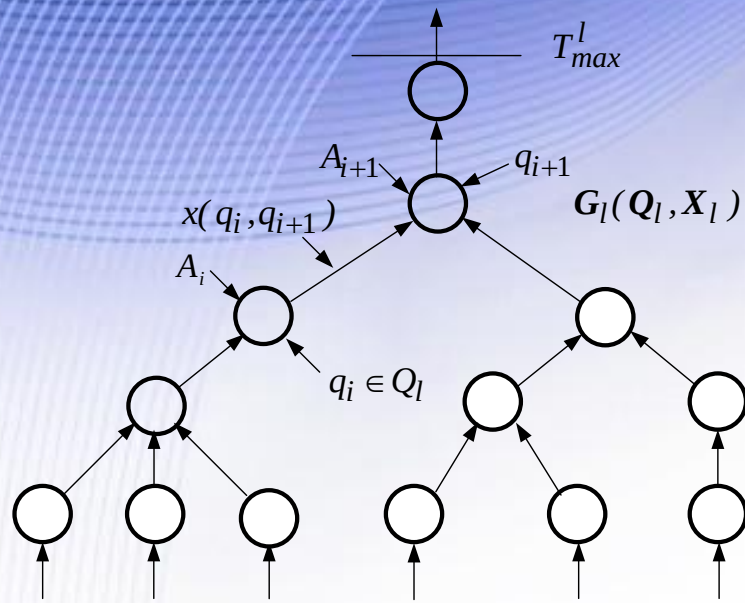


Мультиагентное диспетчирование ресурсов в гетерогенной облачной среде

И.А. Каляев, А.И. Каляев, Я.С. Коровин

Общая постановка проблемы



Граф задачи Z_l

ОВС должна решать некоторое множество (поток) различных крупномасштабных задач $Z = \langle Z_1, Z_2, \dots, Z_M \rangle$, которые могут поступать от пользователей в случайные моменты времени. При этом крупномасштабная задача $Z_l \in Z$ представляется в виде графа $G_l(Q_l, X_l)$, вершинам $q_j \in Q$ которого приписаны некоторые подзадачи $A_j \in A = \langle A_1, A_2, \dots, A_C \rangle$, причем для каждой подзадачи A_j известна ее вычислительная трудоемкость V_j . Пользователь устанавливает момент времени T_{max}^l , к которому он желал получить решение своей задачи Z_l , а также количество премиальных баллов (бонусов) O_l , которое он готов заплатить за это.

В состав ОВС входит множество вычислительных ресурсов $R = \langle R_1, R_2, \dots, R_N \rangle$. Ресурс $R_i \in R$ может решать некоторый набор (подмножество) типов подзадач $A = \langle A_1^i, A_2^i, \dots, A_L^i \rangle \subseteq A$ ($i = 1, 2, \dots, N$), причем его реальная производительность при решении подзадачи $A_j^i \in A_i$ составляет $S_j^i = F(A_j^i)$ ($j = 1, 2, \dots, L$).

Необходимо таким образом распределить подзадачи потока Z по вычислительным ресурсам $R_1, R_2, \dots, R_N$, чтобы все задачи $Z_l \in Z$ были выполнены к требуемым моментам времени T_{max}^l .

Варианты постановки проблемы

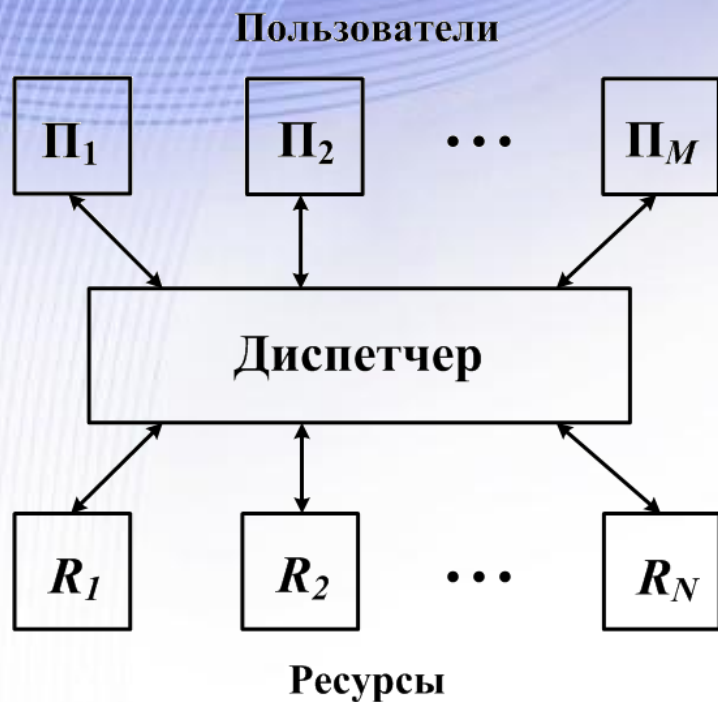
Вариант 1. Все ресурсы $R_j (j = \overline{1, N})$ ОВС одинаковы, и могут решать одинаковые наборы подзадач A , т.е. $A_i = A_j = A (i = \overline{1, N}, j = \overline{1, N})$, причем производительность всех ресурсов $R_j (j = \overline{1, N})$ при решении идентичных подзадач $A_d \in A$ также одинакова, т.е. $S_d^i = S_d^j (i = \overline{1, N}, j = \overline{1, N})$.

