

Вісник Харківського національного університету
Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи
управління»

УДК 004.8:004.65

№ 629, 2004, с.132-143

Управление потоками работ в многоагентных системах

В. А. Ермолаев, С. Л. Плаксин

Запорожский государственный университет, Украина

In the present paper the problem of the information system modeling is considered. The approach to the data modeling using the algebraic structures was made. As a result the mathematical data model has been built. The model obtained consists two levels, these are structural and semantic ones. The mechanism of the correctness verification of the data model was developed.

1. Общая постановка задачи и её актуальность

Управление бизнес-процессами и автоматизация их выполнения по праву являются критическими задачами в таких областях как моделирование деятельности предприятия (Enterprise Modeling), моделирование информационных систем корпорации/предприятия. Доминирующим подходом при моделировании бизнес-процессов является его представление в виде потока работ (workflow). «Системы потоков работ доказали свою применимость для управления «административными» процессами – которые характеризуются понятной, хорошо определенной структурой и постоянной предсказуемой формой – до некоторого времени. В настоящее время существует общее мнение, что современные системы являются недостаточно гибкими для представления сложных динамических процессов с изменчивым контекстом» (цит. из [1]). Дополнительный фактор, влияющий на сложность применения стандартных решений, состоит в том, что знания о том, как должен выполняться бизнес-процесс, не сосредоточены в одном месте, а наоборот, распределены между разными исполнителями на разных организационных уровнях и блоках. Определить весь поток работ целиком становится невозможным в открытой среде (какой, например, является electronic marketplace), а также считая исполнителей более заинтересованными в своей выгоде, нежели бескорыстными, либо считая, что исполнители вынуждены действовать в среде с ограниченным уровнем доверия.

Данная работа касается вопросов кооперации в динамических открытых распределенных бизнес процессах, реализуемых коалициями рациональных агентов, выполняющих задания. Согласно спецификации BPMN [2] (языка разметки бизнес-процессов), под бизнес-процессами понимаются частично упорядоченные множества атомарных или неатомарных работ выполняемых при взаимодействии участников в соответствии с определенным множеством правил для достижения общей цели. Агенты представляют разумных актеров, обладающих своими собственными ролями [1], как членов сообщества организационных блоков, обладающих знаниями о том как разделить и исполнить ту или иную деятельность.

Эти актеры формируют коалиции для оптимального выполнения задач, возникающих в бизнес-процессе. В этом случае, модель потока работ для

выполнения бизнес-процесса генерируется «на лету» и в дальнейшем может быть использована для анализа производительности и точной настройки.

Модели и шаблоны кооперации между агентами являются ключевым моментом для исполнения и управления распределенным бизнес-процессом. Важным фактором является необходимость принятия во внимание баланса между такими свойствами актеров как рациональность, заинтересованность в своей выгоде и бескорыстность (в том смысле, что актеры готовы рационально подходить к совместной работе).

В работе предлагается агент-ориентированная среда кооперации в области управления бизнес-процессами, а также модели кооперации.

Статья организована следующим образом. В разделе 2 сделан обзор существующих подходов моделирования кооперации агентов. В разделе 3 поставлена цель данной работы. В разделе 4 описана модель многоагентной системы, выполняющей потоки работ. В разделе 5 предложена модель кооперации. Раздел 6 содержит выводы и планы будущей работы.

2. Истоки исследования авторов

В работе [3] подчеркивается, что "...Центральным в изучении кооперации является понятие социальной структуры. Социальная структура – это множество отношений, которые наблюдаются между агентами в обществе. Эти отношения определяют зависимости, которые возникают между агентами ... и определяют права и обязанности каждого агента в обществе во взаимоотношениях с коллегами. Для эффективной кооперации с коллегами агент должен иметь репрезентацию тех социальных структур, в которых он участвует, и делать логические выводы над такими представлениями. Процесс логического вывода используется для ответа на вопросы о возможности кооперации, а также для выяснения своего положения по отношению к другим агентам в сообществе...". Социальные структуры рассматриваются как организации в области управления и выполнения бизнес-процессов.

