

Муравьиные алгоритмы



В последние годы интенсивно разрабатывается научное направление Natural Computing — «Природные вычисления», объединяющее математические методы, в которых заложены принципы природных механизмов принятия решений. Это механизмы обеспечивают эффективную адаптацию флоры и фауны к окружающей среде на протяжении миллионов лет. Имитация самоорганизации муравьиной колонии составляет основу муравьиных алгоритмов оптимизации — нового перспективного метода природных вычислений. Колония муравьев может рассматриваться как многоагентная система, в которой каждый агент (муравей) функционирует автономно по очень простым правилам. В противовес почти примитивному поведению агентов, поведение всей системы получается на удивление разумным.

Муравьиные алгоритмы серьезно исследуются европейскими учеными с середины 90-х годов. На сегодня уже получены хорошие результаты муравьиной оптимизации таких сложных комбинаторных задач, как: задачи коммивояжера, задачи оптимизации маршрутов грузовиков, задачи раскраски графа, квадратичной задачи о назначениях, оптимизации сетевых графиков, задачи календарного планирования и других. Особенно эффективны муравьиные алгоритмы при on-line оптимизации процессов в распределенных нестационарных системах, например трафиков в телекоммуникационных сетях.

Задача

Комбинаторные задачи оптимизации: задача коммивояжера, квадратичная задача о назначениях, задача маршрутизации грузовиков, задача календарного планирования, задача раскраски графа и др.

Программные средства

MATLAB

Результаты

Приведены теоретические основы муравьиных алгоритмов оптимизации. Описано решение муравьиными алгоритмами задачи коммивояжера. Проанализированы применения муравьиных алгоритмов.

1. Введение

В последние два десятилетия при оптимизации сложных систем исследователи все чаще применяют природные механизмы поиска наилучших решений. Это механизмы обеспечивают эффективную адаптацию флоры и фауны к окружающей среде на протяжении миллионов лет. Сегодня интенсивно разрабатывается научное направление Natural Computing — «Природные вычисления», объединяющее методы с природными механизмами принятия решений, а именно:

- Genetic Algorithms — генетические алгоритмы;
- Evolution Programming — эволюционное программирование;
- Neural Network Computing — нейро-сетевые вычисления;
- DNA Computing — ДНК-вычисления;
- Cellular Automata — клеточные автоматы;
- Ant Colony Algorithms — муравьиные алгоритмы.

Целью настоящей статьи является изложение теоретических основ и примеров практического применения муравьиных алгоритмов — нового перспективного метода оптимизации, базирую-

щегося на моделировании поведения колонии муравьев. Колония муравьев может рассматриваться как многоагентная система, в которой каждый агент (муравей) функционирует автономно по очень простым правилам. В противовес почти примитивному поведению агентов, поведение всей системы получается на удивление разумным.

Муравьиные алгоритмы серьезно исследуются европейскими учеными с середины 90-х годов. На сегодня уже получены хорошие результаты муравьиной оптимизации таких сложных комбинаторных задач, как: задачи коммивояжера, задачи оптимизации маршрутов грузовиков, задачи раскраски графа, квадратичной задачи о назначениях, оптимизации сетевых графиков, задачи календарного планирования и других. Несмотря на стремительные успехи муравьиных алгоритмов, подавляющее большинство русскоязычных специалистов по исследованию операций не знакомы с этой технологией оптимизации.

Статья состоит из трех частей: в первой части излагается теория муравьиных алгоритмов, во второй — описывается решение муравьиными алгоритмами задачи о коммивояжере, в третьей части дается обзор применения муравьиных алгоритмов. Теоретическая часть статьи базируется на книгах [1, 2], лекции изобретателя муравьиных алгоритмов доктора Марко Дориги в летней школе по сложным системам [3] и материалах сайта [4].

2. Принципы поведения муравьев

Принципы поведения муравьев выдержали испытания далеко не в лабораторных условиях на протяжении 100 миллионов лет — именно столько времени назад муравьи «колонизировали» Землю. Муравьи относятся к *социальным* насекомым, живущим внутри некоторого коллектива — колонии. На Земле около двух процентов насекомых являются социальными, половину из них составляют муравьи — небольшие существа массой от 1 до 5 мг.