Вариант 2. Ресурсы $R_j (j = \overline{1, N})$ ОВС решают одинаковые наборы подзадач, т.е. $A_i = A_j = A (i = \overline{1, N}, j = \overline{1, N})$, однако их производительность при решении идентичных подзадач $A_d \in A$ различна, т.е. $S_d^i \neq S_d^j (i = \overline{1, N}, j = 1, 2, \dots, i-1, i+1, \dots, N)$.

Вариант 3. Ресурсы $R_j (j = \overline{1, N})$ ОВС ориентированы на решение различных наборов подзадач, т.е. $A_i \neq A_j (j = \overline{1, N}, i = 1, 2, \dots, j-1, j+1, \dots, N)$, однако время выполнения однотипных подзадач ($A_i \cap A_j \neq \emptyset$), однако производительность различных ресурсов $R_i \in R$ и $R_j \in R$ при решении идентичных подзадач $A_d \in A_i \cap A_j$ одинакова.

Вариант 4. Ресурсы $R_j (j = \overline{1, N})$ ОВС ориентированы на решение различных наборов подзадач, т.е. $A_i \neq A_j, A_i \cap A_j \neq \emptyset$, причем их производительность при решении подзадач идентичных типов $A_d \in A_i \cap A_j$ также различна, т.е. $S_d^i \neq S_d^j (i = \overline{1, N}, j = 1, 2, \dots, i-1, i+1, \dots, N)$.

Централизованный диспетчер ОВС



Недостатки:

1. Сложность оперативного распределения гетерогенных ресурсов по задачам потока, поступающим в заранее неизвестные моменты времени.
2. Сложность масштабирования ресурсов ОВС.
3. Низкая отказоустойчивость

Децентрализованный мультиагентный диспетчер ОВС



1. Пользователь формирует свою задачу $Z_l = Z$ в виде графа $G_l(Q_l, X_l)$, устанавливает момент времени T_{max}^l , к которому он желает получить ее решение, а также величину премиальных баллов O , которую он готов заплатить за ее решение. Дескриптор задачи размещается на ДО.

2. Агент ресурса R_i опрашивает ДО в поисках работы для своего ресурса. При обнаружении пользовательской задачи агент оценивает экономическую целесообразность своего участия в ее решении.

3. Если относительная стоимость решения задачи удовлетворяет агента, то он выделяет в графе $G_l(Q_l, X_l)$ фрагмент задачи, который он может решить к заданному моменту времени с помощью своего ресурса. Если такой фрагмент обнаруживается, то агент вступает в сообщество R_l по решению данной задачи и приступает к его решению.

4. Агент, вступивший в сообщество R_l , контролирует ход решения принятого к исполнению фрагмента задачи Z_l , периодически оценивая время получения конечного результата. В случае, если агент не успевает обеспечить решение фрагмента к требуемому моменту он сообщает об этом на ДО и выходит из состава сообщества R_l .

5. После успешного завершения решения задачи Z_l всем агентам, сообщества R_l , принявшим участие в ее решении начисляются премиальные баллы, пропорциональные объему выполненных ими вычислений.

Понятие нити

Нить – это последовательность вершин $\mathbf{H}_f = \langle q_1^f, q_2^f, \dots, q_k^f \rangle$ графа $G_l(Q_l, X_l)$ задачи Z_l , в которой вершины q_j^f и q_{j+1}^f ($j=1, 2, \dots, k-1$) соединены дугой $x(q_j^f, q_{j+1}^f)$. Иными словами, нить определяет последовательность подзадач задачи Z_l , в которой каждая последующая подзадача использует результаты выполнения предыдущей подзадачи, и соответственно эти подзадачи могут выполняться только последовательно.

Длина нити $\mathbf{H}_f$ – время t_f^p , затрачиваемое ресурсом $R_p \subseteq \mathbf{R}$ на ее решение, т.е.

$$t_f^p = \sum_{i=1}^k \frac{V_i^f}{S_i^p} + t_{\Pi}^p,$$

где V_i^f – вычислительная трудоемкость подзадачи A_i , приписанной вершине q_i^f ($i=1, 2, \dots, k$) нити $\mathbf{H}_f$;

S_i^p – производительность ресурса R_p при решении подзадачи A_i ;

t_{Π}^p – время пересылки результата решения нити $\mathbf{H}_f$ потребителю (либо другому ресурсу ОВС, либо пользователю). Будем считать, что время t_{Π}^p не зависит от конкретного вычислительного ресурса R_p и равно t_{Π} .

Требуемый момент времени начала выполнения подзадачи A_d , приписанной вершине q_d^f нити $\mathbf{H}_f$ определяется как:

$$T_d^f = T_{k+1}^f - \left(\sum_{i=d}^k \frac{V_i^f}{S_i^p} + t_{\Pi}^p \right) \quad (i=1, 2, \dots, k-1).$$