Абстрактное представление кооперирующих структур, в которых агенты взаимодействуют для достижения целей от имени друг друга и обоснование осуществимости такой кооперации приведены в [3]. Децентрализованная декомпозиция задачи на работы и механизмы размещения работ показываются в [4]. Подробный обзор подходов к переговорам при предоставлении распределенных сервисов можно найти в [5]. В области E-Commerce, например, предложены модели формирования коалиции, основанные на пред- и постпереговорах, которые тестируются в COALA3 - среде общего назначения для изучения коллективного поведения в коалициях агентов. В работе [6] исследуются методы для размещения задач, основанные на формировании коалиции агентов. Подход используемый в данной работе близок к подходу переговоров, ориентированных на сервисы [7], который включает определение контракта с конкретными условиями и сроками. Механизмы предоставления сервисов путем проведения переговоров, обсуждаемые в данной работе, близки к механизмам, предложенным в ADEPT [8]. Среда для переговоров, основанных на аргументации (argumentation-based negotiation) представлена в [9]. Емкостной подход (capability based approach) к нахождению подходящих сервисов предложен для определения нужного кандидата на роль сервис-провайдера.

Методы для выравнивания (adjusting) поведения агентов в кооперативном окружении, основанные на обучающих техниках, предложены в [10]. Образцы координации и языки предложены в [11].

3. Нерешенные проблемы и цели работы

Модели и шаблоны кооперации между агентами являются, ключевым моментом для исполнения и управления распределенным бизнес-процессом.

Классическими методами исследования кооперации являются методы теории игр. Управление в иерархических структурах в организационно-экономических системах моделируется в теории активных систем [12].

Основным недостатком стандартных моделей теории активных систем (задача стимулирования, задача согласованного планирования и др) является их статичность. Кроме того, в моделях теории игр предполагается, что игрок (центр) обладает полной информацией о своих собственных предпочтениях, о предпочтениях остальных игроков (активных элементах).

В отличие от теоретико-игровых моделей в многоагентных системах решения принимаются последовательно, в процессе переговоров. Это позволяет агентам получать недостающую информацию для принятия решений.

Целью данной работы является разработка кооперативной модели управления потоками работ при выполнении бизнес-процессов.

4. Модель многоагентной системы, выполняющей потоки работ

Наличие в реальном мире бизнес-процессов ясно отражает тот факт, что не существует в современных организациях таких исполнителей, которые в состоянии выполнять ту или иную работу целиком самостоятельно. Такие работы чаще выполняются в кооперации группами людей и/или искусственных актеров.

Каждый из таких исполнителей занимает определенную позицию на том или ином уровне организации и характеризуется своими возможностями, обязательствами, правами. Каждый из исполнителей обладает собственными знаниями о том, что означает та или иная работа, как ее можно разделить на частично упорядоченное множество простых работ, какие из этих работ нужно отдать подчиненным или экспертам. Обычно, исполнение работы инициируется исполнителями более высоких организационных уровней, чьи знания о конкретной задаче являются более общими и абстрактными.

Части работы становятся более детальными при переходе по уровням иерархии организации вниз, к исполнителям с более специфичными возможностями и правами. Работы, являющиеся атомарными для «босса», очевидно, могут рассматриваться исполнителями как сложные задачи. Обычно, на любом уровне, исполнитель кооперируется со своими непосредственными «начальниками», сотрудниками своего уровня и своими подчиненными (если они есть) и совершенно не должен знать всех исполнителей верхних уровней, всех исполнителей своего уровня и всех подчиненных.

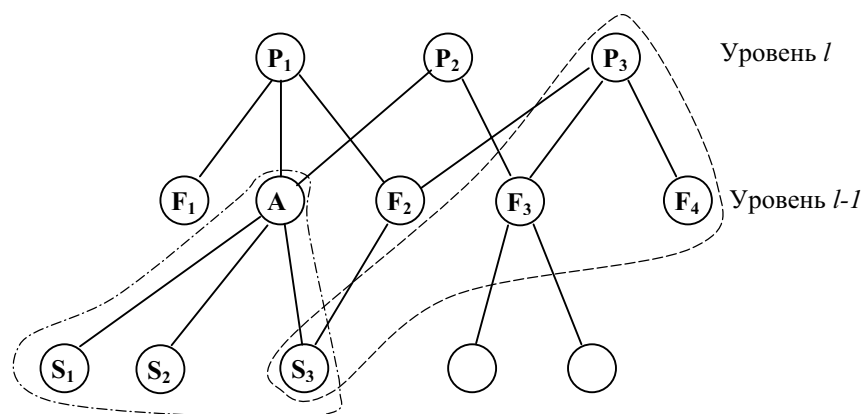


Рис. 1. Графовая модель организации

Если организация имеет структуру в виде графа (как на Рис. 1), сфера ответственности актера A может быть ограничена узлами верхнего (P_1 и P_2) и нижнего уровней (S_1, S_2, S_3) смежных с A , а также узлами того же уровня, что и A , с которыми A связан через своего «начальника» (F_1, F_2, F_3).