Число муравьев в одной колонии колеблется от 30 штук до нескольких миллионов. На Земле около 10^{16} муравьев с общей массой, приблизительно равной массе человечества. Поведение муравьев при транспортировании пищи, преодолении препятствий, строительстве муравейника и других действиях зачастую приближается к теоретически оптимальному. В качестве примера на

рис. 1 приведена структура взаимосвязанных гнезд суперколонии муравьев *Formica lugubris* в Швейцарии. Сеть муравейников близка к минимальному остовному дереву, соединяющему все гнезда колонии – вершины графа на рис. 1.

Какие же механизмы обеспечивают столь сложное поведение муравьев, и что можем мы позаимствовать у этих крошечных существ для решения своих глобальных задач? Основу «социального» поведения муравьев составляет *самоорганизация* — множество динамических механизмов, обеспечивающих достижение системой глобальной цели в результате низкоуровневого взаимодействия ее элементов. Принципиальной особенностью такого взаимодействия является использование элементами системы *только локальной информации*. При этом исключается любое централизованное управление и обращение к глобальному образу, репрезентирующему систему во внешнем мире. Самоорганизация является результатом взаимодействия следующих четырех компонентов:

- случайность;
- многократность;
- положительная обратная связь;
- отрицательная обратная связь.

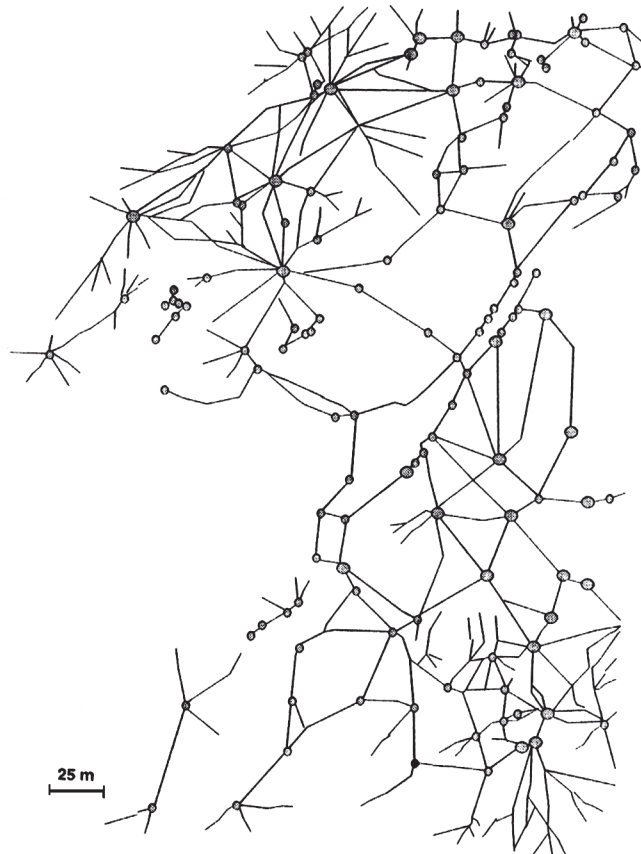
Муравьи используют два способа передачи информации: прямой — обмен пищей, мандибулярный, визуальный и химический контакты, и непрямой — стигмержи (stigmergy). *Стигмержи* — это разнесенный во времени тип взаимодействия, когда один субъект взаимодействия изменяет некоторую часть окружающей среды, а остальные используют информацию об ее состоянии позже, когда находятся в ее окрестности. Биологически стигмержи осуществляется через *феромон* (pheromone) — специальный секрет, откладываемый как след при перемещении муравья. Феромон — достаточно стойкое вещество, он может восприниматься муравьями несколько суток. Чем выше концентрация феромона на тропе, тем больше муравьев будет по ней двигаться. Со временем феромон испаряется, что позволяет муравьям адаптировать свое поведение под изменения внешней среды. Распределение феромона по пространству передвижения муравьев является своего рода динамически изменяемой глобальной памятью муравейника. Любой муравей в фиксированный момент времени может воспринимать и изменять лишь одну локальную ячейку этой глобальной памяти.

Муравьиные алгоритмы основаны на имитации природных механизмов самоорганизации муравьев, использование которых иллюстрируется в следующих разделах на примере оптимизации маршрута коммивояжера.