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 1

$$A_j = A_i = A \quad (j = \overline{1, N}, i = \overline{1, N})$$

$$S_d^i = S_d^j \quad (j = \overline{1, N}, i = \overline{1, N})$$

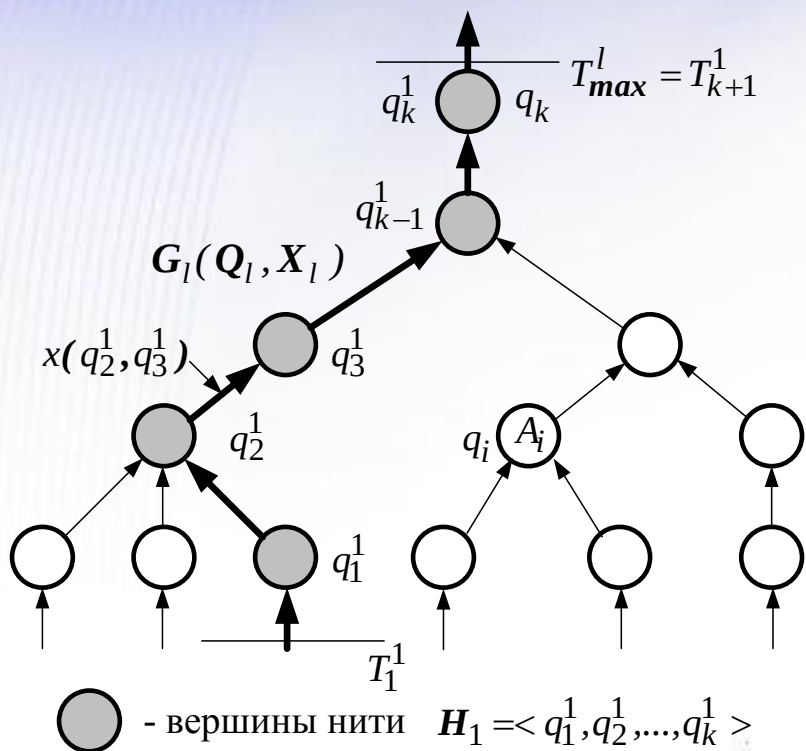
Агент R_p опрашивает ДО в поисках работы. При обнаружении на ДО дескрипторе задачи Z_l агент R_p :

1. Определяет относительную стоимость вычислений как $D = O/V$, где $V = \sum_{i=1}^M V_i$ – суммарная вычислительная трудоемкость задачи Z_l ; O – количество премиальных баллов, предлагаемых пользователям.

2. Если $D \geq D_p$, где D_p – некоторый порог, то агент R_p выделяет в графе $G_l(Q_l, X_l)$ задачи Z_l наиболее длинную нить H_1 , для которой установлен момент времени T_{k+1}^1 ее исполнения и определяет необходимый момент времени начала исполнения подзадачи A_1 , приписанной первой вершине q_1^1 нити H_1

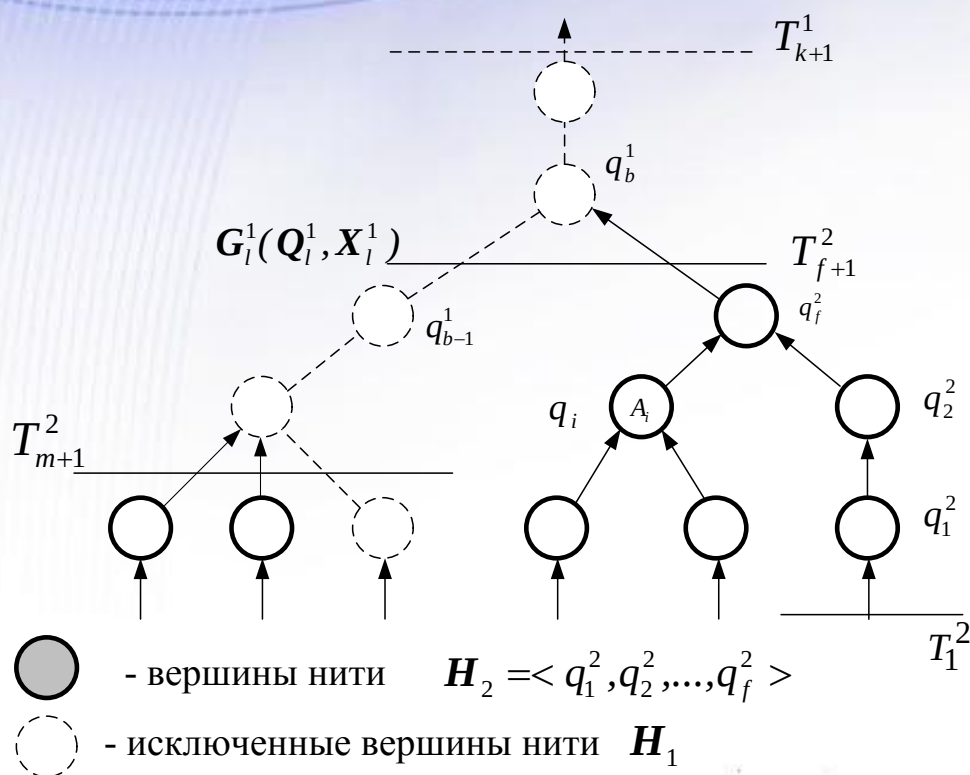
$$T_1^1 = T_{k+1}^1 - \left(\sum_{j=1}^k \frac{V_1^1}{S_1^p} + t_{\pi} \right).$$

Если $T_1^1 < T_{\text{тек}}$, то нить H_1 не может быть решена к установленному моменту времени T_{k+1}^1 . Задача Z_l снимается с решения.



Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 1



Агент R_p принимает на себя выполнение нити H_1 . При этом:

1) записывает свой идентификатор в список членов сообщества R_l ;

2) осуществляет модификацию графа $G_l(Q_l, X_l)$ задания Z_l на ДО, а именно:

– исключает вершины нити H_1 из графа $G_l(Q_l, X_l)$, т.е.

$$G_l^1(Q_l^1, X_l^1) = G_l(Q_l, X_l) / H_1;$$

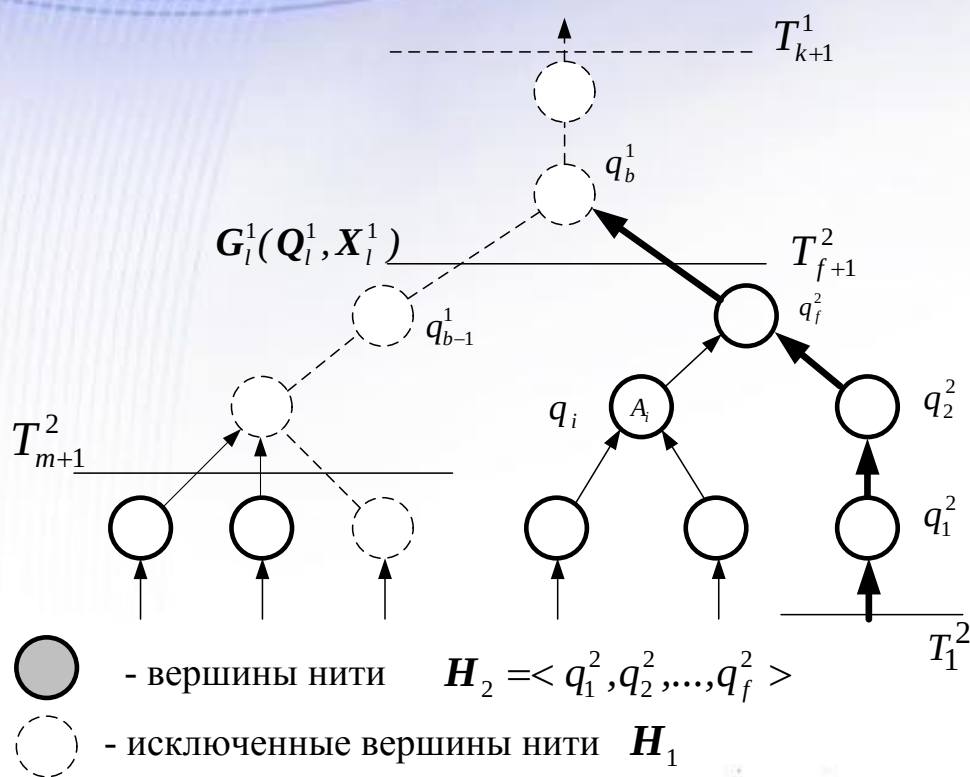
– вершинам графа $G_l^1(Q_l^1, X_l^1)$ инцидентным вершинам нити H_1 приписывается требуемое время их исполнения

$$T_b^1 = T_{k+1}^1 - \left(\sum_{i=b}^k \frac{V_i^1}{S_i^p} + t_{\pi} \right),$$

а также идентификатор ресурса R_p , которому необходимо передать результаты.

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 1



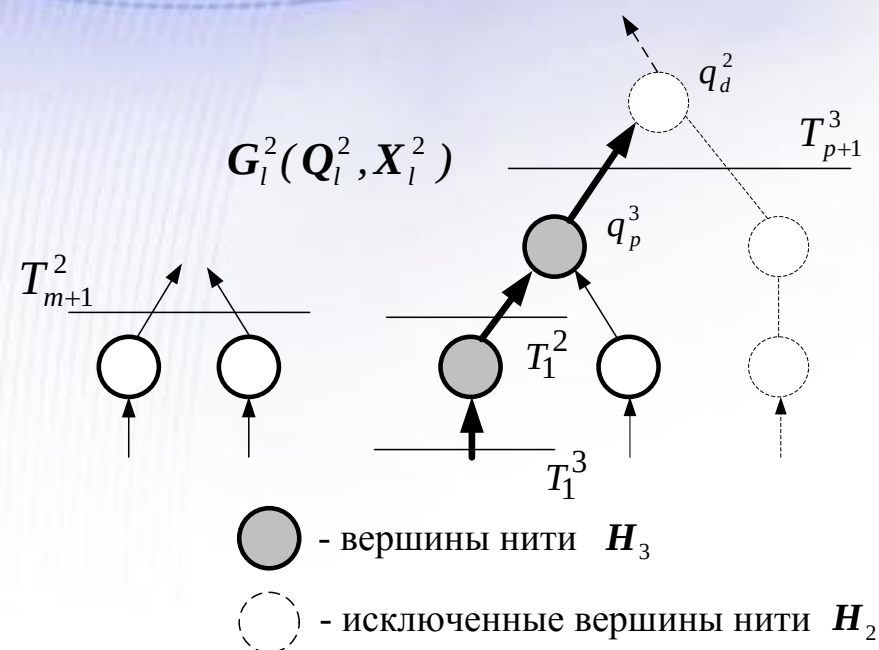
Очередной агент R_c , обнаруживший на ДО задачу Z_l , оценивает экономическую целесообразность своего участия в решении задачи Z_l и, если она его удовлетворяет, то выделяет в графе задачи $G_l^1(Q_l^1, X_l^1)$ наиболее длинную нить H_2 , для которой установлен момент времени T_{f+1}^2 ее исполнения и определяет требуемый момент времени начала исполнения подзадачи A_1 , приписанной первой вершине нити H_2 как

$$T_1^2 = T_{f+1}^2 - \left(\sum_{i=1}^f \frac{V_i^2}{S_i^c} + t_{\pi} \right).$$

Если $T_1^2 < T_{\text{тек}}$, то задача не может быть решена к установленному времени, и она снимается с решения.

