УДК 62-519

ДЕЦЕНТРАЛИЗОВАННОЕ УПРАВЛЕНИЕ АДАПТИВНЫМИ СЕТЯМИ ПОСТАВОК НА ОСНОВЕ ТЕОРИИ КОЛЛЕКТИВНОГО ИНТЕЛЛЕКТА И АГЕНТНОЙ ТЕХНОЛОГИИ
Часть 1: Модель сети поставок

Л. Б. Шереметов,
канд. техн. наук, старший научный сотрудник
Санкт-Петербургский институт информатики и автоматизации Российской академии наук

Приводится один из подходов к децентрализованному управлению сетями поставок, основанный на концепциях коллективного интеллекта и многоагентных системах. Построена модель сети поставок. Для оптимального управления потоками в условиях неопределенной среды на локальном уровне использованы алгоритмы стимулируемого обучения, а для оптимизации глобального поведения сети предложен алгоритм двойной Q-маршрутизации.

Ключевые слова *— сети поставок, многоагентная система, теория коллективного интеллекта, стимулируемое обучение.*

Введение

В данной статье рассматривается эмерджентный тип производственных систем, известных как адаптивные сети поставок (АСП), т. е. сети с изменяющейся топологией, которая может эволюционировать во времени (*adaptive supply networks*)[1]. АСП управляются уже не централизованно компаниями-лидерами, а через виртуальные пространства принятия решений (ПР), где оптимизация решений все больше и больше доминирует как критический фактор, обеспечивающий конкурентоспособность предприятий [1]. При этом ПР по конфигурированию и управлению АСП осуществляется в динамической окружающей среде, где децентрализованная система ПР становится необходимостью [2, 3]. Это диктует необходимость разрабатывать новые подходы к решению задач оптимизации, лежащих в их основе. С одной стороны, традиционные проблемы оптимизации требуют новых алгоритмов, моделирующих возможности членов сети на макро- и микроуровнях, и обмена информации между уровнями. С другой стороны, наличие информации о состоянии сети в реальном времени создает необходимость и дает возможность повторной оптимизации, которая способна быстро скорректировать план в ответ на изменения в исходных данных.

Децентрализация управления в АСП имеет особую характеристику, которая заключается в том, что все локальные системы управления принимают решения в условиях неопределенности, поскольку любая доступная информация о других партнерах АСП может быть неполной, нечеткой и недостоверной. При этом каждый партнер должен работать независимо от другого, анализируя свое собственное состояние, состояние внешнего мира и принимая решения. Иными словами, он должен быть автономным и активным, добиваясь локальных целей таким образом, чтобы глобальная цель АСП была достигнута более эффективно. Такие автономные системы обычно рассматриваются как многоагентные системы (МАС) [4]. Использование технологии агентов для оптимизации процессов обусловлено двумя решающими факторами: возрастающей изменчивостью среды и децентрализованным ПР [5]. Данная работа преследует цель исследовать то, как интеграция агентов со способностями к коллективному обучению способствует распределенному и децентрализованному ПР. Проблема оптимизации поведения АСП рассматривается

[1] Близкими по смыслу терминами являются открытые сети поставок и сети поставок «по требованию» (*open and on-demand supply networks*).

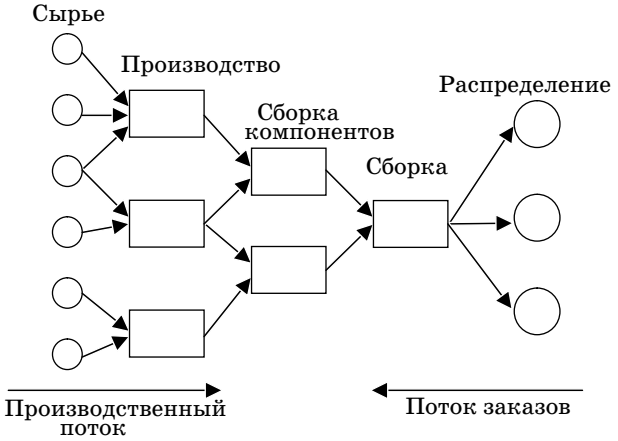
в контексте теории коллективного интеллекта (КОИН) [6], которая является расширением модели динамического программирования и алгоритмов стимулируемого обучения (например, Q-обучения и Q-маршрутизации) [7, 8].