3. Муравьиный подход к задаче коммивояжера

Задача коммивояжера состоит в поиске кратчайшего замкнутого маршрута, проходящего через все города ровно один раз. Выбор задачи коммивояжера для иллюстрации идей муравьиных алгоритмов обусловлен следующим.

1. Задача наглядно интерпретируется в терминах поведения муравьев — перемещения коммивояжера и муравьев интуитивно сопоставимы.
2. Это NP-сложная задача, которая на недетер-



▲ Рис. 1. Сеть гнезд суперколонии муравьев *Formica lugubris* в Швейцарии [9].

минированной машине Тьюринга решается за полиномиальное время.

3. Это традиционный тестовый полигон (benchmark problem) для методов комбинаторной оптимизации. Существует обширная база тестовых задач о коммивояжере и методов их решения, что позволяет сравнить эффективность муравьиных алгоритмов оптимизации с другими подходами.

4. Это дидактическая задача, для которой можно легко, без злоупотребления техническими подробностями алгоритма объяснить процесс нахождения оптимального решения.

5. Это первая комбинаторная задача, решенная муравьиными алгоритмами.

Рассмотрим, как реализовать четыре составляющие самоорганизации муравьев при оптимизации маршрута коммивояжера. Многократность взаимодействия реализуется итерационным поиском маршрута коммивояжера одновременно несколькими муравьями. При этом каждый муравей рассматривается как отдельный, независимый коммивояжер, решающий свою задачу. За одну итерацию алгоритма каждый муравей совершает полный маршрут коммивояжера.

Положительная обратная связь реализуется как имитация поведения муравьев типа «оставление следов–перемещение по следам». Чем больше следов оставлено на тропе — ребре графа в задаче коммивояжера, тем больше муравьев будет передвигаться по ней. При этом на тропе появляются новые следы, привлекающие дополнительных муравьев. Для задачи коммивояжера положительная обратная связь реализуется следующим стохастическим правилом: вероятность включения ребра графа в маршрут муравья пропорциональна количеству феромона на нем.

Применение такого вероятностного правила обеспечивает реализацию и другой составляющей самоорганизации — случайности. Количество откладываемого муравьем феромона на ребре графа обратно пропорционально длине маршрута. Чем короче маршрут, тем больше феромона будет отложено на соответствующих ребрах графа и тем больше муравьев будет использовать их при синтезе своих маршрутов. Отложенный на ребрах феромон выступает как усилитель, он позволяет хорошим маршрутам сохраняться в глобальной памяти муравейника. Эти маршруты могут быть улучшены на последующих итерациях алгоритма.

Использование только положительной обратной связи приводит к преждевременной сходимости решений — к случаю, когда все муравьи двигаются одним и тем же субоптимальным маршрутом. Для избежания этого используется отрицательная обратная связь — испарение феромона. Время испарения не должно быть слишком большим, так как при этом возникает опасность сходимости популяции маршрутов к одному субоптимальному решению. С другой стороны, время испарения не должно быть и слишком малым, так как это приводит к быстрому «забыванию», потере памяти колонии и, следовательно, к некооперативному поведению муравьев. В поведении муравьев кооперативность является очень важной: множество идентичных муравьев одновременно исследуют разные точки пространства решений и передают свой опыт через изменения ячеек глобальной памяти муравейника.

Для каждого муравья переход из города i в город j зависит от трех составляющих: памяти муравья (*tabu list*), видимости и виртуального следа феромона.

- *Tabu list* (память муравья) — это список посещенных муравьем городов, заходить в которые еще раз нельзя. Используя этот список, муравей гарантированно не попадет в один и тот же город дважды. Ясно, что *tabu list* возрастает при совершении маршрута и обнуляется в начале каждой итерации алгоритма. Обозначим через $J_{i,k}$ список городов, которые еще необходимо посетить муравью k , находящемуся в городе i . Понятно, что $J_{i,k}$ является дополнением к *tabu list*.

- *Видимость* — величина, обратная расстоянию: $\eta_{ij} = 1/D_{ij}$, где D_{ij} — расстояние между городами i и j . Видимость — это локальная статическая информация, выражающая эвристическое желание посетить город j из города i — чем ближе город, тем больше желание посетить его. Использование только видимости, конечно, является недостаточным для нахождения оптимального маршрута.

