УДК 519.854

Б. И. Юхименко, О. Ю. Ткаленко
Одесский национальный политехнический университет
Проспект Шевченко, 1, 65044 Одесса, Украина

Алгоритм муравьиной колонии для многомерной задачи о ранце

Приведена модификация муравьиного алгоритма решения многомерной задачи о ранце. Также приведен обзор применяемости муравьиных алгоритмов в различных предметных областях. Все алгоритмы этого типа являются приближенными вероятностными алгоритмами. Эффективность работы алгоритмов зависит от параметров α и β, предопределяющих количество феромонов при передвижении муравьев, а также их испарение соответственно. Приведены формулы расчета величины вероятности, согласно которой принимается решение о присвоении значения «1» компоненте вектора решений. Приведены результаты компьютерного экспериментального исследования. Результаты сравнения точных решений с решением предложенным алгоритмом подчеркивают его эффективность.

Ключевые слова: *методы оптимизации, алгоритм муравьиной колонии, многомерная задача о ранце, приближенные вероятностные алгоритмы.*

Введение

Задачи дискретной оптимизации часто встречаются в экономике, технике, социальной и других предметных областях деятельности человека. Разработано множество методов решения практической реализации дискретных моделей. Однако все точные методы дискретной оптимизации относятся к классу NP-сложности вычислений. Задачи большой размерности зачастую остаются неразрешимыми из-за большого количества итераций и самой сложности вычислений. В связи с этим, приближенные и вероятностные приближенные алгоритмы и их разработка должны занять достойное место в исследованиях по этому направлению.

В последнее время пристальное внимание уделяется поведению живых существ и процессов, происходящих в природе. Это используется в качестве идеи при создании методов и алгоритмов решения оптимизационных задач.

Действительно, природа создает многие объекты и организует процессы оптимально или близко к нему. Никому не удается найти лучшее расположение емкостей, как это делают пчелы, создавая соты для хранения меда. Можно найти множество таких примеров, когда объекты, созданные природой, не могут быть усовершенствованы интеллектуалами.

Лет двадцать тому назад, появилось направление под названием муравьиная оптимизация. Это новые подходы, которые используются при разработке приближенных алгоритмов решения многих комбинаторных экстремальных задач. Всеобщее признание получило направление после присвоения премии фонда Марии Кюри итальянскому ученому Марко Доринго за успехи в создании муравьиных алгоритмов.

Хотя стремительное обсуждение муравьиной оптимизации стартовало только в начале XXI-го века, на сегодняшний день успешно решаются задачи о коммивояжере с большим количеством городов, задачи календарного планирования, маршрутизации транспортных средств и многие другие.

Многомерная задача о ранце является одной из часто используемых математических моделей. К ней приводятся многие экономические, технические и другие проблемы. Методы решений многомерной задачи относятся к классу NP-сложности вычислений. Поэтому разработка или хотя бы попытка получить алгоритмы более простой сложности, а возможно сложности, скорость сходимости которых, оценивается полиномиально, является весьма *актуальным* моментом в дискретной оптимизации.

Муравьиные колонии и поведение муравьев

Внимание к поведению муравьев объясняется тем, что этих социальных насекомых, живущих колониями или семьями, можно рассматривать как *многоагентную систему*. Каждый муравей (агент) функционирует отдельно по очень простым правилам. Однако они объединяются для достижения общей цели. Подтверждение этому — архитектурно сложные муравейники с входами и выходами, помещениями для хранения пищи и др. [1].

Муравьи обладают способами *передачи информации*. Тропинки, по которым они передвигаются, помечаются особым образом, оставляя на них химическое вещество — феромоны. «Чем выше концентрация феромонов на тропе, тем больше муравьев будут по ней двигаться» [2]. В работе [2] приведено описание лабораторного эксперимента о поведении муравьев при поиске кратчайшего пути от источника пищи к гнезду. Изначально предполагались пути различной длины. По какому из них будут передвигаться муравьи, выбирали *случайным образом*. Интенсивность передвижения была одинаковой и не зависела от длины пути. Однако муравьи, находя *кратчайший путь.* При помощи феромонов, передают информацию о нем. Другие члены колонии начинают передвижение по короткой дороге.