Предлагаемая статья состоит из двух частей. В первой части разработана модель сети поставок, основанная на интеграции алгоритмов коллективного интеллекта в систему агентного моделирования. На локальном уровне использованы алгоритмы стимулируемого обучения, а для оптимизации глобального поведения АСП предложен алгоритм двойной Q-маршрутизации. Обмен информацией и коллективное обучение, встроенные в алгоритм оптимизации, являются отличительными чертами данного подхода. Во второй части работы будут рассмотрены вопросы реализации предложенной модели в многоагентной среде моделирования и примеры ее использования для управления поведением и конфигурацией АСП, основанного на анализе производственной мощности.

Состояние исследований в области управления АСП

Эволюция концепции производственных цепочек к адаптивным сетям поставок оказывает существенное влияние на постановку и методы решения задач по управлению как отдельными производствами, так и сетью поставок в целом. Термин «цепь поставок» (*supply chain*) используется с 1980-х гг. для описания полного спектра операций — от заказа сырья, производства и сборки продукции до оптовых и розничных поставок продукции конечным потребителям. Цепь поставок, таким образом, хорошо изображается как сеть поставщиков, производителей и заказчиков (рис. 1).

Традиционно управленческие решения в сетях поставок принято классифицировать на три категории [9]: стратегические, операционные и управляющие. Стратегические решения, такие как, например, выбор участника сети поставок, носят долгосрочный характер. Операционные решения относятся к производственным решениям для выполнения заказов. Наконец, управляющие касаются решений по заказам, находящимся в производстве. Современный этап развития цепей поставок характеризуется преобразованием традиционных линейных структур в сложные, гибкие и открытые сети поставок, в которых связи между их участниками уже не носят столь выраженной линейной структуры. АСП отличаются возрастанием сложности ПР, что приводит к изменению характера решаемых задач. Свойства открытости и гибкости сетей видоизменяют временной масштаб ПР, когда приходится переоптимизировать решения, например о переконфигурировании сети, каждый раз, когда принят в рассмотрение новый заказ [10]. Это обеспечивает возможность разумно объединить новый заказ со всеми другими внеплановыми заказами. Новый заказ может иметь ассоциируемый с ним предельный срок, и, следовательно, вся операция должна осуществляться с мягкими ограничениями в реальном времени. Этот тип операций в сети поставок называется операциями в реальном времени.



■ *Рис. 1. Общая схема цепочки поставок*

Существующие многочисленные алгоритмы и методы локальной оптимизации (например, системы календарного планирования, управления запасами, заказами, закупками, поставками и т. д.) обычно затрачивают много времени для нахождения наиболее подходящего решения, которое, при этом, далеко не всегда обеспечивает оптимизацию общего бизнес-процесса на глобальном уровне из-за конфликтов между локальными целями различных партнеров АСП [11, 12]. Кроме того, большинство этих подходов отражают традиционную модель сети поставок, являющуюся в значительной мере статической, опирающейся на долгосрочные отношения между партнерами.

Использование динамических схем конфигурации требует достаточно гибкого подхода, обеспечивающего получение компромиссных решений между локальными целями партнеров. Опубликован ряд исследовательских работ по использованию техник мягких вычислений, машинного обучения и агентов в задачах моделирования динамической сети поставок и их оптимизации в условиях неопределенности [13, 14]. Общее представление о многоагентном моделировании и управлении АСП можно найти в работах [15, 16]. Тем не менее, в агентных моделях не рассматривался вопрос о том, как строить модели принятия сложных решений в условиях неопре-

деленной среды. К тому же отсутствуют разработки механизмов обучения для систем, включающих большое число агентов. Поэтому в данной исследовательской работе основное внимание уделяется подходу к динамической децентрализованной оптимизации поведения элементов АСП, разработанному в рамках парадигмы многоагентных самоорганизующихся систем в условиях неопределенной среды.

Многоагентная модель адаптивной сети поставок

Предложенная модель децентрализованного управления основана на динамическом взаимодействии каждого партнера АСП со средой, а именно на моделировании по принципам теории КОИН, отражающем предположение об ограниченном знании среды.

Основы теории коллективного интеллекта.

Традиционный подход к оптимизации больших распределенных систем заключается в явном моделировании динамики системы в целом, что часто приводит к весьма неустойчивым решениям и техникам оптимизации с ограниченной применимостью. В качестве альтернативы, разработанной в рамках теории КОИН, предлагается использовать агентов, выполняющих алгоритмы стимулируемого обучения (СО), берущие основу в динамическом программировании (ДП) [17, 18]. Принцип ДП состоит в замене глобальной оптимизации на последовательную, т. е. оптимизацию каждого этапа решения (или промежутка времени) по очереди, но с учетом при этом как принятых ранее, так и оставшихся решений.