Организационный блок (например, отдел) уровня l состоит из исполнителя уровня l плюс всех его подчиненных (например, $\{A, S_1, S_2, S_3\}$). Некоторые исполнители могут участвовать в нескольких организационных блоках (напр., S_3 в $\{A, S_1, S_2, S_3\}$ на уровне l и $\{P_3, S_3, F_3, F_4\}$ на уровне $l-1$).

Такие исполнители могут принадлежать к сферам ответственности внешних (по отношению к организации) актеров и могут воспринимать внешние воздействия от членов разных организационных блоков. Исполнители способны воспринимать внешние воздействия через организационные блоки – посредники (Proxies, например, A, P_3).

Посредник, если рассматривать его извне по отношению к организационному блоку, является простым исполнителем – функциональным компонентом. Он представляет организационный блок (функциональную систему) в организационному блоку верхнего уровня (например, в системе ADEPT [8] и отделы и индивидуальные исполнители представляются агентами.)

Организация, очевидно, является множеством организационных блоков на уровне l . В соответствии с принципами организационного структурирования (напр, см. [13]) предполагается, что организационный блок есть множество активных сущностей (актеров) обладающих соответствующими возможностями и общающимися в соответствии с данным множеством образцов. Актеры моделируются экономически рациональными программными агентами, разработанными в рамках [14]. Возможности агента обеспечиваются в виде множества макромоделных программ для исполнения работ [14].

Организации и их функциональные блоки взаимодействуют с внешней средой. Функциональная система обеспечивает окружения для своих функциональных компонентов, которые, в свою очередь, могут распадаться на функциональные системы более низкого уровня. Так как основной миссией организации является выполнение бизнес-процессов, предполагается, что

окружение моделируется при помощи генерирующей функции, задающей задачи (*tasks*) T (см. [15] где используется подобная терминология) как множества работ w_i :

$$E \rightarrow T = \{w_1, w_2, \dots, w_k\}$$

Эти задачи воспринимаются посредниками. Организация, таким образом, предназначена для выполнения задач, которые воспринимаются из внешней среды как внешние воздействия.

Предполагается, что задача $T = \{w_1, w_2, \dots, w_k\}$ есть множество атомарных (для данного актера) работ. Актеры внутри организации в состоянии исполнять атомарные деятельности, принадлежащие к множеству допустимых атомарных работ. Эти актеры также в состоянии генерировать (под)задачи без внешнего воздействия, а реагируя на некоторые события (внутренние по отношению к организации) или в процессе выполнения той или другой атомарной работы.

Для выполнения каждой конкретной атомарной работы актеры формируют субъективные Частичные Локальные Планы (Partial Local Plans, ЧЛП). ЧЛП формализуются в виде «легковесных» онтологий задач и представляются в языке Standard OIL [16]. ЧЛП отличаются от глобальных планов (GPGP [17]) в том смысле, что ЧЛП не содержат субъективных представлений о том, какими могут быть действия других актеров. В то же время, обновления информации об изменениях в возможностях актеров, оценок кредитоспособности (*credibility*) проводятся каждым актером индивидуально в процессе их кооперативной деятельности. Актеры вовлечены в кооперативное выполнение задач либо по результатам торгов по назначению задачи или отдельных работ, или путем получения директивы от непосредственного начальника.

После того, как воздействие воспринято актером, он может:

- принять и выполнить некоторые из работ, составляющих задачу;
- отклонить выполнение некоторых работ;
- принять решение о передаче множества работ одному из партнеров, в соответствии со своими представлениями о возможностях партнеров, кредитоспособности, и готовности к выполнению работы;
- потребовать выполнения некоторой новой работы, исполнение которой (согласно знаниям, формализованном виде онтологии задачи) является существенным для успешного завершения всей задачи.