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 1



Агент R_c принимает на себя исполнение нити H_2 . При этом он:

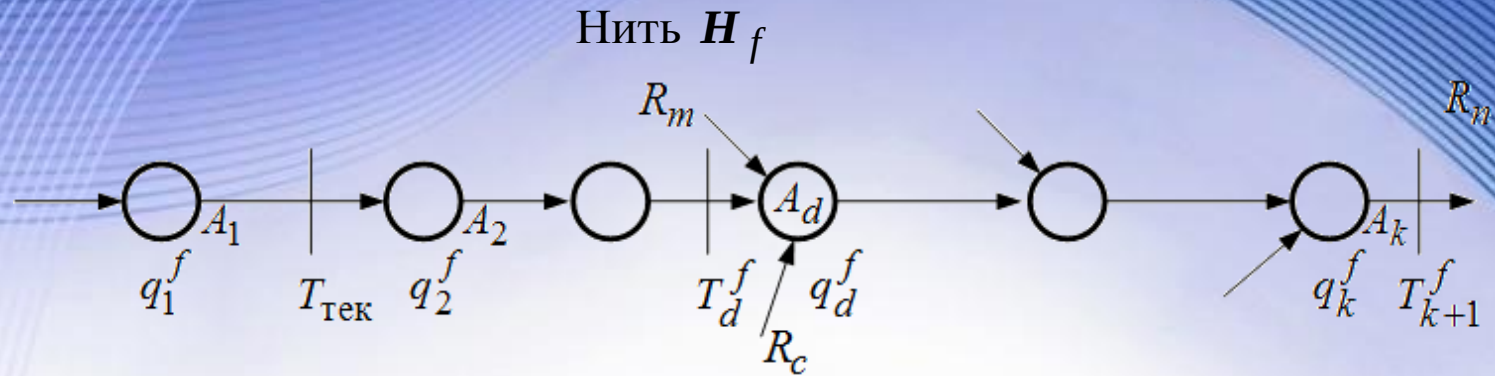
- 1) записывает свой идентификатор в список членов сообщества R_l ;
- 2) модифицирует граф $G_l^1(Q_l^1, X_l^1)$ на ДО:
 - исключает вершины нити H_2 из графа, т.е. $G_l^2(Q_l^2, X_l^2) = G_l^1(Q_l^1, X_l^1) / H_2$;
 - приписывает вершинам графа $G_l^2(Q_l^2, X_l^2)$, инцидентным вершинам нити H_2 требуемое время из исполнения

$$T_{p+1}^3 = T_{f+1}^2 - \left(\sum_{i=d}^f \frac{V_i^2}{S_i^c} + t_{\pi} \right)$$

и идентификатор ресурса R_c .

Процесс распределения продолжается до тех пор пока $G_l^j(Q_l^j, X_l^j) \neq 0$.

Контроль агентом R_p выполнения временного графика исполнения нити H_f



Если к моменту начала выполнения агентом R_p очередной подзадачи A_d , приписанной вершине $q_d^f \in H_f$, получены еще не все исходные данные, необходимые для ее выполнения, то агент R_p переходит в режим ожидания, который длится до тех пор, пока $T_d^f \geq T_{\text{тек}}$, где

$$T_d^f = T_{k+1}^f - \left(\sum_{i=d}^k \frac{V_i^f}{S_i^p} + t_{\text{п}} \right).$$

Если $T_d^f < T_{\text{тек}}$, то это означает, что произошел сбой временного графика выполнения нити H_f .

Если ресурс R_p успешно завершил выполнение нити H_f , то агент R_p исключается из списка членов сообщества R_l и ему начисляются премиальные баллы $O_p = O \cdot V^p / V$, где V^p – суммарная трудоемкость подзадач нити H_p ; V – суммарная трудоемкость всей задачи Z_l .

Алгоритм работы агента. Вариант 1

1. Агент свободного ресурса R_p опрашивает ДО.
2. При обнаружении на ДО задачи Z_l , агент R_p считывает ее дескриптор и анализирует граф задачи $G_l^j(Q_l^j, X_l^j)$. Если $G_l^j(Q_l^j, X_l^j) = \emptyset$, то переход к п.1, иначе
3. Агент R_p оценивает экономическую целесообразность своего участия в решении задачи Z_l , сравнивая предлагаемую пользователем относительную стоимость вычислений $D = O/V$ с установленным для него порогом D_p . Если $D < D_p$, то переход к п.1.
4. Агент R_p выделяет в графе $G_l^j(Q_l^j, X_l^j)$ наиболее длинную нить $H_j = \langle q_1^j, q_2^j, \dots, q_k^j \rangle$, конечной вершине q_l^j которой приписано требуемое время ее исполнения T_{k+1}^j (если $H_j = H_1$, то $T_{k+1}^j = T_{max}^l$), и определяет допустимый момент времени, когда необходимо начать ее выполнение как
$$T_1^j = T_{f+1}^j - \left(\sum_{i=1}^k \frac{V_i^j}{S_i^p} + t_n \right).$$
5. Если $T_1^j < T_{тек}$, где $T_{тек}$ – текущий момент времени, то переход к п.14, иначе

Алгоритм работы агента. Вариант 1

6. Агент R_p принимает на себя выполнение нити H_j , для чего модифицирует дескриптор задания Z_l на ДО: записывает в список участников сообщества свой идентификатор; модифицирует граф $G_l^j(Q_l^j, X_l^j)$ задания Z_l путем исключения из него вершин нити H_j , т.е. $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1}) = G_l^j(Q_l^j, X_l^j) / H_j$; приписывает вершинам q_b^{j+1} модифицированного графа $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1})$, инцидентным вершинам q_d^j нити H_j , требуемое время исполнения, определяемое как

$$T_{b+1}^{j+1} = T_{k+1}^j - \left(\sum_{i=d}^k \frac{V_i^j}{S_i^p} + t_{\pi} \right),$$

а также идентификатор агента R_p , которому необходимо передать результаты исполнения вершины q_b^{i+1} .