В контексте АСП следует искать оптимальность локальных решений в условиях ограничений, наложенных оптимальным поведением АСП в целом, подразумевая отсутствие первоначальной математической модели. Наиболее важными характеристиками, отличающими этот тип моделей обучения от других, являются: а) обучение путем проб и ошибок и б) наличие вознаграждения/наказания (получаемых с определенной задержкой), которые оказывают влияние на будущее поведение агента.

Одним из наиболее важных вкладов в развитие СО стала разработка алгоритма Q-обучения [8], который следует автономной (*off-line*) стратегии [17]. В этом случае функция прибыли-действия Q, полученная в результате обучения, аппроксимирует функцию оптимальной прибыли-действия Q^* независимо от последующей стратегии. В состоянии $x(t)$, если Q-значения точным образом представляют модель среды, лучше всего выполнить такое действие, которое имеет наибольшее/наименьшее Q-значение (согласно рассматриваемому случаю) среди всех возможных действий $a_{i,k} \in A_i$. Модифицированные Q-значения вычисляются по правилу обновления, использующему награду $r(t + 1)$, рассчитанную средой в результате выполнения действия $a_{x(t)}$ в состоянии $x(t)$. Правило обновления Q-обучения определяется формулой

$$Q_{\left(x(t),a_{x(t)}\right)}(t+1)=Q_{\left(x(t),a_{x(t)}\right)}(t)+\alpha\times$$

$$\times\left[r(t+1)+\gamma\min_{a_{x(t+1)}}Q_{\left(x(t+1),a_{x(t+1)}\right)}(t+1)-Q_{\left(x(t),a_{x(t)}\right)}(t)\right],$$

где $Q_{\left(x(t),\,a_{x(t)}\right)}$ — приблизительная оценка среды, способ обновления которой можно считать одной из наиболее важных задач, требующих решения при моделировании реальных систем; α — скорость обучения; $r(t + 1)$ — усиление произведенного действия; γ — скорость сокращения (reduction rate). Результат этой функции может представлять любую цену, связанную с выполнением определенного действия. При нашем подходе оно символизирует частичное время производства продукта, состоящее из времени перехода, времени ожидания и времени операции.

Метод Q-обучения позволяет решать задачи обучения только для одного агента. Тем не менее, когда несколько агентов работают в общей среде, этот метод не является достаточно эффективным, так как обуславливает эгоистичное поведение агентов (так называемая стратегия ε_*greedy*). Рассмотрим концептуальную модель АСП, основанную на алгоритме коллективного обучения.

Концептуальная модель АСП.

В рамках предложенного подхода АСП рассматривается как МАС, работающая в режиме реального времени и обучающаяся путем наблюдения своего взаимодействия с реальной средой, где:

- среда имеет динамическую и неопределенную природу; ее первоначальная модель поведения неизвестна;
- каждый партнер АСП представляется как агент, имеющий автономное поведение и характеризующийся локальной функцией полезности, т. е. каждый из них имеет индивидуальное восприятие среды;
- управление и схемы взаимодействия между агентами являются децентрализованными;
- обмен сообщениями и продуктами между агентами имитирует информационные и материальные потоки;
- агенты приспосабливают свое локальное поведение к изменяющейся среде, выполняя алгоритмы стимулируемого обучения;

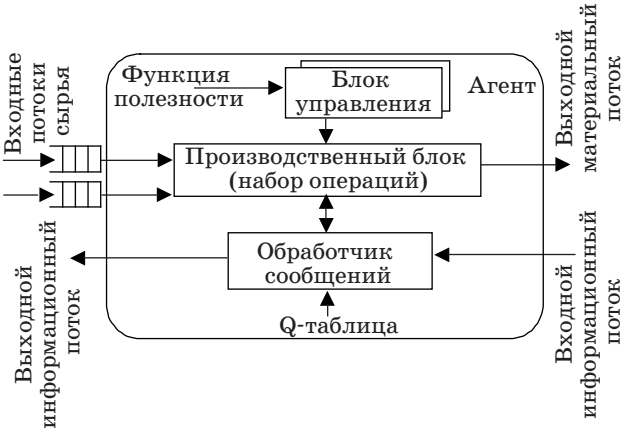
• агенты используют технику коллективного обучения для оптимизации глобального поведения, применяя алгоритм двойной Q-маршрутизации.