Модель функционального компонента (см. Рис. 2) строится на идее «поглощения» и «генерации» работ из множества допустимых работ $W = \{w_1, w_2, \dots\}$ для этого функционального компонента. Предполагается, что сенсорных вход функционального компонента i принимает задачу $W_i \subseteq W$. Некоторая часть работ W_i^p может быть выполнена («абсорбирована») данным компонентом и оставшаяся часть работ может быть либо делегирована другим компонентам системы – W_i^d , либо отклонена – W_i^r . Функциональный компонент может также генерировать дополнительное множество работ W_i^g для

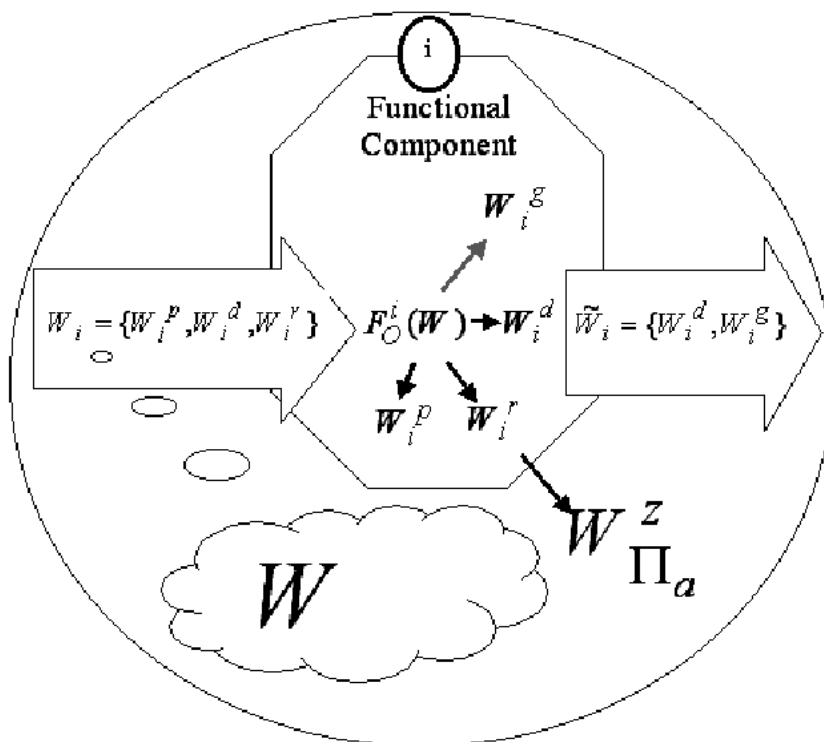


Рис. 2. Модель функционального компонента

выполнения работ W_i^p . W_i^g так же как и W_i^d могут делегироваться другим компонентам:

$$W_i \rightarrow F_O^i(W) \rightarrow \tilde{W}_i,$$

где: $W_i = \{W_i^p, W_i^d, W_i^r\}$, $\tilde{W}_i = \{W_i^d, W_i^g\}$, $F_O^i(W)$ – макромоделльные программы.

В специальном случае компонент i может сгенерировать новое множество работ W_i^g без внешнего воздействия W_i - т.е. может «вызвать» новую задачу.

$$F_O^i(W) \rightarrow \tilde{W}_i,$$

где: $\tilde{W}_i = \{W_i^g\}$, $F_O^i(W)$ – макромоделльная программа.

Процесс Π_a исполнения задачи начинается с принятия или генерации новой задачи $W_a \subseteq W$. Задача W_a , так же как и все производные задачи \tilde{W}_a , рассматривается как связанная с процессом Π_a и помечена уникальным идентификатором процесса. Компонент считается связанным с процессом Π_a в случае, если он поглощает часть W_a , \tilde{W}_a , или генерирует W_a^g .

Процесс Π_a считается завершенным в случае, когда все компоненты прекратили поглощение атомарных работ из задач, связанных с процессом Π_a .

Множество работ $W_{\Pi_a}^z$ не поглощенных процессом Π_a помечается как множество невыполнимых работ.

На практике, будем считать множество допустимых работ функционального компонента ограниченным: $W = \{w_1, w_2, \dots, w_n\}$.