- *Виртуальный след феромона* на ребре (i, j) представляет подтвержденное муравьиным опытом желание посетить город j из города i . В отличие от видимости след феромона является более глобальной и динамичной информацией — она изменяется после каждой итерации алгоритма, отражая приобретенный муравьями опыт. Количество виртуального феромона на ребре (i, j) на итерации t обозначим через $\tau_{ij}(t)$.

Важную роль в муравьиных алгоритмах играет вероятностно-пропорциональное правило, определяющее вероятность перехода k -го муравья

из города i в город j на t -й итерации:

$$P_{ij,k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_{i,k}} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta}, & \text{если } j \in J_{i,k}, \\ P_{ij,k}(t) = 0, & \text{если } j \notin J_{i,k}, \end{cases} \quad (1)$$

где α и β — два регулируемых параметра, задающие веса следа феромона и видимости при выборе маршрута. При $\alpha = 0$ будет выбран ближайший город, что соответствует жадному алгоритму в классической теории оптимизации. Если $\beta = 0$, тогда работает лишь феромонное усиление, что влечет за собой быстрое вырождение маршрутов к одному субоптимальному решению.

Обратим внимание, что правило (1) определяет лишь вероятности выбора того или иного города. Собственно выбор города осуществляется по принципу «колеса рулетки»: каждый город на ней имеет свой сектор с площадью, пропорциональной вероятности (1). Для выбора города нужно бросить шарик на рулетку — сгенерировать случайное число, и определить сектор, на котором этот шарик остановится.

Заметим, что хотя правило (1) не изменяется на протяжении итерации, значения вероятностей $P_{ij,k}(t)$ для двух муравьев в одном и том же городе могут отличаться, т. к. $P_{ij,k}(t)$ — функция от $J_{i,k}$ — списка еще не посещенных городов муравьем k .

После завершения маршрута каждый муравей k откладывает на ребре (i, j) такое количество феромона:

$$\Delta\tau_{ij,k}(t) = \begin{cases} \frac{Q}{L_k(t)}, & \text{если } (i, j) \in T_k(t), \\ 0, & \text{если } (i, j) \notin T_k(t), \end{cases}$$

где $T_k(t)$ — маршрут, пройденный муравьем k на итерации t ; $L_k(t)$ — длина этого маршрута; Q — регулируемый параметр, значение которого выбирают одного порядка с длиной оптимального маршрута.

Для исследования всего пространства решений необходимо обеспечить испарение феромона — уменьшение во времени количества отложенного на предыдущих итерациях феромона. Обозначим коэффициент испарения феромона через $p \in [0, 1]$. Тогда правило обновления феромона примет вид:

$$\tau_{ij}(t+1) = (1-p) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t), \quad (2)$$

где $\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij,k}(t)$, m — количество муравьев в колонии.

В начале оптимизации количество феромона принимается равным небольшому положительному числу τ_0 . Общее количество муравьев в колонии остается постоянным на протяжении выполнения алгоритма. Многочисленная колония приводит к быстрому усилению субоптимальных маршрутов, а когда муравьев мало, возникает опасность потери кооперативности поведения через ограниченное взаимодействие и быстрое испарения феромона. Обычно число муравьев назначают равным количеству городов — каждый муравей начинает маршрут со своего города.

Для улучшения временных характеристик муравьиного алгоритма вводят так называемых элитных муравьев. Элитный муравей усиливает ребра наилучшего маршрута, найденного с начала работы алгоритма. Количество феромона, откладываемого на ребрах наилучшего текущего маршрута T^+ , принимается равным Q/L^+ , где L^+ — длина маршрута T^+ . Этот феромон побуждает муравьев к исследованию решений, содержащих несколько ребер наилучшего на данный момент маршрута T^+ . Если в муравейнике есть e элитных муравьев, то ребра маршрута T^+ будут получать общее усиление