Распределение феромонов является своего рода *памятью* для муравьев. Однако отложенная в памяти информация *динамически меняется*. Со временем *феромоны испаряются*, что позволяет муравьям адаптироваться, менять свое поведение с учетом изменения внешней среды.

Такие принципы как случайность, многократность, положительная и отрицательная обратная связи, предопределяют самоорганизованность поведения муравьев. На наш взгляд, это и есть основой при математизации поведения многочисленных насекомых и создании правдоподобных наборов действий — алгоритмов, имеющих мотивацию и целенаправленность.

Принципы формализации поведения муравьев

Математическое представление многократности в известных муравьиных алгоритмах реализуется путем самостоятельного решения задачи несколькими агентами (муравьями) отдельно, независимо друг от друга. К примеру, в [4] под названием «многие коммивояжеры» приводится алгоритм решения задачи о коммивояжере. В принципе, все алгоритмы, созданные на основе идей о муравьиной оптимизации, предусматривают многократное одновременное решение задачи одинаковым способом. Это дает возможность накопить информацию и использовать ее в дальнейшем. Математическая оценка наличия количества феромонов определяется через накопленную статистическую информацию. В каждом алгоритме эта информация учитывается по-разному (см. [2] и [3]), но количественное значение параметра α непосредственно зависит от количества «проходов» муравьев.

Уменьшение количества феромонов — испарение (параметр β) предопределяет возможность поступать в дальнейшем иначе, то есть «изменить тропу» при достижении цели.

Недетерминированное изменение количества феромонов предопределяет случайный выбор передвижения муравьев. Математически описывается при помощи *вероятности*. Величина вероятности зависит от параметров α и β. Вид зависимости — это собственное виденье создателя алгоритма.

Существующие подходы компоновки решений

Муравьиные алгоритмы оптимизации относятся к классу приближенных вероятностных алгоритмов. Получение варианта решения принято называть итерацией. Поскольку одновременно работает несколько агентов — искусственных муравьев, то за одну итерацию получаем столько допустимых решений, сколько муравьев выполняли работу. Процесс получения допустимого решения представляется как непосредственная его компоновка. На каждом шаге конкретизируется одна компонента вектора решений. Это было предложено в методе последовательного построения решений [5]. В детерминированных алгоритмах номер конкретизируемой компоненты выбирается по-разному. Скажем, в жадных алгоритмах — согласно величине компонент вектора стоимостей. Приоритет отдается той компоненте, которая внесет наибольший (наименьший) вклад в значение целевой функции [6]. В алгоритмах генетического типа очередность конкретизации определяется частотой появления положительного значения компоненты в нескольких лучших решениях [4]. В работах [7, 8] приводится комбинированный подход при присвоении приоритета компоненте. Любая дополнительная работа по упорядочению компонент вектора решений вносит позитивный вклад в оценку скорости сходимости комбинаторных алгоритмов. Сама идея упорядочения компонент появилась достаточно давно. Алгоритм Балаша, представляющий метод ветвей и границ с односторонним ветвлением, использовал это в качестве основного подхода при частичном переборе вариантов. В методах ветвей и границ с двусторонним ветвлением идея алгоритма Балаша использовалась неоднократно, в результате чего были получены хорошие результаты (см. напр. [9]).

В муравьиных алгоритмах номер конкретизируемой компоненты на каждом шаге работы алгоритма выбирается случайно, пропорционально величине вероятности. Количественное значение вероятности зависит от параметров α и β. Другими словами, зависит от накопления феромонов и их испарения. В работе [3] экспериментально показано, что величина погрешности приближенного решения зависит от β.

Данная работа продолжает исследование по совершенствованию комбинаторных алгоритмов и посвящена разработке алгоритма муравьиной колонии для решения многомерной задачи о ранце.

Идея муравьиного алгоритма для многомерной задачи о ранце

Пусть имеется математическая модель многомерной задачи о ранце:

$$z = \max \sum_{j=1}^{n} c_j x_j ,$$

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = \overline{1,m} ,$$

$$x_j \in \{0,1\}, \quad j = \overline{1,n} .$$

Все составляющие модель значения — неотрицательные целые числа.

