проблем оптимизации могут быть найдены в разумные сроки, то нет никакой гарантии, что они окажутся оптимальными.

---

\* Алсагарова Евгения Владимировна – магистрант, кафедра информатики и кибернетики, Байкальский государственный университет экономики и права, г. Иркутск, [evimist@gmail.com](mailto:evimist@gmail.com).

Выделяют два основных компонента любого метаэвристического алгоритма: интенсификация и диверсификация, или разработка и исследование. Диверсификация означает генерацию разнообразных решений для того, чтобы исследовать пространство поиска в глобальном масштабе, в то время как интенсификация означает фокусирование поиска в одной локальной области, когда известно, что текущее лучшее решение находится там. Хороший баланс между интенсификацией и диверсификацией должен быть найден в ходе отбора лучших решений для увеличения скорости сходимости алгоритма. Выбор лучшего гарантирует то, что решения сходятся к оптимальному, в то время как диверсификация через рандомизацию позволяет поиску избегать локального оптимума и, в то же время, увеличивает разнообразие решений. Хорошее сочетание этих двух основных компонентов, как правило, обеспечивает достижимость глобального оптимума [14].

Согласно Пантелееву, метаэвристические методы можно разбить на следующие группы:

- 1) эволюционные методы (генетические алгоритмы, кукушкин поиск);
- 2) методы «роевого» интеллекта (метод роя частиц, алгоритм светлячков);
- 3) методы, имитирующие физические процессы (гармонический поиск, метод имитации отжига);
- 4) мультистартовые методы (поиск с запретами, жадный адаптивный метод случайного поиска) [10, с. 2].

Каждый метод обладает рядом особенностей, и для определения более узкого проведем обзор и сравнительный анализ этих методов.

Генетический алгоритм (ГА, с англ. genetic algorithm) – один из самых популярных метаэвристических методов. Название происходит от того, что в алгоритм работает по принципу механизма генетического наследования и естественного отбора. Среди основных понятий используются понятия хромосомы, кроссинговера (обмена генетического материала между хромосомами), мутации (получения нового генетического материала, предотвращающее «застревание» алгоритма в области локального оптимума).

Основные принципы работы генетического алгоритма заключены в следующем:

1. Генерация начальной популяции из  $n$  хромосом.
2. Вычисление для каждой хромосомы ее пригодности.
3. Выбор пары хромосом-родителей с помощью одного из способов отбора.
4. Проведение кроссинговера двух родителей с вероятностью  $p_c$ , произведение двух потомков.
5. Проведение мутации потомков с вероятностью  $p_m$ .
6. Повтор шагов 3–5, пока не будет сгенерировано новое поколение популяции, содержащее  $n$  хромосом.

7. Повтор шагов 2–6, пока не будет достигнут критерий окончания процесса [11, с. 13].

Генетические алгоритмы нашли широкое применение в экономических задачах, таких как задача оптимизации инвестиционного портфеля, создание оптимального маршрута, задача планирования, составление расписания, интеллектуальный анализ данных, выбор поставщиков, оценка активов, моделирование сложных адаптируемых систем типа конкуренции [13].

В кукушкином поиске (КП, с англ. cuckoo search) используются следующие основные предположения: каждое яйцо в гнезде – это решение, а яйцо кукушки в данном случае – новое решение. По аналогии с гнездовым паразитизмом реальных кукушек, яйца кукушек как новые и потенциально лучшие решения используются для замены менее хороших решений – других яиц в гнездах. Самая простая форма алгоритма – одно яйцо на гнездо. В более сложных задачах используется несколько яиц, то есть некоторое множество решений в каждом гнезде [9].

Кукушкин поиск основан на трех правилах:

1) Каждая кукушка откладывает одно яйцо за один раз и подкладывает его в случайно выбранное гнездо.

2) Лучшие гнезда с яйцами высокого качества переходят в следующее поколение.

3) Число доступных гнезд фиксировано, а яйцо кукушки может быть обнаружено птицей – хозяином гнезда – с вероятностью  $p_a \in (0,1)$ . В этом случае птица может либо избавиться от этого яйца, либо покинуть гнездо и построить новое.

Для осуществления глобального поиска кукушкин поиск использует полеты Леви: в отличие от обычного случайного блуждания, полеты Леви используют вероятность для большего разброса поиска в глобальном пространстве решений [7, с. 8].

Хотя метод появился только в 2009 году, уже есть разработки по его применению в таких задачах, как создание оптимального маршрута, составление расписания, выбор поставщиков, задача планирования для поточной линии, задача оптимизации производства, задача диспетчеризации, интеллектуальный анализ данных [7, с. 11].

Метод роя частиц (МРЧ, с англ. particle swarm optimization) – еще один метод, вдохновленный живой природой. В методе роя частиц каждая частица представляет собой возможное решение. Она характеризуется позицией в пространстве поиска и скоростью перемещения. Частицы перемещаются в пространстве согласно «принципу наилучшего найденного в пространстве положения, которое постоянно изменяется при нахождении частицами более выгодных положений». [9] Таким образом находится оптимум функции.