5. Модели координации

Координация внутри функциональной системы, выполняющей конкурентные задачи является, критическим моментом, поскольку предоставляет возможности исполнителям работать согласованно. Модели координации для кооперативного выполнения задачи решают две главные задачи управления: первая – переговоры при назначении работ и формирование коалиции для решения задачи, и вторая – обеспечение правильного порядка выполнения работ.

5.1 Переговоры при назначении работ

В описываемом типе организаций не существует исполнителей, выполняющих всю задачу целиком. Каждый исполнитель выполняет работы, которые он в состоянии выполнить, и передает оставшиеся работы своим коллегам. Процедура передачи происходит всякий раз, когда возникает такая необходимость. Образец координации для назначения работ называется Фаза Назначения (*Arrangement Phase*). Цель фазы назначения состоит в поиске исполнителя некоторой работы, которая (в соответствии с ЧЛП или из-за перегрузки исполнителя) должна быть передана другому исполнителю. Назначение работ производится путем проведения торгов между Инициатором и группой актеров-участников. Цель Инициатора состоит в исполнении работы наиболее оптимальным образом. Предполагается, что в процессе переговоров о назначении работ Инициатор решает задачу двухкритериальной оптимизации. Первый критерий касается оптимальности времени исполнения работы. Вторым критерий – оптимальная плата (*incentive*), которая должна быть заплачена партнерам за выполнение работы. Инициатор предлагает работу некоторым своим партнерам, которые, как он предполагает, смогут выполнить такую работу и которые являются достаточно кредитоспособными для того, чтобы завершить работу к назначенному сроку. Актеры, при получении предложения о работе, становятся участниками переговоров. Ответы Участников отражают степень готовности к выполнению предложенной работы. Участник готов выполнить работу, если его ответ содержит пересечения с функцией желательности, предложенной Инициатором переговоров. Иначе, он отвергает предложение. Инициатор затем выбирает лучший лот из множества предложенных пересечений своей функции желательности с полученными оценками. Участник, чей лот был признан оптимальным, становится Контрактором (*Contractor*). Он присоединяется к коалиции исполнения задачи и начинает выполнение работы.

Вступая в коалиции исполнения задачи, актер обязуется следовать нескольким системным правилам, которые регулируют уровень между бескорыстностью и своими интересами. Такие правила могут быть классифицированы, следуя гипотезе Дженнингса о Традициях и Требованиях (*Commitment-Convention hypothesis* [18]) как индивидуальные и коалиционные требования, и коалиционные традиции:

Правило 1: Требование относительной кооперации. Члены коалиции подтверждают свое относительное стремление при кооперации достичь общей цели: выполнить задачу с максимально возможной эффективностью (максимальным качеством, сбалансированной нагрузкой, минимальным временем, ...). Исполнение этого требования зависит от разногласий между персональными намерениями актера и общей целью коалиции задачи.

Правило 2: Традиция при назначении работ. На фазе назначения участник коалиции, предлагающий работы (Инициатор) обязывается предоставить правдивую информацию о функции желательности результатов. С другой стороны, участники торгов должны отправить правдивую информацию о степени готовности к исполнению работы, путем сообщения информации о разделяемой емкости, которая будет потрачена на выполнения задачи.

Правило 3: Требование доставки результатов. С момента, когда актер принимается за выполнение работы, он обязуется безусловно закончить эту работу и предоставить результату сразу же по ее окончании.

5.2 Координация потока работ

Работы, выполняемые в процессе исполнения задачи, могут быть неявно зависимыми друг от друга (как, например, в TÆMS [11]): сильно связанными – работа w^i существенно зависит от результатов работы w^j в случае, если результаты работы w^j являются необходимыми для начала работы w^i , или слабо связанными – работа w^j облегчает исполнение работы w^i . Таким образом, поток работ ограничен отношением частичного порядка. Иногда такой поток может содержать итеративные или рекурсивные циклы. Актер может генерировать подзадачу, содержащую, или приводящую к выполнению атомарной работы, которая, в конце концов, будет назначена этому же актеру. Координация последовательности выполнения работ нужна для управления эффективностью и согласованностью выполнения потока работ, так же как и для всей задачи.