Оптимальное действие с точки зрения одного партнера АСП не обязательно оказывается оптимальным действием для всей АСП. Это обусловлено недостатком знания о том, что происходит у других партнеров на разных уровнях (например, на нижнем или верхнем участках сети, *downstream and upstream supply chains*). Опишем алгоритмы, используемые для согласования локальных решений, принятых каждым партнером, с глобальной целью, состоящей в оптимизации поведения всей АСП. Другими словами, ищется ответ на вопрос, каким является лучшее действие, совершенное на локальном уровне каждого партнера, позволяющее оптимизировать всю АСП.

У партнеров АСП есть общие функции, которые определяют входящие и выходящие потоки, их преобразование и управление [19]. Материалы в АСП представляются как объекты, формирующие часть среды. Поэтому каждый агент может их изменять или воздействовать на них. Детали объектов хранятся как свойства. Каждый агент имеет функцию локальной полезности и управляет Q-таблицей, которая содержит воспринимаемую информацию о среде (как об объектах среды, так и о соседних агентах). Общую схему агента (рис. 2) можно применить как к партнерам АСП, так и к компонентам каждого партнера в зависимости от необходимого уровня абстракции. Сеть агентов представляет АСП.

Согласно рис. 1 и 2, в модель АСП входят следующие элементы:

• множество l агентов складов готовой продукции (*distribution warehouse agents*, DWA) — «голова» сети: $W = \{W_1, W_2, ..., W_l\}$;



■ ***Рис. 2.*** *Общая схема агента в среде моделирования сети поставок*

• множество m агентов-производителей (PA) — производители, сборщики и поставщики компонентов — промежуточные партнеры сети, обладающие как входными, так и выходными потоками: $M = \{M_1, M_2, ..., M_m\}$;

• множество k агентов-поставщиков (SA) — «хвост» сети: $S = \{S_1, S_2, ..., S_k\}$;

• множество q операций, выполняемых агентами: $O = \{O_1, O_2, ..., O_q\}$;

• вектор неотрицательных значений r признаков для каждой операции $O_i : \mathbf{V}_i = < v_1^i, \ldots, v_r^i >$, где, например, v_1^i — усредненное время операции;

• множество s объектов, соответствующих типу сырья: $MP = \{MP_1, MP_2, ..., MP_s\}$;

• множество n объектов, соответствующих типу конечного продукта: $P = \{P_1, P_2, ..., P_n\}$;

• вектор неотрицательных значений r признаков для каждого продукта $P_j : \mathbf{PV}_j = < pv_1^j, ..., pv_r^j >$, где, например, pv_1^j — приоритет продукта.

В предложенной модели каждый агент обладает следующими признаками.

• Множество состояний среды: $X = \{x_1, x_2, x_3, ...\}$. Знание (обычно неполное) о других агентах считается частью состояния среды. Например, в некоторых случаях агент-производитель может принимать решения без знания о том, что поставщик часто не укладывается в срок поставки.

• Способность агента к действию представляется как множество $A = \{a_1, a_2, a_3, ...\}$. Отметим, что для PA это множество является подмножеством множества операций O.

• Отношения между агентами в АСП определяются как $R = \{r_1, r_2, r_3, ...\}$. Агенты, известные данному агенту, образуют перечень его соседей $N = \{n_1, n_2, n_3, ...\}$. В случае линейной модели в этот перечень включены только агенты от ближайшего эшелона сети. Для каждого соседнего агента учитываются следующие параметры: а) его отношение к данному агенту (заказчику, поставщику); б) природа соглашения, которое обуславливает взаимодействие (гарантии производства) и в) права доступа к информации между агентами (локальное состояние агентов, которое нужно учитывать в процессе ПР).

• Приоритеты каждого агента представляются как $Q = \{q_1, q_2, q_3, ...\}$. Они могут использоваться при установлении последовательности обработки входящих сообщений.

• Функция локальной полезности (LUF) представляется в виде уравнения Q-обучения.

• Набор элементов управления $C = \{c_1, c_2, c_3, ...\}$. Элемент управления вызывается, когда есть решение, которое необходимо принять во время обработки сообщения. Например, для того чтобы определить следующий пункт назначения при транспортировании материалов, используется алгоритм управления маршрутизацией.

• Каждый агент имеет обработчик сообщений, который несет ответственность за отсылку и доставку различных сообщений с целью обеспечить связь между агентами.

Чтобы решить задачу оптимизации АСП, предложен алгоритм коллективного обучения, называемый алгоритмом двойной Q-маршрутизации.

Алгоритм двойной Q-маршрутизации.