Введем обозначения:

n — количество искомых величин $(j = \overline{1,n})$;

m — количество ограничений $(i = \overline{1,m})$;

r — количество агентов $(k = \overline{1,r})$, одновременно и независимо друг от друга определяющих допустимые решения;

$a_{\min}$ — параметр, пороговое значение наличия феромонов;

β — параметр, определяющий уровень испарения феромонов;

T — заранее заданное количество итераций $(t = \overline{1,T})$;

γ_j — количество допустимых решений, в которых j-ая компонента конкретизирована значением «1» на конкретной итерации;

V_j^k — множество индексов j для k-го допустимого решения таких, что x_j может быть конкретизировано значением «1» при выполнении очередной итерации;

p_j^k — вероятность выбора x_j-й компоненты для конкретизации в k-м решении;

I^k — множество индексов конкретизированных компонент в k-м решении.

После завершения итерации пересчитывается наличие феромонов для каждой k-й компоненты по формуле:

$$a_j = \frac{a_{\min} + \frac{\gamma_j}{n}(a_j - a_{\min})}{\beta}. \tag{1}$$

На каждом шаге итерации для каждого муравья определяется множество V_j^k таким образом:

$$V_j^k = \left\{ j \,\middle|\, \left(b_i^k - \sum_{j \in I^k} a_{ij} \right) \geq 0,\ i = \overline{1,m} \right\},\ k = \overline{1,r}.$$

Вероятность выбора компонента x_j подлежащей конкретизации рассчитывается следующим вероятностно-пропорциональным правилом:

$$p_j^k = \frac{a_j(c_j - c_{\min})}{\sum\limits_{j \in V_j^k} a_j(c_j - c_{\min})}, \quad j \in V_j^k, \quad k = \overline{1,r}, \tag{2}$$

где $c_{\min} = \min\limits_{j \in V_j^k} c_j - 1$.

На каждой итерации согласно правилу (2) определяется величина вероятности. По ее величине выбирается компонента, подлежащая конкретизации. Выбор имеет случайный характер. Для этого генерируется значение равномерно распределенной случайной величины из интервала (0,1). Попаданием его в соответствующий интервал пропорционально величине вероятности определяется номер конкретизируемой компоненты.

Допустимое решение для каждого муравья считается полученным, если $V_j^k = 0,\ (k = \overline{1,r})$. После получения всех r допустимых решений $X^k = (x_1^k, x_2^k, ..., x_n^k)$ вычисляется значение γ_j:

$$\gamma_j = \sum_{k=1}^{r} x_j^k, \quad j = \overline{1,n}.$$

Далее повторяется итерация t $(t \geq 2)$ с новыми значениями a_j, определенными по формуле (1). Для первой итерации принимаются все a_j одинаковыми, имеющими некоторое значение q_0 $(q_0 \in (0,1))$.

Описание алгоритма

1. Ввести значения параметров β, $a_{\min}$, числа агентов $r1$, числа итераций T.
2. $ZM = 0, \quad a_j = q_0, \quad j = \overline{1,n}, \quad t = 1.$
3. Подготовить массивы $X^k = (x_1^k, x_2^k, ..., x_n^k), \quad k = \overline{1,r}, \quad I^k, \quad r = r1.$
4. Определить множества $V_j^k, \quad k = \overline{1,r}.$

5. Если существует $V_j^{k_0} = 0,$ то перейти на п.12.

6. Определить p_j^k, $j = \overline{1,n}$, $k = \overline{1,r}$.

7. Согласно p_j^k и на основе V_j^k определяются компоненты $x_{j_0}^k$, подлежащие конкретизации значением «1», $k = \overline{1,r}$.

8. Определить $x_{j_0}^k = 1$, $k = \overline{1,r}$, и множество $I^k = I^k \cup x_{j_0}^k$, $k = \overline{1,r}$.

9. Определить значение γ_j, $j = \overline{1,n}$.

10. Определить новые значения a_j, $j = \overline{1,n}$.

11. Переход к п. 4.

12. $r = r - 1$; вычислить значение z_{k_0} полученного допустимого решения.

13. Если $r = 0$, то п. 15.

14. Переход на п. 4.

15. $r = r1$, $z_{k_0} = \max_{k=\overline{1,r}} Z_k$, $X = X^{k_0}$.

16. Если $z_{k_0} < ZM$, то п. 18.

17. $ZM = z_{k_0}$, $XM = X^{k_0}$; печать.

18. Переход к следующей итерации.

19. $t = t + 1$.