Модель координации последовательности выполнения работ базируется на использовании зависимостей между работами. Субъективные знания о таких зависимостях находятся в ЧЛП актеров и формализованы в виде онтологии задачи. На основании этих знаний актер (пусть A) может составить предварительный **план $\eta(w, t)$ выполнения работ**, из полученного задания $W_A = \{w_1, w_2, \dots, w_n\}$. Функция $\eta(w, t)$ определяется следующим образом:

$$\eta(w, t) = \begin{cases} 1, & \text{если работа } w \text{ должна выполняться на интервале } [t_n, t_n + \tau), \\ 0, & \text{если работа } w \text{ не должна выполняться на интервале } [t_n, t_n + \tau) \end{cases}$$

Здесь $w \in W_A$, t_n - дискретные моменты времени с интервалом τ , $n \in \{0, 1, 2, \dots\}$.

На рисунке 3 приведены примеры графиков функции плана для работ w^i и w^j . Период времени выполнения работы может быть разрывным, как это показано на рисунке 3 для работы w^j .

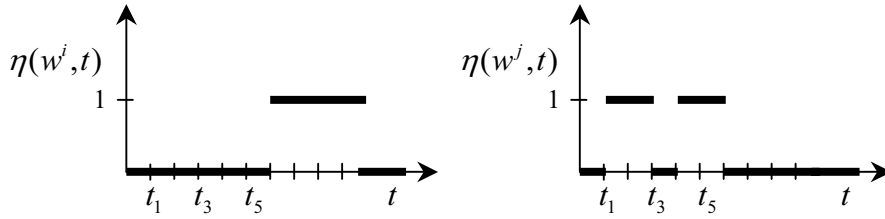


Рис. 3. Примеры функций плана выполнения работ.

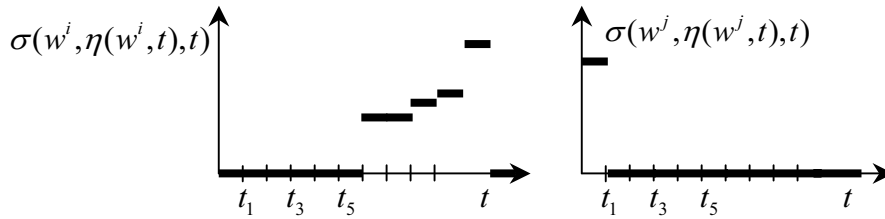


Рис. 4. Примеры функций стимулирования.

Полученную функцию предварительного плана выполнения задания агент может сохранить в базе знаний для повторного использования в будущем.

Далее актер A выбирает **функцию стимулирования** $\sigma(w, \eta, t)$. Функция стимулирования $\sigma_B(w^j, \eta(w^j, t), t)$ определяет выигрыш актера B , который будет выполнять работу w^j по плану $\eta(w^j, t)$. Примеры функций стимулирования показаны на рисунке 4.

Содержательно, функция стимулирования распределяет выигрыш актера-исполнителя по мере выполнения задания в соответствии с планом. Это может быть либо “предоплата”, когда выигрыш выплачивается перед выполнением работы. Как раз этот случай изображен на рисунках 3 и 4 для работы w^j . В некоторых случаях выигрыш может распределяться равномерно на протяжении всего периода времени, либо каким-нибудь другим способом.

В общем случае функция стимулирования σ зависит от актера-исполнителя, точнее от *оценки его возможностей и кредитоспособности* (подробнее об этих показателях изложено в [17]). Тогда функцию стимулирования можно записать $\sigma(A, w, \eta, t)$. Здесь $A \in \{A_1, A_2, \dots, A_k\}$ - множество актеров, $w \in W$, η - функция плана выполнения работ, t_n - дискретные моменты времени с интервалом τ , $n \in \{0, 1, 2, \dots\}$.

Если актер A выполнит работу w в соответствии с планом η , то он получит общий выигрыш в сумме $G_A(w, \eta) = \sum_t \sigma(A, w, \eta, t)$.