Алгоритм коллективного обучения строится на основе алгоритма Q-маршрутизации, используемого для маршрутизации пакетов в коммуникационных сетях [7, 20], и алгоритмов оптимизации по принципу «муравьиной колонии» (это название было придумано изобретателем алгоритма Марко Дориго (Marco Dorigo)) [21]. Введение дополнительных обратных связей обусловило его название — алгоритм двойной Q-маршрутизации. Обучение производится на двух уровнях: вначале локально, на уровне агента, с использованием правила стимулируемого обучения и затем глобально, на уровне системы, путем настройки функции полезности. Управляющие сообщения позволяют пересматривать состояния партнеров АСП, модифицируя Q-значения, которые аппроксимируются аппроксиматором функций, в качестве которого могут быть использованы таблицы поиска, нейронные сети и т. д.

В алгоритме двойной Q-маршрутизации существует 5 типов управляющих сообщений.

1. *'Сообщение среды'* (*environment-message*), генерируемое промежуточным агентом-производителем после приема сырья, если интервал времени ω уже прошел.

2. *'Сообщение «муравей»'* (*ant-message*), генерируемое DWA в соответствии с интервалом времени *w_ants*, когда оконченный продукт поступает на склад.

3. *'Сообщение обновления'* (*update-message*), генерируемое в фазе планирования каждые ε_*update* секунд в целях запроса у соседних агентов-производителей их оценок об операциях над продуктом.

4. *'Сообщение обратного обновления'* (*update-back-message*), генерируемое после получения сообщения обновления с целью ускорить знакомство со средой.

5. *'Сообщение наказания'* (*punishment-message*), применяемое для того, чтобы наказать РА, использующего перегруженный ресурс.

Эти сообщения применяются в рамках алгоритма двойной Q-маршрутизации, состоящего из алгоритмов планирования, «муравья» и наказания, каждый из которых выполняет особую функцию, позволяющую осуществлять децентрализованную оптимизацию. Рассмотрим каждый алгоритм более подробно (алгоритмы в псевдокоде приведены в приложении).

Алгоритм планирования исследует лучшие возможности (с точки зрения оценок *Q*-значений), вытекающие из непредвиденных изменений в среде. Исследование среды может вызвать существенную потерю времени! В алгоритме двойной Q-маршрутизации механизм планирования (в рамках значения этого термина в СО) был разработан на локальном уровне каждого агента. Он состоит в посылке сообщения обновления каждые ε_*update* секунд. Это сообщение запрашивает оценки *Q*-значений всех продуктов, которые на данный момент известны соседям.

В алгоритме «муравья» генерируются сообщения-«муравьи», которые используются в качестве обратной связи системы: каждое сообщение переносит статистические данные, полученные на своем пути (также в терминах *Q*-значений), и позволяет осуществить процесс передачи информации о среде между агентами. Когда сообщение среды поступает на DWA, «муравей» посылается в ответ в том случае, если период времени ω_*ants* уже прошел. Поступив на склад сырья, сообщение-«муравей» умирает. «Муравей» измеряет *Q*-значение каждого РА, через которого прошло сырье до прибытия на DWA.

Наконец, алгоритм наказания пытается идентифицировать и разрешить конфликты (между лучшими оценками) среди соседних агентов. В некоторых случаях различные агенты из одного эшелона могут иметь одинаковую наилучшую оценку соседей (предпочитая общего партнера или одинаковый маршрут). Если они действуют «эгоистичным» образом, партнер может оказаться перегружен и в очереди к нему происходит скученность. С целью предотвратить скученность агент должен принести в жертву свою личную полезность и использовать другой маршрут. Чтобы рассмотреть эту проблему, разработан алгоритм наказания, заставляющий агента, который получает сообщение наказания, рассчитать вторую наилучшую оценку.

Алгоритм двойной Q-маршрутизации интегрирует описанные выше алгоритмы для получения обратной связи от среды и сопоставления локальных поощрений с глобальной целью. Первый шаг алгоритма заключается в фиксировании исходных *Q*-значений и параметров СО, таких как α, γ и ε_*update*. Затем каждый агент, выполняющий алгоритм двойной Q-маршрутизации, читает заголовок сообщения с информацией о среде, полученного от другого агента, и посылает сообщения обратной связи. Следующим шагом является выполнение оптимального действия в соответствии с усвоенной в результате обучения политикой. Наконец, агент получает поощрение или наказание от среды (агент может быть частью среды для другого агента). Алгорит-

мы планирования, наказания и «муравья» выполняются одновременно и помогают сократить время обнаружения локальной оптимальной политики для каждого агента, выполняющего этот алгоритм.

