

ИССЛЕДОВАНИЕ ЗАДАЧ ПРИНЯТИЯ РЕШЕНИЙ В СИСТЕМАХ УПРАВЛЕНИЯ ОГРАНИЧЕННЫМИ РЕСУРСАМИ

Рассматриваются методы решения задач, направленных на оптимизацию в системе управления с ограниченными ресурсами. Выделяются преимущества и недостатки традиционных и эволюционных методов.

1. Введение

В последнее десятилетие во всем мире используются подходы экономико-математического моделирования для оценки эффективности управления цепями поставок и производственного менеджмента. В большинстве случаев для этого применяются модели линейного и целочисленного линейного программирования, где в качестве критериев принимают либо прибыль, полученную предприятием при реализации той или иной производственной программы, либо издержки, связанные с производством конечной продукции. Основным недостатком этого подхода является жесткое закрепление производственных ресурсов при выполнении всего комплекса работ на всем временном интервале планирования. В данной работе рассмотрены методы, позволяющие преодолеть это ограничение, разрешив перераспределять производственные мощности в процессе реализации производственной программы. Предложены получаемые при этом модели оптимального управления ресурсами предприятия и методы их анализа.

Актуальность исследования. Создать такую организацию, которая обеспечит выпуск высококачественной продукции (или услуги) в нужном объеме и ассортименте, без проведения исследований невозможно. Особую роль здесь играют исследования систем управления. Эта проблема была актуальной всегда, однако до недавнего времени она в большей степени решалась в рамках математических дисциплин, таких как теория вероятностей, математическая статистика, логика, теория множеств и др.

Целью данного исследования является изучение эффективных методов принятия решений, их сравнение, определение лучшего оптимизационного метода для решения задач управления.

2. Традиционные методы решения

В начале, для сравнения, приведем описание линейных алгоритмов. Одним из них, вероятно, самым популярным, является симплекс-метод.

Симплекс-метод – алгоритм решения оптимизационной задачи линейного программирования путём перебора вершин выпуклого многогранника в многомерном пространстве. Данный метод, имеющий несколько различных форм (модификаций), был разработан в 1947 году Г. Данцигом.

Задача линейного программирования состоит в том, что необходимо максимизировать или минимизировать некоторый линейный функционал на многомерном пространстве при заданных линейных ограничениях.

Следует указать, что каждое из линейных неравенств на переменные ограничивает полупространство в соответствующем линейном пространстве. В результате все неравенства ограничивают некоторый многогранник (возможно, бесконечный), называемый также полиэдральным конусом. Уравнение $W(x) = c$, где $W(x)$ — максимизируемый (или минимизируемый) линейный функционал порождает гиперплоскость $L(c)$. Зависимость от c порождает семейство параллельных гиперплоскостей. Тогда экстремальная задача приобретает следующую формулировку требуется найти такое наибольшее c , чтобы гиперплоскость $L(c)$ пересекала многогранник хотя бы в одной точке. Заметим, что пересечение оптимальной гиперплоскости и многогранника будет содержать хотя бы одну вершину, причём их будет более одной, если пересечение содержит ребро или k -мерную грань. Поэтому максимум функционала можно искать в вершинах многогранника.

Принцип симплекс-метода состоит в том, что выбирается одна из вершин многогранника, после чего начинается движение по его рёбрам от вершины к вершине в сторону увеличения значения функционала. Когда переход по ребру из текущей вершины в другую вершину с более высоким значением функционала невозможен, считается, что оптимальное значение с найдено.

Для решения системы все неизвестные произвольно подразделяют на базисные и свободные. Число базисных переменных определяется числом линейно-независимых уравнений. Остальные неизвестные свободные. Им придают произвольные значения и подставляют в систему. Любому набору свободных неизвестных можно придать бесчисленное множество произвольных значений, которые дадут бесчисленное множество решений. Если все свободные неизвестные приравнять к нулю, то решение будет состоять из значений базисных неизвестных. Такое решение называется базисным.

В теории линейного программирования существует теорема, которая утверждает, что среди базисных решений системы можно найти оптимальное, а в некоторых случаях и несколько оптимальных решений, но все они обеспечат экстремум целевой функции. Таким образом, если найти какой-либо базисный план, а затем улучшить его, то получится оптимальное решение. На этом принципе и построен симплекс-метод.

Последовательность вычислений симплекс-методом можно разделить на две основные фазы:

- нахождение исходной вершины множества допустимых решений (нахождение базисного решения),
- последовательный переход от одной вершины к другой, ведущий к оптимизации значения целевой функции (последовательное улучшение найденного на первом этапе базисного решения).