7. Агент R_p переходит к исполнению последовательности подзадачи, приписанных вершинам нити $H_j = \langle q_1^j, q_2^j, \dots, q_k^j \rangle$; $d = 1$.

8. Если $T_d^j < T_{\text{тек}}$, где $T_d^j = T_{k+1}^j - \left(\sum_{i=d}^k \frac{V_i^j}{S_i^p} + t_{\pi} \right)$ – требуемое время начала выполнения операций A_d , приписанной вершине $q_d^j \in H_j$, то перейти к п.14, иначе

Алгоритм работы агента. Вариант 1

9. Агент R_p проверяет наличие исходных данных, необходимых для решения подзадачи A_d . если исходные данные еще не поступили, то перейти к п.8, иначе

10. Агент R_p решает подзадачу A_d , приписанную вершине $q_d^f \in \mathbf{H}_j$.

11. Если агенту R_p поступило сообщение о прекращении решения задачи Z_l , то переход к п.1, иначе

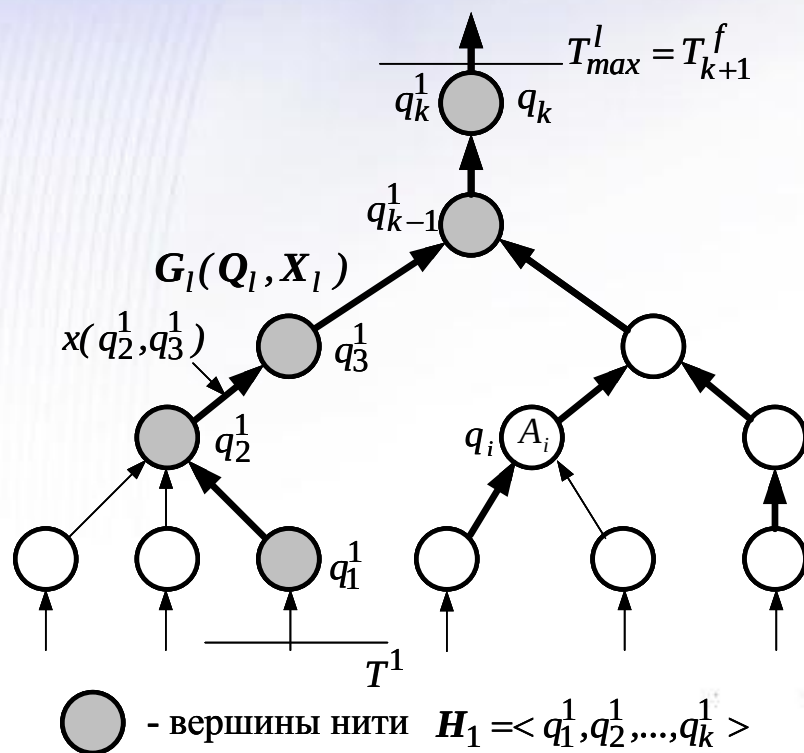
12. $d = d + 1$, если $d \leq k$, то переход к п.8, иначе

13. Агента R_p сообщает на ДО об успешном завершении исполнения нити $\mathbf{H}_j$ задачи Z_l , а результат ее исполнения передается агенту R_c , идентификатор которого приписан последней вершине $q_k^j \in \mathbf{H}_j$, либо потребителю, если $q_k^j = q_k$ – конечная вершина графа $\mathbf{G}_l(\mathbf{Q}_l, \mathbf{X}_l)$, задачи Z_l . Идентификатор агента R_p исключается из списка $\mathbf{R}_l$ членов сообщества по решению задачи Z_l , а ему начисляются премиальные баллы $O_p = O \cdot V^p / V$, переход к п.1.

14. Задача Z_l не может быть решена к требуемому моменту времени T_{max}^l . Дескриптор задачи Z_l удаляется с ДО, пользователю направляется сообщение о невозможности решения его задачи в установленное время, а всем агентам, номера которых записаны в списке участников сообщества по выполнению задачи Z_l , передается сообщение о прекращении ее решения. Переход к п.1.