Заключение

Оптимизация общих бизнес-процессов современных предприятий является актуальной проблемой. Ограничения традиционных подходов к решению задачи динамической глобальной оптимизации АСП заключаются в невозможности работать с неполной информацией, в условиях сложных динамических взаимодействий между элементами либо в необходимости централизации управления и информации. При этом большая часть эвристических методов не гарантирует глобальную оптимизацию системы.

Задача динамической оптимизации АСП нами решается в рамках теории КОИН. Построена модель АСП и разработаны алгоритмы коллективного обучения (двойной Q-маршрутизации). Предлагаемый подход основывается на динамическом построении модели среды каждым агентом во время моделирования (с использованием алгоритмов коллективного обучения). В результате он получает возможность принимать локальные решения по отношению к изменяющейся конфигурации сети поставок и к любым динамическим изменениям в производственной программе, применяя модели управления в реальном времени и модели повторной оптимизации.

Приложение

Алгоритм двойной Q-маршрутизации и его компоненты

При обозначении агентов-производителей введен дополнительный индекс, определяющий эшелон АСП, к которому относится соответствующий агент. При этом, без потери общности алгоритмы разработаны для трехэшелонной АСП, где эшелоны обозначены индексами *x, y, z*.

Алгоритм 1: Алгоритм планирования

В каждый момент ε_*update*
  послать update-message всем соседям с запросом их оценок всех известных продуктов
если (*update-message получено*)
тогда *отправить update-back-message агенту-источнику update-message с оценками* $Q^{\mu}_{(x(t),\,a_{x(t)})}(t)$ *всех известных продуктов, появившихся в момент t*
если (*update-back-message получено*)
тогда *обновить Q-значение таким же способом, как и в случае environment-message*

Алгоритм 2: Алгоритм «муравья»

Если (*продукт прибывает на DWA*)
тогда *DWA генерирует сообщение ant-message, если время w_ant уже прошло*
если (*агент* M_x *получает ant-message от соседнего агента* M_y *или DWA*)
тогда *прочитать оценку* $Q^{A}_{(x(t),\,a_{x(t)})}(t)$ *из заголовка ant-message*
  получить лучшую оценку на текущий момент времени $Q^{M}_{(x(t),\,a_{x(t)})}(t)$
  если $\left(Q^{M}_{(x(t),\,a_{x(t)})}(t) > Q^{A}_{(x(t),\,a_{x(t)})}(t)\right)$ **и** (*цикл не обнаружен*)
  тогда *обновить Q-значение, используя* $Q^{M}_{(x(t),\,a_{x(t)})}(t)$
  в противном случае *не обновлять*

Алгоритм 3: Алгоритм наказания

Если (*punishment-message получено* M_y *от* M_z)
тогда *вычислить вторую лучшую оценку* $Q^{M_y}_{(x(t),\,z')}(t)$ *для доставки на DWA путем использования линии, которая бы не являлась* $l_{y,z}$
отправить сообщение всем соседним агентам-производителям M_{xi}
получить вторую лучшую оценку каждого соседнего агента-производителя M_{xi}
выбрать лучшую оценку среди всех оценок соседей: $\arg\min Q^{M_{xi}}_{(x(t),\,y')}(t)$

если (*вторая оценка* $Q^{M_y}_{(x(t),\,y')}(t)$ *существует*)

тогда
  если (*лучшая оценка* $\arg\min Q^{M_{xi}}_{(x(t),\,y')}(t)$ *соседей существует*)
  тогда
    если $\left(\left(Q^{M_y}_{(x(t),\,z')}(t) < \left(\arg\min Q^{M_{xi}}_{(x(t),\,y')}(t)\right)\right)\right)$
      тогда *наказать линию* $l_{y,z}$: $Q^{M_y}_{(x(t),\,z)}$ $=$*вторая оценка* $Q^{M_y}_{(x(t),\,z')}(t)+\Delta$
      в противном случае *наказать линию* $l_{xi,y}$: *отправить punishment-message*
  в противном случае *наказать линию* $l_{y,z}$:
    $Q^{M_y}_{(x(t),\,z)}$ $=$*вторая оценка* $Q^{M_y}_{(x(t),\,z')}(t)+\Delta$

в противном случае
  если (*вторая лучшая оценка* $\min Q^{M_{xi}}_{(x(t),\,y')}(t)$ *соседей существует*)

тогда *наказать линию* $l_{xi,y}$: $Q^{M_{xi}}_{(x(t),y)}(t)$

$=$*вторая оценка* $Q^{M_{xi}}_{(x(t),y')}(t)+\Delta$

отправить punishment-message с оценкой

Алгоритм 4: Алгоритм двойной Q-маршрутизации

Инициализировать в момент $t = 0$: *все Q-значения* $Q_{x(t),a_{x(t)}}$ *с большими значениями, параметры стимулируемого обучения*: α, γ, ε_*update*, *w*, *w_ants*