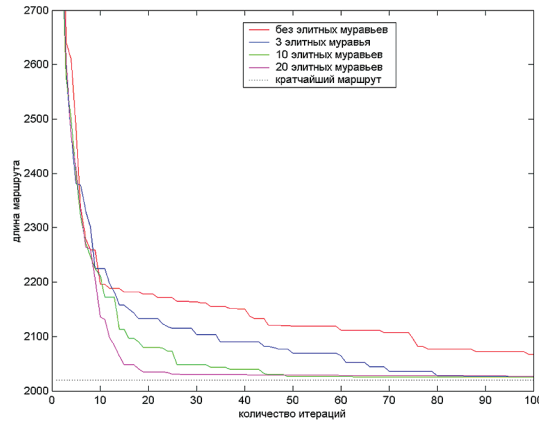
$$\Delta\tau_e = e \cdot Q/L^+ \quad (3)$$

4. Муравьиная оптимизация маршрута коммивояжера

Ниже приводится муравьиный алгоритм оптимизации маршрута коммивояжера, реализующий все идеи предыдущего раздела (в угловых скобках содержится словесное описание соответствующего кода).

```
<Ввод матрицы расстояний D>
<Инициализация параметров алгоритма  $\alpha$  (alpha),
 $\beta$  (beta), e, Q,  $\tau_0$  (tau0)>
m=n % Количество муравьев равно числу городов
For i=1:n
  For j=1:n % Для каждого ребра
    If i=j
      eta(i,j)=1/D(i,j); % Видимость
      tau(i,j)= tau0; % Феромон
    Else tau(i,j)=0;
  % Переход из одного города в тот же самый
  % невозможен
  End
End
End
For k=1:m
  <Разместить муравья k в случайно выбранный город>
  End
  <Выбрать условно-кратчайший маршрут  $T^+$  и
  рассчитать его длину  $L^+$ >
  % Основной цикл
  For t=1:tmax
    % tmax - количество итераций алгоритма
    For k=1:m % Для каждого муравья
      <Построить маршрут  $T_k(t)$  по правилу (1) и
      рассчитать длину  $L_k(t)$ >
      End
    If < "Лучшее решение найдено?" >
      <Обновить  $T^+$  и  $L^+$ >
      End
    For i=1:n
      For j=1:n % Для каждого ребра
        <Обновить следы феромона по правилам (2) и
        (3).>
        End
      End
    End
  <Вывести кратчайший маршрут  $T^+$  и его длину  $L^+$ >
```

Муравьиный алгоритм был запрограммирован в системе MATLAB и протестирован на серии задач из библиотеки [5]. Для задачи с 29 населенными пунктами в Баварии «Bays-29» алгоритм без элитных муравьев после 100 итераций нашел оптимальный маршрут длиной 2020 только в одном случае из пяти. Решение можно улучшить простым увеличением количества итераций до 1-2 тысяч. Длины маршрутов муравьев на одной итерации отличаются незначительно, поэтому для ускорения нахождения оптимума необходимо искусственно усиливать наилучшие текущие решения с помощью элитных муравьев. На рис. 2



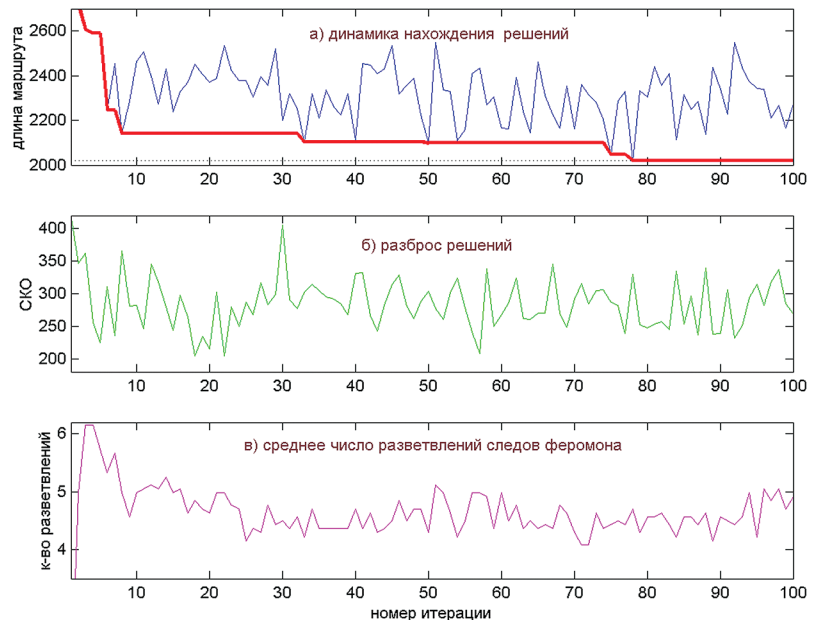
◀ Рис. 2. Сравнение эффективности алгоритмов с разным числом элитных муравьев.