При этом в некоторых случаях исходное решение очевидно или его определение не требует сложных вычислений, например, когда все ограничения представлены неравенствами вида «меньше или равно» (тогда нулевой вектор совершенно точно является допустимым решением, хотя и, скорее всего, далеко не самым оптимальным). В таких задачах первую фазу симплекс-метода можно вообще не проводить. Симплекс-метод, соответственно, делится на однофазный и двухфазный.

3. Генетические алгоритмы

Предложенные сравнительно недавно — в 1975 году — Джоном Холландом генетические алгоритмы (ГА) основаны на принципах естественного отбора Ч. Дарвина. ГА относятся к стохастическим методам. Эти алгоритмы успешно применяются в различных областях деятельности (экономика, физика, технические науки и т.п.). Генетические алгоритмы относят к области мягких вычислений. Это понятие объединяет такие области, как нечеткая логика, нейронные сети, вероятностные рассуждения, сети доверия и эволюционные алгоритмы, которые дополняют друг друга и используются в различных комбинациях или самостоятельно для создания гибридных интеллектуальных систем. Генетические алгоритмы — это адаптивные методы поиска, которые в последнее время используются для решения задач оптимизации. В них применяются как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде и основные понятия линейной алгебры.

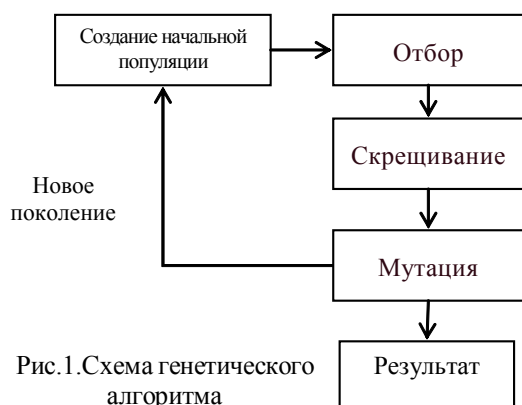


Рис.1. Схема генетического алгоритма

терминология в упрощенном виде и основные понятия линейной алгебры.

Основной идеей генетических алгоритмов (рис.1) является организация «борьбы за существование» и «естественного отбора» среди пробных решений, которые выбираются случайным образом на требуемом отрезке. Поскольку генетические алгоритмы используют биологические аналогии, то и применяющаяся терминология напоминает биологическую. Так, одно пробное решение называется особью или хромосомой, а набор всех пробных решений — популяцией. Каждая особь

оценивается с использованием «функции приспособленности», в результате чего с каждым генотипом ассоциируется определенное значение («приспособленность»), которое определяет, насколько хорошо фенотип, который им описывается, решает поставленную задачу.

Далее наступает процесс «размножения». Этот процесс в разных алгоритмах реализован по-разному, все зависит от представления данных. Главное требование к размножению — чтобы потомок или потомки имели возможность унаследовать черты обоих родителей, «смешав» их каким-либо способом.

Также возможен этап «мутации». Он подразумевает изменение особи по определенной формуле с целью улучшить ее приспособленность аналогично случайным мутациям в природе. В случае ухудшения результата особь остается неизменной. Это позволяет привнести в результат сторонние значения, которые вероятно хорошо скажутся на следующей популяции.

На этапе отбора должны быть выделены из популяции наиболее приспособленные особи. Обычно их количество задается параметром в начале расчетов. Все особи, не прошедшие отбор, считаются умершими и далее не учитываются. А оставшиеся становятся родителями для следующей итерации алгоритма.

Генетический алгоритм – это простая модель эволюции в природе, реализованная в виде компьютерной программы. Результатом его работы является не оптимальный результат, а просто лучший для данного количества итераций, но данный результат часто является удовлетворительным для сложных систем.

К преимуществам генетических алгоритмов можно отнести простоту их реализации, параллельный поиск решения, что позволяет решить проблему локальных оптимумов, и довольно высокую скорость работы. Но возможен исход, когда все представители популяции идентичны, что останавливает дальнейшую эволюцию и начинается вырождение популяции. Также требуется определить систему кодирования особей и способы их смешивания, что бывает проблематично в зависимости от поставленной задачи.