Метод роя частиц применяется в таких задачах, как создание оптимального маршрута, задача диспетчеризации, интеллектуальный анализ данных, задача портфельной оптимизации, задача планирования, составления расписания [6].

Алгоритм светлячков (АС, с англ. firefly algorithm) был впервые предложен Янгом в 2007 году и основан на модели поведения светлячков. Алгоритм светлячков основан на следующих трех правилах:

1) Все светлячки одного пола, поэтому один светлячок может быть привлечен другими светлячками независимо от их пола.

2) Привлекательность прямо пропорциональна яркости, и они оба уменьшаются с увеличением расстояния. Поэтому для двух светлячков менее яркий будет двигаться в сторону более яркого. Если для светлячка нет никого более яркого, чем он сам, то он перемещается случайным образом.

3) Яркость светлячка определяется значением фитнес-функции в его текущем положении [9].

Как и кукушкин поиск, алгоритм светлячков появился совсем недавно - в 2007 году, но уже есть разработки по его применению в таких задачах, как задача оптимизации бизнес процессов, портфельной оптимизации, создании оптимального маршрута, задача планирования, составление расписания, продвижение товаров в Интернете, задача диспетчеризации, интеллектуальный анализ данных [7, с. 18].

Классический гармонический поиск (ГП, с англ. harmony search) работает следующим образом. Музыкантам ставятся в соответствие индивиды  $s_i$ , а оркестру – популяция  $S = (s; i \in [1:|S|])$ . Аккорду, который берет музыкант  $s_i$  в данный момент времени, сопоставляется значение вектора варьируемых параметров  $X_i$ . Гармонию звуков формализует глобальный экстремум фитнес-функции  $f(X)$ . Совокупность текущих координат  $X_i, i \in [1:|S|]$ , образует так называемую  $(|S| \times |X|)$  матрицу памяти гармоний (harmony memory)  $H_M$ .

Схема гармонического поиска включает в себя следующие шаги:

- 1) инициализируем алгоритм;
- 2) формируем вектор гармонии;
- 3) выполняем пошаговую настройку вектора гармонии;
- 4) обновляем матрицу памяти гармоний;
- 5) если условие окончания итераций не выполнено, то возвращаемся к шагу 2 [9].

Гармонический поиск используется при составлении расписания, в задачах планирования, в теории игр, создании оптимального маршрута, интеллектуальном анализе данных [12].

Метод имитации отжига (с англ. simulated annealing) основан на процессе отжига, применяемом в металлургии, когда металл или сплав нагревают до высокой температуры, а затем охлаждают его до комнатной температуры.

В процессе алгоритма ищется точка или множество точек потенциальных решений, где достигается минимум функции. Алгоритм последовательно вычисляет точки заданного пространства и на каждом шаге после вычисления

точки понижает «температуру» – изначально положительную величину. При достижении нулевой температуры алгоритм останавливается.

С каждым шагом измеряется вероятность и не только учитываются изменения для улучшения целевой функции, но и сохраняются изменения, которые функцию не улучшают.

Точная настройка метода имитации отжига представляет собой довольно сложную задачу, поскольку от этого прямо зависит его эффективность для решения выбранной задачи [4].

Метод симуляции отжига применяется в задаче портфельной оптимизации, задачах планирования, составлении расписания, задаче о назначениях, интеллектуальном анализе данных, создании оптимального маршрута [3].

Поиск с запретами (ПЗ, англ. *tabu search*) оперирует понятием запрет (*tabu*). На каждом шаге алгоритм рассматривает множество решений по соседству с текущим решением и накладывает на некоторые из них запрет на выбор этого решения на данном шаге. Это позволяет алгоритму выйти за пределы локальных оптимумов. Лучшее решение выбирается среди незапрещенных до следующего шага.

Запреты могут накладываться разными способами, но главная цель – не возвращаться в локальный оптимум в течение определенного времени. Алгоритм хранит запреты в кратковременной памяти, то есть, на определенное время.

Особенно строгие запреты могут привести к «стагнации процесса поиска», поэтому иногда нужно запреты нарушать [14].

Поиск с запретами нашел широкое применение в экономических задачах: задачах планирования, составлении расписания, оптимизации производства, инвестирования, производства, планирование запасов, создании оптимального маршрута, производства точно в срок, в MRP системах, задачах о назначении [2].

Жадный адаптивный метод случайного поиска (с англ. *greedy randomized adaptive search procedure – GRASP*) состоит из итераций, созданных из конструкции жадного рандомизированного решения и последующих итерационных усовершенствований решения через локальный поиск. Жадные рандомизированные решения генерируются добавлением элементов в набор решений задачи из списка элементов, которые упорядочиваются с помощью жадной функции в зависимости от качества их решения. Для получения неоднородности в наборе кандидатов жадных решений упорядоченные элементы-кандидаты часто помещают в ограниченный список кандидатов (с англ. *restricted candidate list* или RCL) и случайным образом выбирают при построении решения [1].

GRASP используется в задачах планирования, оптимизации производства, составлении расписания, создания оптимального маршрута, задачах о назначении, распределении запасов, интеллектуальном анализе данных [5].