показаны усредненные динамики решения задачи «Bays-29» алгоритмами с различным числом элитных муравьев.

Эксперименты для каждого алгоритма повторялись 5 раз. На первый взгляд кажется, что чем больше элитных муравьев, тем лучше работает алгоритм. Действительно, алгоритмы с большим количеством элитных муравьев очень быстро, за 30-40 итераций находят субоптимальные маршруты длиной 2033, 2028, 2026, однако после этого надолго застревают в локальных оптимумах, т. к. элитные муравьи очень сильно усиливают эти субоптимальные решения. В пяти экспериментах за 100 итераций алгоритм с тремя элитными муравьями трижды нашел оптимальный маршрут, с десятью элитными муравьями — дважды, а с двадцатью — только один раз. Несмотря на то, что алгоритм с тремя элитными муравьями медленнее приближается к хорошим маршрутам, он намного быстрее проходит субоптимальные решения-ловушки.

Проведенные нами эксперименты свидетельствуют, что популяция решений никогда не вырождается к одному, общему для всех муравьев маршруту. Наоборот, алгоритм продолжает синтезировать новые, возможно лучшие решения. На рис. 3, а синей линией показаны наилучшие решения, найденные на каждой итерации алгоритма с тремя элитными муравьями. Красной линией показаны наилучшие решения, найденные с начала работы алгоритма. Как видно, линии не совпадают.

▼ Рис. 3. Исследование процесса решения задачи коммивояжера муравьиным алгоритмом.



ют, следовательно, муравьиный алгоритм генерирует новые решения на каждой итерации. Об этом свидетельствует и рис. 3, б, на котором показано среднеквадратическое отклонение длин маршрутов, найденных муравьями на текущей итерации алгоритма.

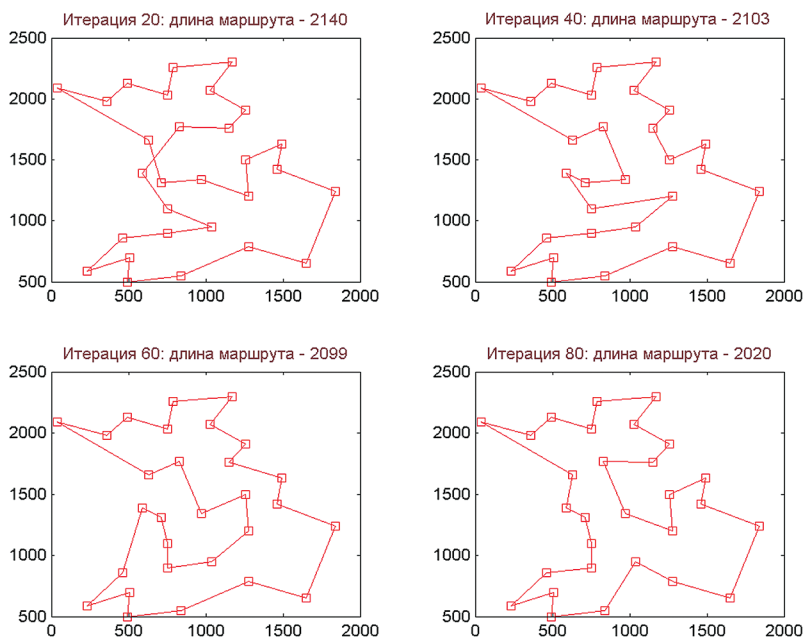
Даже если оптимальный маршрут уже найден, алгоритм продолжает поддерживать разнообразие популяции решений. На рис. 3, в показано среднее по городам число разветвлений следов феромона на текущей итерации алгоритма. Оно получено подсчетом ребер, инцидентных вершине графа, след феромона на которых превышает некоторый порог. Это число колеблется около 4–5 на протяжении всей работы алгоритма, следовательно, в любом городе для муравья существует несколько перспективных альтернатив продолжения маршрута. Эволюция маршрутов коммивояжера, найденных алгоритмом с тремя элитными муравьями, показана на рис. 4.