4. Муравьиные алгоритмы

Имитация самоорганизации муравьиной колонии составляет основу муравьиных алгоритмов оптимизации (рис.2). 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 сегодняшний день уже получены хорошие результаты для оптимизации таких сложных комбинаторных задач, как задача коммивояжера, задача оптимизации маршрутов грузовиков, задача раскраски графа, квадратичная задача о назначениях, задача оптимизации сетевых графиков, задача календарного планирования и многие другие. Особенно эффективны муравьиные алгоритмы при динамической оптимизации процессов в распределенных нестационарных системах, например, трафиков в телекоммуникационных сетях.

Идея муравьиного алгоритма – моделирование поведения муравьев, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи и адаптироваться к изменяющимся условиям, находя новый кратчайший путь. При своём движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути. Это элементарное правило поведения и определяет способность муравьев находить новый путь, если старый оказывается недоступным. Например, на оптимальном доселе пути возникает преграда. В этом случае необходимо определение нового оптимального пути. Дойдя до преграды, муравьи с равной вероятностью будут обходить её справа и слева. То же самое будет происходить и на обратной стороне преграды. Однако те муравьи, которые случайно выберут кратчайший путь, будут быстрее его проходить, и за несколько передвижений он будет более обогащён феромоном. Поскольку движение муравьев определяется концентрацией феромона, то следующие будут предпочитать именно этот путь, продолжая обогащать его феромоном до тех пор, пока этот путь по какой-либо причине станет недоступен.

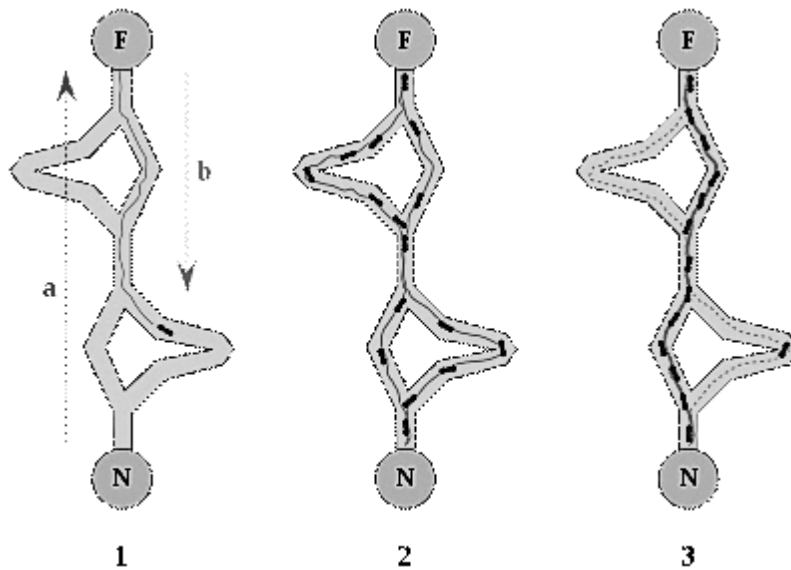


Рис. 2. Схема муравьиного алгоритма

Как было сказано выше, муравьиный алгоритм моделирует многоагентную систему. Ее агентов в дальнейшем будем называть муравьями. Как и настоящие муравьи, они довольно просто устроены: для выполнения своих обязанностей они требуют небольшое количество памяти, а на каждом шаге работы выполняют несложные вычисления.

Каждый муравей хранит в памяти список пройденных им узлов. Этот список называют списком запретов (tabu list) или просто памятью муравья. Выбирая узел для следующего шага, муравей «помнит» об уже пройденных узлах и не рассматривает их в качестве возможных для перехода. На каждом шаге список запретов пополняется новым узлом, а перед новой итерацией алгоритма – т.е. перед тем, как муравей вновь проходит путь – он опустошается.

Кроме списка запретов, при выборе узла для перехода муравей руководствуется «привлекательностью» ребер, которые он может пройти. Она зависит, во-первых, от расстояния между узлами (т.е. от веса ребра), а во-вторых – от следов феромонов, оставленных на ребре прошедшими по нему ранее муравьями. Естественно, что в отличие от весов ребер, которые являются константными, следы феромонов обновляются на каждой итерации алгоритма: как и в природе, со временем следы испаряются, а проходящие муравьи, напротив, усиливают их.

Пусть муравей находится в узле i , а узел j – это один из узлов, доступных для перехода: $j \in S_i$. Обозначим вес ребра, соединяющего узлы i и j , как w_{ij} , а интенсивность феромона на нем – как t_{ij} . Тогда вероятность перехода муравья из i в j будет равна:

$$p_{ij} = \frac{t_{ij}^\alpha + \frac{1}{w_{ij}^\beta}}{\sum_{l \in S_i} \left(t_{il}^\alpha + \frac{1}{w_{il}^\beta} \right)} \quad (1)$$

где α и β – это регулируемые параметры, определяющие важность составляющих (веса ребра и уровня феромонов) при выборе пути. Очевидно, что при $\alpha=0$ алгоритм превращается в классический жадный алгоритм, а при $\beta=0$ он быстро сойдется к некоторому субоптимальному решению. Выбор правильного соотношения параметров является предметом исследований, и в общем случае производится на основании опыта.