Сравнительный анализ по всем рассмотренным методам представлен в таблице.

Сравнительная таблица метаэвристических методов

Группа	Метод	Компоненты интенсификации и диверсификации	Достоинства	Недостатки
Эволюционные методы	ГА	Кроссинговер, мутация, селекция	Широко изучен	Сложность настройки параметров, относительно медленная сходимость
	КП	Рандомизация и лучшие гнезда	Использование полетов Леви	Мало изучен
Методы «роевого» интеллекта	МРЧ	Обновление параметров положения в пространстве и скорости	Широко изучен	Сложность настройки параметров
	АС	Обновление параметра положения в пространстве и привлекательность светлячка	Автоматическое разделение популяции на подгруппы	Мало изучен
Методы, имитирующие физические процессы	ГП	Настройка вектора гармонии и матрица памяти гармоний	Относительно быстрая сходимость	Сложность настройки параметров
	МИО	Поиск по соседству и понижение температуры	Широко изучен	Сложность настройки параметров, преждевременная сходимость
Мультистартовые методы	ПЗ	Поиск по соседству и список запретов	Широко изучен	Сложность настройки параметров, преждевременная сходимость
	GRASP	Рандомизация и ограниченный список кандидатов	Легкость настройки параметров	Нет учета истории решений из предыдущих итераций

При выборе метода следует учитывать, что алгоритмы могут показывать различную эффективность в зависимости от поставленной задачи.

Метаэвристические методы оптимизации представляют собой мощные средства для решения существующих проблем в экономике. Этому способствуют следующие факторы:

- Алгоритмы не делают никаких предположений (или не требуют априорной информации) о целевой функции.
- Они не требуют, чтобы целевая функция была непрерывной или дифференцируемой.
- Они справляются со сложными моделями с ограничениями.
- Почти все из них имеют варианты для решения непрерывных и комбинаторных задач.

– Практически все они имеют расширения для многокритериальных задач.

– Большинство из них поддерживают распараллеливание, что позволяет решать огромные по масштабу проблемы. [8]

Таким образом, можно сделать вывод, что метаэвристические методы, способные в огромном по масштабу пространстве решений находить оптимальные или близкие к оптимальным глобальные решения, используя малое количество ресурсов, подходят для решения большинства сложных задач оптимизации – задач, приближенных к реалиям современной экономики.

### **Список использованной литературы**

1. Feo T.A., Resende M.G.C. Greedy randomized adaptive search procedures / T.A. Feo, M. G. C. Resende // Journal of Global Optimization. – 1995. – № 6. – Pp 109–133.
2. Glover F., Laguna M. Tabu Search / F. Glover, M. Laguna. – Boston : Kluwer Academic Publishers, 1999. – 18 p.
3. Henderson D., Jacobson S. Y., Johnson A. W. The theory and practice of simulated annealing / D. Henderson, S. H. Jacobson, A. W. Johnson // Handbook of Metaheuristics. – Boston : Kluwer Academic Publishers, 2003. – Pp 287–319.
4. Kirkpatrick S., Gelatt C. D., Vecchi M. P. Optimization by simulated annealing / S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi // Science. – №220. – 1983. – Pp 671–680.
5. Resende M. G. C., Ribeiro C. C. Greedy randomized adaptive search procedures / M. G. C. Resende and C. C. Ribeiro // Handbook of Metaheuristics. – Boston : Kluwer Academic Publishers, 2003. – Pp 219–249.
6. Talukder S. Mathematical Modelling and Applications of Particle Swarm Optimization / S. Talukder. – Sweden, Blekinge : Blekinge Institute of Technology, 2011. – 59 p.
7. Yang X.-S. Cuckoo Search and Firefly Algorithm: Theory and Applications / X.-S. Yang. – New York : Springer, 2014. – 360 p.
8. Yilmaz A. E., Weber G.-W. Why You Should Consider Nature-Inspired Optimization Methods in Financial Mathematics / A. E. Yilmaz, G.-W. Weber // Non-linear and Complex Dynamics. – New York : Springer, 2011. – Pp 241–255.
9. Карпенко А. П. Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов / А. П. Карпенко // Информационные технологии. – 2012. – № 7. – С. 1–32.
10. Пантелеев А. В. Метаэвристические алгоритмы поиска условного экстремума: монография / А. В. Пантелеев. – М : МАИ-Принт, 2009. – 159 с.
11. Панченко Т. В. Генетические алгоритмы: учебно-методическое пособие / Т. В. Панченко. – Астрахань : Издательский дом «Астраханский университет», 2007. – 87 с.
12. Harmony Search Algorithm [Электронный ресурс] // <http://www.harmonysearch.info/>.

13. Kar A. Genetic Algorithm Applications [Электронный ресурс] / A. Kar // <http://business-fundas.com/2011/genetic-algorithm-applications/>.
14. Yang X.-S. Metaheuristic Optimization [Электронный ресурс] / X.-S. Yang // [http://www.scholarpedia.org/article/Metaheuristic\\_Optimization](http://www.scholarpedia.org/article/Metaheuristic_Optimization).