Повторить

обновить момент времени t

если (*материал получен агентом-производителем* M_i)

тогда *считать входной вектор x из заголовка материала и переменных среды отправить сообщение агенту* M_i, *к которому поступает материал со значением функции усиления* $r_{(t+1)}$ *и оценкой* $Q^{\text{ц}}_{(x(t),a_{x(t)})}(t)$

выполнить операцию O_q *и выбрать действие по маршрутизации полуфабриката* $a_{x(t)} = M_j$ *в функции входного вектора x путем использования стратегии* ε_*greedy*, *полученной из* $Q_{(x(t),a_{x(t)})}(t)$

отправить материал следующему агенту-производителю $a_{x(t)} = M_j$

на следующем временном шаге получить сообщение от агента M_j *со значением функции усиления* $r(t+1)$ *и оценкой* $Q_{x(t+1),a_{x(t+1)}}(t+1)$

применить правило обновления Q-обучения:

$$Q_{(x(t),a_{x(t)})}(t+1)=Q_{(x(t),a_{x(t)})}(t)+$$

$$+\alpha\left[r(t+1)+\gamma\min_{a_{x(t+1)}}Q_{(x(t+1),a_{x(t+1)})}(t+1)-Q_{(x(t),a_{x(t)})}(t)\right]$$

алгоритм планирования ε_*update*

алгоритм наказания

алгоритм «муравья»

пока *x(t) — не конечное состояние*

Окончание следует.

Литература

1. **Nah F., Rosemann M., Watson E.** Guest editorial: E-business Process Management // Business Process Management J. Emerald Group Publishing Limited. 2004. Vol. 10. N. 1. P. 1–15.
2. **Emelyanov V. V.** Combining multi-agent approach with intelligent simulation in resources flow management // Proc. of the Int. Conf. on Fuzzy Sets and Soft Computing in Economics and Finance FSSCEF. St. Petersburg, Russia / St. Petersburg State Polytechnical University, 2004. Vol. II. P. 311–320.
3. **Wang W., Ryu J., Rivera D. et al.** A Model Predictive Control Approach for Managing Semiconductor Manufacturing Supply Chains under Uncertainty // Annual AIChE Meeting. San Francisco, CA: Omnipress, 2003. P. 1–34.
4. **Fox M., Barbuceanu M., Teigen R.** Agent-Oriented Supply-Chain Management // Int. J. of Flexible Manufacturing Systems. Amsterdam, The Netherlands: Springer Netherlands, 2000. Vol. 12. N. 2–3. P. 165–188.
5. **Radjou N.** Is It Prime Time For Agents In Business? // Proc. of the AAMAS'04. N. Y.: IEEE Computer Society Press, 2004. P. 6–7.
6. **Wolpert D., Kagan T.** An introduction to collective intelligence // Technical Report NASA-ARCIC-99-63. – Mountain View, CA: NASA Ames Research Center, 1999. 88 p.
7. **Littman M., Boyan J.** A distributed reinforcement learning scheme for network routing // Proc. of the Int. Workshop on Applications of Neural Networks to Telecommunications. Hillsdale, NJ: Lawrence Erlbaum Associates, 1993. P. 45–51.
8. **Watkins C.** Learning from Delayed Rewards: PhD Dissertation. – Cambridge, MA: Cambridge University, 1989.
9. **Gaither N., Frazier G.** Operations Management. – Cincinnati, OH: Southwestern Thomson Learning, 2002. – 864 p.
10. **Shapiro J. F.** Modeling the Supply Chain. 2nd Edition. Duxbury: Thomson Learning Inc., 2007. – 618 p.
11. **Hoover W., Eloranta E., Holmström J., Huttunen K.** Managing the Demand-Supply Chain: Value Innovations for Customer Satisfaction. – N. Y.: John Wiley & Sons, 2001. – 272 p.
12. **Julka N., Srinivasan R., Karimi I.** Agent-based supply chain management-1: framework // Computers & Chemical Eng. Shannon, Ireland: Elsevier Ireland Ltd, 2002. Vol. 26. N. 12. P. 1755–1769.
13. **Smirnov A., Sheremetov L., Chilov N., Romero-Cortes J.** Soft-computing Technologies for Configuration of Cooperative Supply Chain // Applied Soft Computing. Amsterdam, The Netherlands: Elsevier, 2004. Vol. 4. N. 1. P. 87–107.
14. **Swaminathan J. M., Smith S. F., Sadeh N. M.** Modeling Supply Chain Dynamics: A Multiagent Approach // Decision Sciences. Atlanta, GA: American