20. Если $t < T$, то п. 3.

21. Конец.

Результаты исследования

Вычислительный эксперимент по исследованию алгоритма муравьиной колонии для решения многомерной задачи о ранце проводился при следующих условиях. Решались задачи размерностей 30×70, 40×90, 50×110 и больших размерностей 250×510, 500×1010. Параметры c_j и a_{ij} генерировались как независимые случайные целые числа, равномерно распределенные в интервалах (1:10) и (1:10) соответственно. Компоненты вектора условий зависели от величин a_{ij} таким образом:

$$b_i = \frac{\sum_{j=1}^{n} a_{ij}}{3}, \quad i = \overline{1,m}.$$

Параметр α вычислялся согласно алгоритму, а параметр β принимал значение либо 0,1, либо 0,3. Для любой пары $m \times n$ было решено по 10 задач. Решение проводилось при разных количествах агентов (муравьев). Менялось количество итераций. Выделялись номера итераций, при которых было получено оптимальное или хорошее решение. Для небольших размерностей, с целью сравнения, было определено оптимальное решение методом ветвей и границ [4]. Для больших размерностей точное (оптимальное) решение не было получено, поскольку это трудоемкий вычислительный процесс и выполняется большое количество итераций до определения экстремума целевой функции.

Экспериментальное исследование показало следующее.

1. Величина параметров β предопределяет частоту появления оптимального или хорошего решения среди допустимых решений, получаемых различными муравьями. На рисунке приведен график зависимости частоты хороших решений от количества муравьев и значений β .

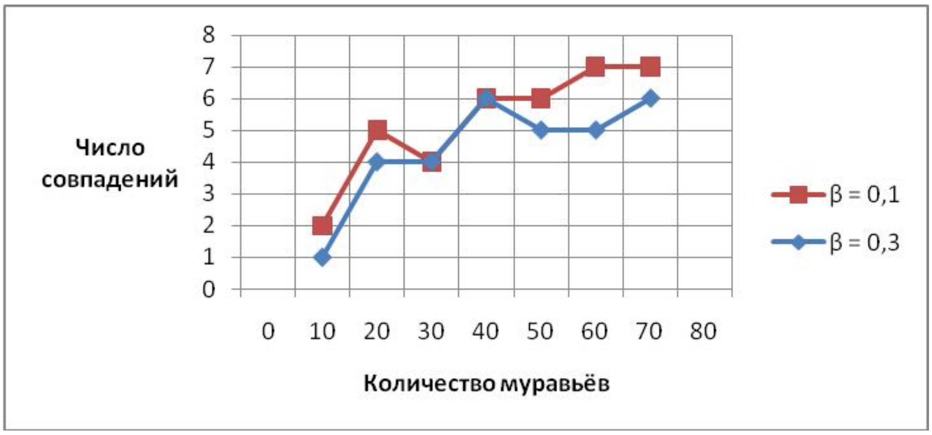

График частоты получения оптимального решения в зависимости от количества муравьев

2. Оптимальное решение для задач небольших размерностей получалось в пределах 10 итераций.

3. Приближенное решение при отклонении не больше, чем на 1 % или 2 %, получалось довольно часто. В табл. 1 приведены данные о частоте их появления.

Процентное отклонение хорошего решения от оптимального приведено ниже.

Таблица 1.

% отклонения / Число муравьев	1 %		2 %	
	β = 0,1	β = 0,3	β = 0,1	β = 0,3
10	3	6	5	9
20	7	6	9	9
30	9	9	10	10
40	9	8	10	10
50	9	9	10	10
60	10	7	10	10
70	9	8	10	10

Примечание. Частота отклонений хороших решений от оптимального в зависимости от количества муравьев и значений β .

4. Среди решенных 10 задач выделились задачи, оптимальное решение которых получалось при числе агентов 10, независимо от значения β , и сохранялось при изменении числа агентов (задача 10). Решение задачи с номером «1» не привело к экстремальному значению целевой функции, ни при каких условиях.

Остальные задачи, для которых получено оптимальное решение, стали его выдавать, начиная с 20–40 муравьев.

5. В процессе выполнения итераций наблюдается такой момент. Для некоторых задач оптимальное (наилучшее) решение повторялось многократно после первого его появления на какой-то итерации. В табл. 2 приведены эти данные.