Вариант 2

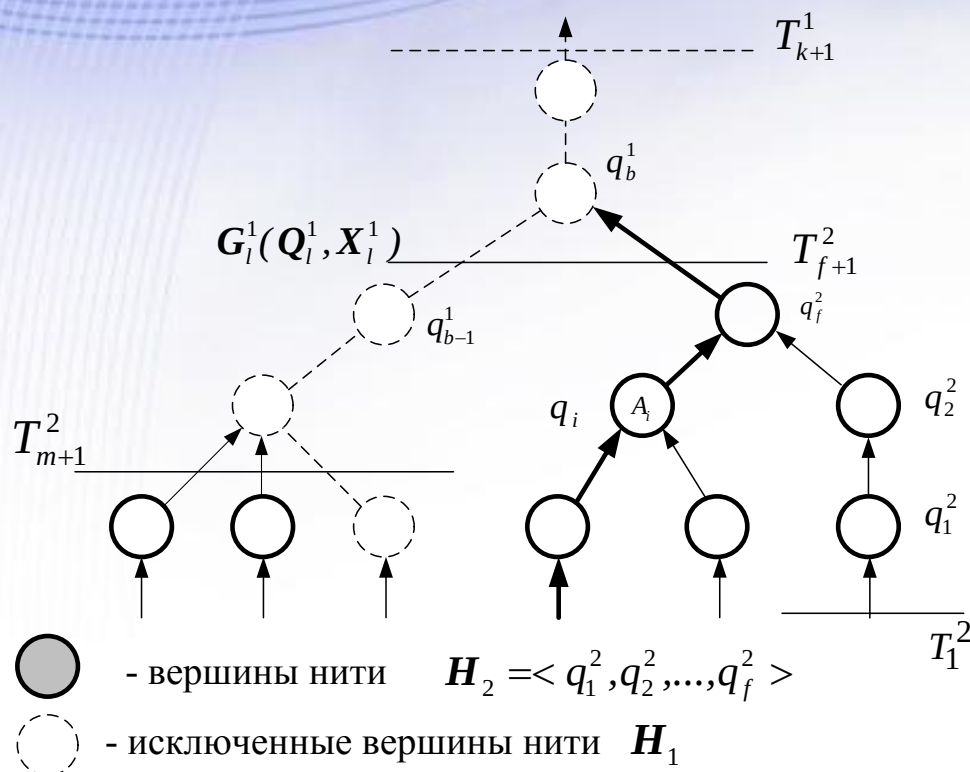
$$S_d^i \neq S_d^j \quad (j = \overline{1, N}, i = 1, 2, \dots, j-1, j+1, \dots, N)$$



3. Если $T_1^1 < T_{\text{тек}}$, то агент R_p выделяет следующую по длине нить и определяет необходимый момент времени начала ее исполнения, и т.д. до тех пор, пока не обнаружится нить, для которой выполняется условие $T_1^m > T_{\text{тек}}$.

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 2



Если агент R_p нашел нить H_1 , которую он может выполнить к требуемому моменту времени, то он входит в состав сообщества R_l и осуществляет модификацию графа $G_l(Q_l, X_l)$ задачи Z_l на ДО:

– исключает вершины нити H_1 из графа $G_l(Q_l, X_l)$, т.е.

$$G_l^1(Q_l^1, X_l^1) = G_l(Q_l, X_l) / H_1;$$

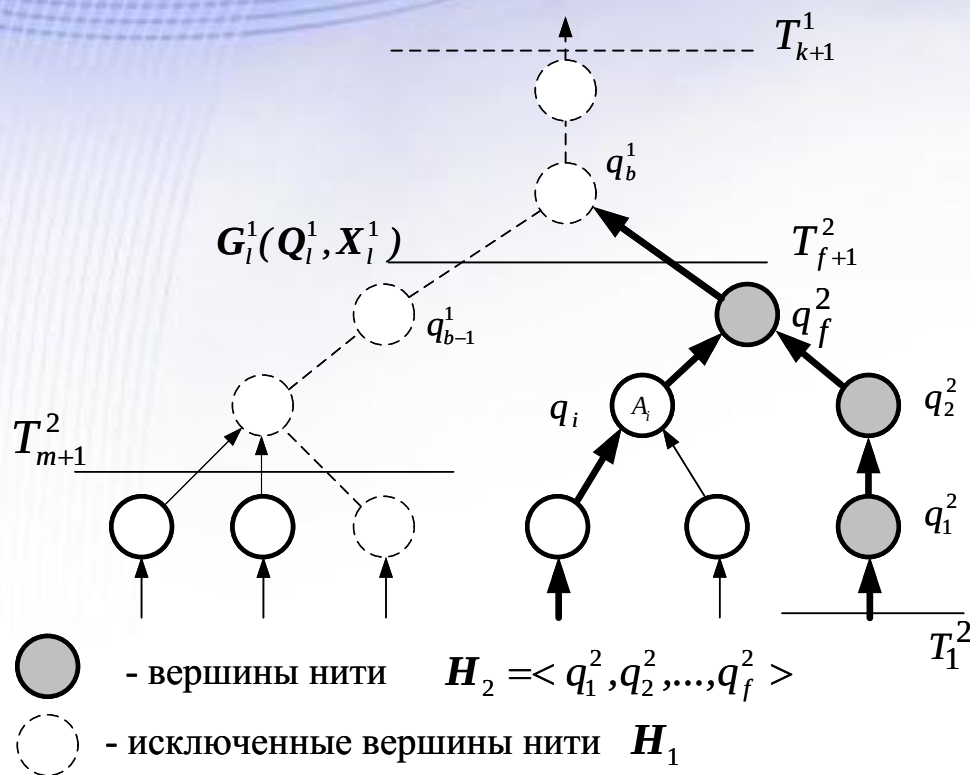
– приписывает вершинам графа $G_l^1(Q_l^1, X_l^1)$, инцидентным вершинам нити H_1 требуемое время их исполнения

$$T_{f+1}^2 = T_{k+1}^1 - \left(\sum_{i=b}^k \frac{V_i^1}{S_i^p} + t_{\pi} \right)$$

и идентификатор ресурса R_p , куда необходимо передать результаты.