По сравнению с точными методами, например динамическим программированием или методом ветвей и границ, муравьиный алгоритм находит близкие к оптимуму решения за значительно меньшее время даже для задач небольшой размерности ($n > 20$). Время оптимизации муравьиным алгоритмом является полиномиальной функцией от размерности $O(t \cdot n^2 \cdot m)$, тогда как для точных методов зависимость экспоненциальная.

5. Обзор применения муравьиных алгоритмов оптимизации

Приведенный в статье муравьиный алгоритм оптимизации маршрута коммивояжера после незначительных модификаций может использоваться для решения различных комбинаторных задач: квадратичной задачи о назначениях (Quadratic Assignment Problem), задачи об оптимизации маршрутов грузовиков (Vehicle Routing Problem), задачи календарного планирования (Job-Shop Schedule Planing), задачи раскраски графа (Graph Coloring Problem) и др. Муравьиные алгоритмы находят решения дискретных задач оптимизации не хуже других общих метаэвристических технологий и некоторых проблемно-ориентированных методов.

Рис. 4. Эволюция маршрутов коммивояжера для задачи «Bays-29».



В табл. 1 приведены длины маршрутов коммивояжера, найденные различными эвристическими методами оптимизации. В табл. 2 сравниваются методы решения квадратичной задачи о назначениях. В ячейках таблицы указаны стоимости решений. В табл. 1 и 2 число в названии тестовой задачи указывает ее размерность. В табл. 3 сравниваются три метода оптимизации маршрутов грузовиков для задач большой размерности. Время решения задач пересчитано для компьютера с процессором Pentium-900MHz. Во всех таблицах полужирным шрифтом выделены наилучшие на сегодняшний день решения.

Высокую эффективность муравьиные алгоритмы демонстрируют при оптимизации распределенных нестационарных систем. Ярким примером может служить нахождение муравьиными алгоритмами оптимальных трафиков в телекоммуникационных сетях [1, 7]. В качестве примера в табл. 4 приведены результаты маршрутизации в американской сети NSFNET, содержащей 14 узлов с 21 двунаправленной линией связи.

Сравнивались следующие алгоритмы: AntNet — муравьиный алгоритм; OSRF — официальный интернетовский алгоритм маршрутизации; SRF — алгоритм, использующий динамическую метрику при оценке стоимости соединений; Daemon — аппроксимация идеального алгоритма маршрутизации; BF — алгоритм Беллмана-Форда. В таблице приведены средние значения времени задержки и пропускной способности при интенсивной загрузке сети. Числа в скобках — значения среднеквадратических отклонений при 10-кратном прогоне алгоритмов. Детальную информацию об этих и других применениях муравьиных алгоритмов можно найти на сайтах [4, 8].

6. Выводы

Муравьиные алгоритмы основаны на имитации самоорганизации социальных насекомых посредством использования динамических механизмов, с помощью которых система достигает глобальной цели в результате локального низкого уровня взаимодействия элементов. В статье на примере задачи коммивояжера показано, как в алгоритмы решения дискретных задач оптимизации внедрить составляющие самоорганизации муравьев: случайность, многократность взаимодействия, отрицательную и положительную обратные связи. Проведенные компьютерные эксперименты показывают, что муравьиные алгоритмы находят хорошие маршруты коммивояжера значительно быстрее, чем точные методы комбинаторной оптимизации. Эффективность муравьиных алгоритмов увеличивается с ростом размерности задачи оптимизации.

Муравьиные алгоритмы обеспечивают решения и других комбинаторных задач не хуже общих метаэвристических технологий оптимизации и некоторых проблемно-ориентированных методов. Особенно хорошие результаты муравьиной оптимизации получаются для нестационарных систем, параметры которых изменяются во времени, например телекоммуникационных и компьютерных сетей.