Таблица 2.

Задачи / Муравьи	$\beta = 0{,}1$											$\beta = 0{,}3$									
	1	2	3	4	5	6	7	8	9	10		1	2	3	4	5	6		8	9	10
10	1	0	0	0	0	13	0	0	0			13	0	13	13	0	8		0	0	0
20	15	2	0	15	6	0	0	0	0			0	9	0	0	0	0		0	0	0
30	14	4	4	2	0	0	14	0	0			3	0	0	0	3	0		0	11	14
40	3	3	0	0	3	1	0	1	0			1	0	3	3	3	0		0	0	4
50	0	2	1	0	1	0	1	7	0			0	0	0	5	0	0		1	0	0
60	1	0	4	0	2	0	14	1	2			0	0	0	2	3	0		0	0	0
70	3	0	0	0	4	0	0	1	0			1	0	1	1	0	0		12	1	2

6. Элитные муравьи, многократно предопределяющие лучшие решения, практически не выделялись (термин используется в [2]).

7. Следует отметить, что чем больше муравьев участвует в работе при генерации допустимых решений, тем больше их разброс при выделении лучших значений целевой функции.

8. Для больших размерностей задач оптимальное решение муравьиным алгоритмом получить не удалось.

Заключение

Предложенный алгоритм муравьиной колонии для решения многомерной задачи о ранце обеспечивает получение оптимальных решений. Для задач небольшой размерности это случается довольно часто. Приближенное решение, как показал вычислительный эксперимент, обеспечивается почти всегда, мало зависит от числа муравьев, генерирующих допустимые решения, и от значений параметра β. Также можно утверждать, что нет необходимости проделывать большое количество итераций. Лучшие решения появляются в пределах десяти итераций. Увеличение количества муравьев не приводит к значительному улучшению получаемых решений. Выделение номера муравья, обеспечивающего лучшее решение, показало, что их разброс растет с увеличением их количества.

Интересно было бы выделить параметры, значения которых определяли такие задачи, для которых можно быстро получить хорошее решение и для которых его получить не удается. Невыясненным моментом является и то, что оптимальное или хорошее решение многократно появляется с выполнением нескольких итераций и обеспечивается различными агентами.

Самое главное для будущих исследований — дать оценку процессу решений и качеству получаемых результатов задач большой размерности.

1. Захаров А.А. Муравей, семья, колония. Москва: Наука. 2000. 132 с.

2. Штовба С.Д. Муравьиные алгоритмы: теория и применение. *Программирование*. 2005. № 4. С. 1–16.

3. Леванова Е.В., Лореш М.А. Алгоритмы муравьиной колонии и имитации отжига для решения задачи о p-медиане. *Автоматика и телемеханика*. 2004. № 3. С. 80–88.

4. Сигал И.Х., Иванова А.П. Введение в прикладное дискретное программирование. Москва: Физматлит, 2007. 303 с.

5. Михалевич В.С. Последовательные алгоритмы оптимизации и их применение. Часть I, II. *Кибернетика*. 1965. № 1, 2.

6. Дюбин Г.Н., Корбут А.А. Жадные алгоритмы для задачи о ранце: поведение в среднем. *Сибирский журнал индустриальной математики*. 1999. Т. 2. № 2(4). С. 68–93.

7. Сербин Д.А. Модификация метода ветвей и границ для решения задач целочисленного линейного программирования с булевыми переменными. Перша Міжнародна науково-практична конференція «Project, Program, Portfolio, Management». 2016. Т. 1. С. 143–146.

8. Юхименко Б.И. Ускоренный алгоритм одностороннего ветвления для решения задач линейного программирования с булевыми переменными. *Информатика и математические методы в моделировании*. 2015. Т. 5. № 4. С. 389–395.

9. Юхименко Б.И., Козина Ю.Ю. Сравнительная характеристика алгоритмов метода ветвей и границ для решения задач целочисленного линейного программирования. *Труды Одесск. политех. ун-та*. 2005. Вып. 2. С. 199–204.

10. Юхименко Б.И., Волкова Н.П. Приближенные алгоритмы решения задач о многомерном ранце. *Дослідження в математиці і механіці*. 2017. Т. 22. Вып. 2(30). С. 104–113.

Поступила в редакцию 27.05.2019