Institute for Decision Sciences, 1998. Vol. 29. N. 3. P. 607–631.
15. **Moyaux T., Chaib-draa B., D'Amours S.** Supply Chain Management and Multiagent Systems: An Overview // Multiagent based Supply Chain Management. Ser. Studies in Computational Intelligence. Heidelberg: Physica Verlag, 2006. Vol. 28. P. 1–27.
16. **Deshpande U., Gupta A., Basu A.** Multi-agent Modeling and Fuzzy Task Assignment for Real-Time Operation in a Supply Chain // Multiagent based Supply Chain Management. Ser. Studies in Computational Intelligence. Heidelberg: Physica Verlag, 2006. Vol. 28. P. 179–202.
17. **Sutton R., Barto A.** Reinforcement Learning: An Introduction. — Cambridge, MA: The MIT Press, 1998. — 342 p.
18. **Bellman R.** On a routing problem // Quarterly of Applied Mathematics. Providence, RI: Brown University, 1958. Vol. 16. P. 87–90.
19. **Chandra C., Kumar S., Smirnov A. V.** E-Management of Scalable Cooperative Supply Chains: Conceptual Modeling and Information Technologies Framework // Human Systems Management. Amsterdam, The Netherlands: IOS Press, 2001. Vol. 20. N. 2. P. 83–94.
20. **Rocha-Mier L. E.** Learning in a Neural Collective Intelligence: Internet packet routing application: PhD Dissertation. — Grenoble, France: National Polytechnic Institute of Grenoble, 2002.
21. **Dorigo M., Stützle T.** Ant Colony Optimization. — Cambridge, MA: The MIT Press, 2004. — 319 p.

ВСЕРОССИЙСКАЯ НАУЧНАЯ МОЛОДЕЖНАЯ ШКОЛА «БИОМЕДИЦИНСКАЯ ИНЖЕНЕРИЯ»

26–30 октября 2009 г.

Место проведения: Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени В. И. Ульянова (Ленина).
Адрес: 197376, г. Санкт-Петербург, ул. Профессора Попова, 5.

Задачи проведения научной школы

Определение перспективных инновационных направлений научных исследований и разработок в области биомедицинской инженерии в России и за рубежом.
Обсуждение актуальных теоретических и практических достижений в области биомедицинской инженерии.
Поиск и выявление оптимальных путей интеграции научных знаний и практических достижений в области биомедицинской инженерии.
Активное привлечение молодых исследователей к научному творчеству.

Формы проведения научной школы

Чтение лекций ведущими отечественными и зарубежными специалистами по фундаментальным и прикладным направлениям развития биомедицинской инженерии.
Практические занятия.
Дискуссии и обсуждение актуальных проблем биомедицинской инженерии с ведущими специалистами за круглым столом.
Выполнение и защита индивидуальных заданий (проектов) участниками научной школы.
Организация конкурса научных работ молодых ученых, аспирантов и специалистов.

Лекции, практические занятия, дискуссии и круглые столы будут проводиться ведущими отечественными и зарубежными специалистами в области биомедицинской инженерии.
По завершении работы научной школы участникам будет выдан сертификат о повышении квалификации в области биомедицинской инженерии, победители конкурса научных работ будут отмечены грамотами и призами.

Издание трудов научной школы

Лучшие работы участников школы по результатам выполнения ими индивидуальных заданий (проектов) будут опубликованы в научно-практических журналах, рекомендованных ВАК РФ для публикации результатов кандидатских и докторских диссертаций.
Будут изданы конспекты лекций, прочитанные ведущими специалистами в области биомедицинской инженерии, а также материалы дискуссий и круглых столов.

Контрольные сроки

Желающие принять участие в работе молодежной научной школы должны в срок до 10 октября 2009 г. направить в адрес оргкомитета заявку по установленной форме.

Дополнительная информация и справки

Организационный комитет конференции:
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени В. И. Ульянова (Ленина), кафедра Биомедицинской электроники и охраны среды.
197376, г. Санкт-Петербург, ул. Профессора Попова, 5.
Подробная информация о проводимой научной школе представлена на сайте университета http://www.eltech.ru, раздел Всероссийская научная молодежная школа «Биомедицинская инженерия».
Эл. адрес: bme@eltech.ru
Телефон/факс: (812) 234–01–33.