Важным свойством муравьиных алгоритмов является неконвергентность: даже после большого числа итераций одновременно исследуется

множество вариантов решения, вследствие чего не происходит длительных временных задержек в локальных экстремумах. Все это позволяет рекомендовать применение муравьиных алгоритмов для решения сложных комбинаторных задач оптимизации. Перспективными путями улучшения муравьиных алгоритмов являются on-line адаптация параметров с помощью базы нечетких правил, а также их гибридизация с другими методами природных вычислений, например генетическими алгоритмами. Гибридизация может осуществляться по островной схеме, когда различные алгоритмы решают задачу параллельно и автономно (каждый на отдельном «острове») с обменом наилучшими решениями через определенное время, или по принципу «мастер-подмастерье», когда основной алгоритм — «мастер» передает решение типовых подзадач «подмастерью» — специализированному, быстрому алгоритму.

Литература

1. Bonavear E., Dorigo M. Swarm Intelligence: from Natural to Artificial Systems.— Oxford University Press, 1999.— 307 p.
2. Corne D., Dorigo M., Glover F. New Ideas in Optimization.— McGraw-Hill, 1999.
3. Dorigo M. Swarm Intelligence, Ant Algorithms and Ant Colony Optimization // Reader for CEU Summer University Course «Complex System».— Budapest, Central European University, 2001.— P. 1–38.

4. <http://irida.ulb.ac.de/ACO/ACO.html>.
5. <http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html>.
6. Reimann M. Ant Based Optimization in Good Transportation. PhD Thesis. University of Vienna.— Vienna, Austria, 2002.— 149 p.
7. Caro G. D., Dorigo M. Anet: a Mobile Agents Approach to Adaptive Routing, Technical Report IRIDA 97-12. IRIDA — Universite Libre de Bruxelles.— Brussels, Belgium, 1997.— 27 p.
8. <http://www.swarm.org>.
9. Cherix D. Note preliminaire sur la structure, la phenologie et le regime alimentaire d'une super-colonie de Formica lugubris Zett. // Insects Sociaux 27, 1980.— P. 226–236.



Автор:

Штовба Сергей Дмитриевич, доцент, кандидат технических наук, доцент кафедры компьютерных систем управления; Винницкий государственный технический университет, г. Винница, Украина



Таблица 1

Сравнение эвристических методов оптимизации маршрута коммивояжера [1]

Тестовая задача	Eil50	Eil75	KroA100
Имитационный отжиг	443	580	нет данных
Генетические алгоритмы	428	545	22761
Эволюционное программирование	426	542	нет данных
Муравьиные алгоритмы	425	535	21282

Таблица 2

Сравнение эвристических методов решения квадратичной задачи о назначениях [1]

Тестовая задача	Nugent (12)	Nugent (15)	Nugent (20)	Nugent (30)	Elsbafei (19)	Kraup (30)
Имитационный отжиг	578	1150	2570	6128	17937024	89800
Табу-поиск	578	1150	2570	6124	17212548	90090
Генетические алгоритмы	588	1160	2688	6748	17640584	108830
Эволюционные стратегии	598	1168	2654	6308	196000212	97880
Муравьиные алгоритмы с локальным поиском	578	1150	2570	6128	17212548	88900

Таблица 3

Сравнение эвристических методов оптимизации маршрутов грузовиков [6]

Число заказчиков	Муравьиный алгоритм			Генетический алгоритм		Гранулированный табу-поиск	
	Средняя за 10 прогонов длина маршрутов	Длина маршрутов для лучшего решения	Время решения, мин	Длина маршрутов для лучшего решения	Время решения, мин	Длина маршрутов для лучшего решения	Время решения, мин
200	6460.98	6460.98	7.13	6460.98	1.04	6697.53	2.38
255	589.28	586.87	139.27	596.89	14.32	593.35	11.67
300	1007.81	1007.07	32.55	1018.74	39.33	1016.83	21.45
399	932.58	927.27	158.93	933.74	78.5	936.04	33.12
420	1836.87	1834.79	239.47	1846.55	210.42	1915.83	43.05
480	13958.68	13816.98	240.00	13728.8	187.6	14910.62	15.13

Таблица 4

Сравнение алгоритмов маршрутизации для сети NSFNET [7]

Алгоритм	AntNet	OSRF	SRF	Daemon	BF
Средняя задержка передачи сообщения, с	0.93 (0.2)	5.85 (1.43)	3.58 (0.83)	0.10 (0.03)	4.27 (1.22)
Пропускная способность, $\times 10^7$ бит/с	2.392 (0.011)	2.100 (0.002)	2.284 (0.003)	2.403 (0.010)	1.410 (0.047)