После того, как муравей успешно проходит маршрут, он оставляет на всех пройденных ребрах след, обратно пропорциональный длине пройденного пути:

$$\Delta_{ij} = \begin{cases} \frac{k}{L}, (ij) \in P \\ 0, (ij) \notin P \end{cases} \quad (2)$$

где L – длина пути, а k – регулируемый параметр. Кроме этого, следы феромона испаряются, т.е. интенсивность феромона на всех ребрах уменьшается на каждой итерации алгоритма. Таким образом, в конце каждой итерации необходимо обновить значения интенсивностей:

$$t_{ij} = (1 - e) \cdot t_{ij} + \Delta t_{ij} \quad (3)$$

Результаты первых экспериментов с применением муравьиного алгоритма для решения задачи коммивояжера были многообещающими, но далеко не лучшими по сравнению с уже существовавшими методами. Однако простота классического муравьиного алгоритма (названного «муравьиной системой») оставляла возможности для доработок – и именно алгоритмические усовершенствования стали предметом дальнейших исследований Марко Дориго и других специалистов в области комбинаторной оптимизации. В основном, эти усовершенствования связаны с большим использованием истории поиска и более тщательным исследованием областей вокруг уже найденных удачных решений.

Эффективность муравьиных алгоритмов сравнима с эффективностью общих метаэвристических методов, а в ряде случаев – и с проблемно-ориентированными методами. Наилучшие результаты муравьиные алгоритмы показывают для задач с большими размерностями областей поиска. Муравьиные алгоритмы хорошо подходят для применения вместе с процедурами локального поиска, позволяя быстро находить начальные точки для них.

Наиболее перспективными направлениями дальнейших исследований следует считать анализ способа выбора настраиваемых параметров алгоритмов. В последние годы предлагаются различные способы адаптации параметров алгоритмов «на лету». Поскольку от выбора параметров сильно зависит поведение муравьиных алгоритмов, именно к этой проблеме обращено наибольшее внимание исследователей на данный момент.

Выводы

Изучены эффективные методы принятия решений. Важно понять, что выбор того либо иного метода зависит от типа решаемой задачи.

В противовес традиционному симплекс-методу генетический и муравьиный алгоритмы представляют собой сложную самоорганизующуюся систему и перспективны в применении с использованием современных информационных технологий.

Научная новизна: результатом проведенного исследования является решение многокритериальной задачи с применением эволюционных алгоритмов.

Практическая значимость: предложенная информационная подсистема позволяет управлять предприятием с ограниченными ресурсами с применением современных информационных технологий.

Список литературы: 1. Гвоздинский А.Н., Якимова Н.А., Губин В.А. Методы оптимизации в системах принятия решений. Харьков, ХНУРЭ. 2006. 325 с. 2. Бондаренко М.Ф., Гвоздинский А.Н. Оптимизационные задачи в системах принятия решений. Харьков, ХТУРЭ. 1998. 3. Гвоздинский А.Н., Клименко Е.Г. Применение генетических алгоритмов для решения оптимизационных задач. Харьков, ХНУРЭ. 2001. С.390-391. 4. Гвоздинский А.Н., Малышкин В.А. Применение методов эволюционной оптимизации для решения задач производственного планирования. Харьков, ХНУРЭ. 2011. С.97-102. 5. Мазур И., Шапиро В. Управление проектами. М.: ОМЕГА. 2007. 6. Будков В.М. Математические основы управления проектами. М.: Наука. 2005.

Поступила в редколлегию 11.12.2012

Гвоздинский Анатолий Николаевич, канд. тех. наук, профессор кафедры искусственного интеллекта ХНУРЭ. Научные интересы: оптимизация процедур принятия решений в сложных системах управления. Адрес: Украина, 61115, Харьков, ул. Академика Ляпунова, 7, кв. 9, тел. 702-38-23.

Янов Дмитрий Михайлович, студент, бакалавр специальности интеллектуальные системы принятия решений, факультет КН ХНУРЭ. Адрес: Украина, 61172, ул. Луи Пастера, 322, кв. 109, тел. 716-36-99, email: dmyto.yanov@gmail.com.