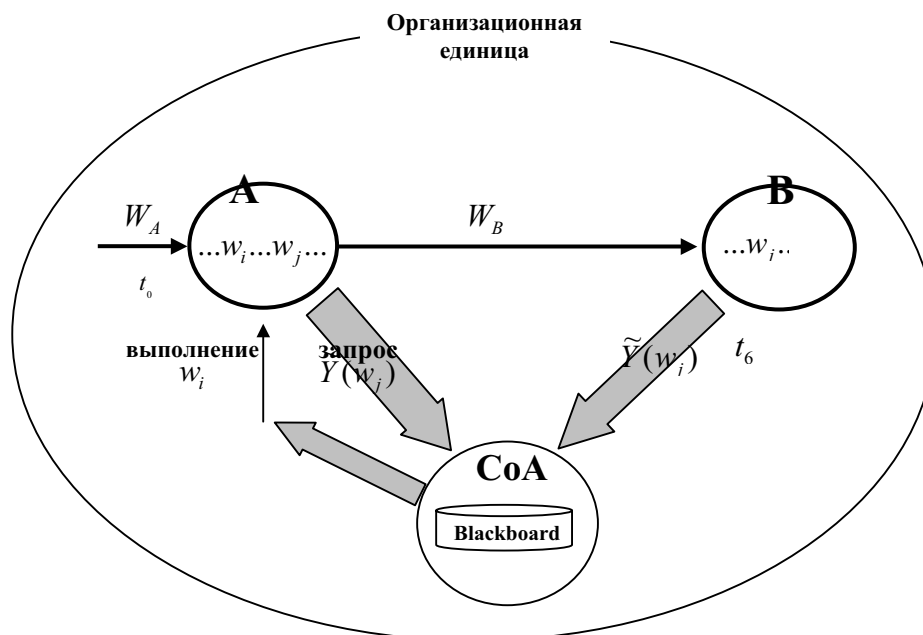


Рис. 5. Координация потока работ при помощи CoA

Именно план $\eta(A_k, w^i, t)$ выполнения работы w^i и функция стимулирования $\sigma(A_k, w^i, \eta(A_k, w^i, t), t)$ являются объектом переговоров при распределении работ (в данных обозначениях актеру A_k предлагается выполнить работу w^i).

Использование функций плана η и стимулирования σ позволяет актерам выполнять задания в правильной последовательности.

При выполнении потоков работ, не редко могут возникать ситуации, когда результаты выполнения одной работы, являются входными параметрами другой работы (возможно даже не одной). Кроме того, не всегда эти работы выполняются одна за другой. В подобных ситуациях часто используют доску объявлений (blackboard), на которой размещают результаты выполнения промежуточных работ.

В данной работе такая доска объявлений контролируется специальным агентом – Агентом Координации (Coordination Agent, CoA). Исполнители, завершившие выполнение своих работ, публикуют результаты на доске объявлений CoA. Актеры, собирающиеся выполнить работы, зависящие от других, обращаются к CoA за нужными данными.

Пусть в момент времени t_0 актер A соглашается выполнить задачу $W_A = \{w_i, w_j\}$ (рис. 5). После декомпозиции и анализа W_A A решает выполнить работу w^i самостоятельно и передать работу w^j партнерам. После завершения фазы назначения, A передает работу w^j актеру B . Пусть функции планов выполнения работ такие, как указано на рисунке 3.

Тогда B исполняет w^j , в точно запланированное время $[t_1, t_6)$ и публикует результаты $\tilde{Y}(w^j)$ у CoA. К моменту времени t_6 $\tilde{Y}(w^j)$ уже являются

доступными, A запрашивает результаты $\tilde{Y}(w^j)$ у CoA , получает их и использует $\tilde{Y}(w^j)$ в качестве параметров для w^j .

6. Выводы по результатам и направления дальнейших исследований

Кооперация при выполнении бизнес-процессов автономными, распределенными актерами, обладающими рациональным и иногда противоречивым поведением, в открытой организации является достаточно сложным вопросом. Несмотря на то, что Inverno, Luck и Woldridge считают кооперацию “несомненно наиболее хорошо изученным процессом в исследованиях по МАС” (цит. по [3]), в данной области все еще есть много открытых проблем. Одной из самых сложных проблем является отсутствие консенсуса в том, как все это должно быть организовано и структурировано.

В данной работе выделено две главные задачи координации кооперативного выполнения потоков работ: первая – переговоры при назначении работ и формирование коалиции для решения задачи, и вторая – обеспечение правильного порядка выполнения работ, и предложены механизмы их решения. Первая задача решается при помощи протокола контрактных сетей. Вторая – при помощи механизма стимулирования и доски объявлений.

Дальнейшая работа по данному исследованию будет заключаться в разработке правил принятия решений агентами, участвующими в переговорах по динамическому формированию коалиции, а также верификации правил, разработке и реализации агент-ориентированной среды кооперации.

ЛИТЕРАТУРА

1. J. Moore, R. Inder, P. Chung, A. Macintosh, and J. Stader: "Who Does What? Matching Agents to Tasks in Adaptive Workflow."; In Proceedings of the 2nd International Conference on Enterprise Information Systems (ICEIS 2000), B. Sharp, J. Cordeiro, and J. Filipe (eds), Stafford, July 2000, pp 181-185, ISBN 972-98050-1-6.
2. BPMI.org, Business Process Modeling Language (BPML), Working Draft 0.4, August, 2001, <http://www.bpmi.org> last accessed 12.01.2002.
3. M. d'Inverno, M. Luck, and M. Wooldridge, (1997). Cooperation structures. IJCAI '97.
4. William E. Walsh and Michael P. Wellman. A market protocol for decentralized task allocation. In: Proc. of the Third International Conference on Multi-Agent Systems, pages 325--332, 1998.
5. Faratin, P.: Automated Service Negotiation Between Autonomous Computational Agents. PhD Thesis, Univ. of London (2000)
6. O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. Artificial Intelligence, 101(1-2):165--200, 1998.
7. Faratin, P., Jennings, N. R., Buckle, P., Sierra, C.: Automated Negotiation for Provisioning Virtual Private Networks Using FIPA-Compliant Agents. In: Proc. 5th Int. Conf. on the Practical Application of Intelligent Agents and Multi-Agent Systems (PAAM-2000), Manchester, UK (2000) 185–202

8. Jennings, N. R., Faratin, P., Norman, T. J., O'Brien, P., Odgers, B. Alty, J. L.: Implementing a Business Process Management System using ADEPT: A Real-World Case Study. *Int. Journal of Applied Artificial Intelligence* 14 (5) (2000) 421-465
9. C. Sierra, N.R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In M.P. Singh, A. Rao, and M.J. Wooldridge, editors, *Proc. ATAL-97*, pages 177--192, Berlin, Germany, 1998. Springer-Verlag.
10. Hitoshi Matsubara, Itsuki Noda, and Kazuo Hiraki. Learning of cooperative actions in multi-agent systems: a case study of pass play in soccer. In *Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium*, pages 63--67, Menlo Park, CA, March 1996. AAAI Press. AAAI Technical Report SS-96-10
11. Keith S. Decker, *Environment Centered Analysis and Design of Coordination Mechanisms*. Umass CMPSCI Tech. Rep. 95-69, 1995, 199 p.
12. Бурков Н. В., Новиков Д. А., *Теория активных систем: состояние и перспективы*. М: Синтег, 1999. – 128 с.
13. Gasser, L., "DAI Approaches to Coordination" in *Distributed Artificial Intelligence: Theory and Praxis* (eds. N. M. Avouris and L. Gasser) Kluwer Academic Publishers pp 31-51.
14. V. A. Ermolayev, S. U. Borue, V. A. Tolok, N. G. Keberle: Use of Diakoptics and Finite Automata for Modelling Virtual Information Space Agent Societies // "Lecture Notes of Zaporozhye State University", ISBN 966-599-058-4, Vol. 3, No 1, 2000, pp. 34-44.
15. Foundation for Intelligent Physical Agents (FIPA) Spec: FIPA Agent Message Transport Service Specification Doc No XC00067B, 2000, <http://www.fipa.org>, last accessed Jan 19, 2002.
16. D. Fensel, M. Crubezy, F. van Harmelen, and I. Horrocks: OIL & UPML: A Unifying Framework for the Knowledge Web. In *Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI'00*, Berlin, Germany August 20-25, 2000
17. V. A. Ermolayev, S. L. Plaksin: Cooperation Layers in Agent-Enabled Business Process Management. In: *the Proceedings of The 3-d Intl. Scientific-Practical Conference on Programming (UkrPROG'2002)*, May, 21-24, 2002, Kiev, Ukraine, p. 354-368
18. Jennings, N. R.: *Coordination Techniques for Distributed Artificial Intelligence*. In: (O'Hare, G. M. P., Jennings, N. R. Eds.) *Foundations of Distributed Artificial Intelligence*. Wiley (1996)