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 2



Следующий агент R_c обнаруживший задачу Z_l на ДО выделяет наиболее длинную нить H_2 в модифицированном графе $G_l^1(Q_l^1, X_l^1)$ и определяет необходимое время начала ее выполнения, как

$$T_1^2 = T_{f+1}^2 - \left(\sum_{i=1}^f \frac{V_i^2}{S_i^p} + t_{\pi} \right).$$

Если $T_1^2 < T_{\text{тек}}$, то агент R_c выделяет следующую по длине нить и т.д. до тех пор, пока не обнаружит нить, которую он может исполнить к требуемому моменту времени.

Алгоритм работы агента. Вариант 2

1. Агент R_p свободного ресурса опрашивает ДО.
2. При обнаружении на ДО задачи Z_l , агент R_p считывает ее дескриптор и анализирует граф $G_l^j(Q_l^j, X_l^j)$. Если $G_l^j(Q_l^j, X_l^j) = \emptyset$, то переход к п.1, иначе
3. Если $T_{\text{тек}} \geq T_{\text{max}}^l$, где $T_{\text{тек}}$ – текущий момент времени, то переход к п.19, иначе
4. Агент R_p оценивает экономическую целесообразность своего участия в решении задачи Z_l , сравнивая предлагаемую пользователем относительную стоимость вычислений $D = O/V$ с установленным для него порогом D_p . Если $D < D_p$, то переход к п.1.
5. $i = 1$; $G_i(Q_i, X_i) = G_l^j(Q_l^j, X_l^j)$
6. Агент R_p выделяет в графе $G_i(Q_i, X_i)$ наиболее длинную нить $H_i = \langle q_1^i, q_2^i, \dots, q_k^i \rangle$, для конечной вершины q_l^i которой определено требуемое время исполнения T_{k+1}^i (если $H_i = H_1$, то $T_{k+1}^i = T_{\text{max}}^l$), и определяет момент времени, когда он должен приступить к ее выполнению как
$$T_1^i = T_{k+1}^i - \left(\sum_{m=1}^k \frac{V_m^i}{S_m^p} + t_{\text{п}} \right).$$
7. Если $T_1^i \geq T_{\text{тек}}$, где $T_{\text{тек}}$ – текущий момент времени, то переход к п.11, иначе

Алгоритм работы агента. Вариант 2

8. Нить H_i исключается из графа $G_i(Q_i, X_i)$, т.е.

$$G_{i+1}(Q_{i+1}, X_{i+1}) = G_i(Q_i, X_i) / H_i.$$

9. Если $G_{i+1}(Q_{i+1}, X_{i+1}) = \emptyset$, то перейти к п.1, иначе

10. $i = i + 1$, перейти к п.5.

11. Агент R_p принимает на себя выполнение нити H_i , для чего модифицирует дескриптор задачи Z_l : записывает в список участников сообщества по решению задачи Z_l свой идентификатор; модифицирует граф задачи Z_l путем исключения из него вершин нити H_i , т.е. $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1}) = G_l^j(Q_l^j, X_l^j) / H_i$; приписывает вершинам q_b^{j+1} модифицированного графа $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1})$, инцидентным вершинам q_d^i нити H_i , требуемое время их исполнения, определяемое как:

$$T_{b+1}^{j1} = T_{k+1}^j - \left(\sum_{m=d}^k \frac{V_m^j}{S_m^p} + t_n \right),$$

а также идентификатор агента R_p , которому необходимо передать результаты исполнения вершины q_b^{j+1} .

12. Агент R_p переходит к исполнению последовательности подзадач, приписанных вершинам нити $H_j = \langle q_1^j, q_2^j, \dots, q_k^j \rangle$; $d = 1$.

Алгоритм работы агента. Вариант 2

13. Если $T_d^j < T_{\text{тек}}$, где $T_d^j = T_{k+1}^j - \left(\sum_{i=d}^k t_p(A_i) + t_{\Pi} \right)$ – требуемое время начала

выполнения подзадачи A_d , приписанной вершине $q_d^j \in \mathbf{H}_j$, то перейти к п.19, иначе.

14. Агент R_p проверяет наличие исходных данных, необходимых для выполнения подзадачи A_d . Если исходные данные еще не поступили, то перейти к п.13, иначе

15. Агент R_p решает подзадачу A_d , приписанную вершине $q_d^f \in \mathbf{H}_j$.

16. Если агенту R_p поступило сообщение о прекращении решения задачи Z_l , то переход к п.1, иначе

17. $d = d + 1$, если $d \leq k$, то переход к п.13, иначе

18. После выполнения всех операций нити $\mathbf{H}_i$ идентификатор агента R_p исключается из списка членов сообщества по выполнению задачи Z_l в ее дескрипторе на ДО, а результат выполнения последней операции нити $\mathbf{H}_i$ передается агенту R_c , идентификатор которого приписан конечной вершине q_k^i нити $\mathbf{H}_i$, или пользователю, если последняя вершина выполненной нити $\mathbf{H}_i$ – это конечная вершина графа $\mathbf{G}_l(\mathbf{Q}_l, \mathbf{X}_l)$, т.е. $q_k^i = q_k$. В последнем случае дескриптор задания Z_l удаляется с ДО. Агенту начисляются премиальные баллы $O_p = O \cdot V^p / V$. Переход к п.1.

Алгоритм работы агента. Вариант 2

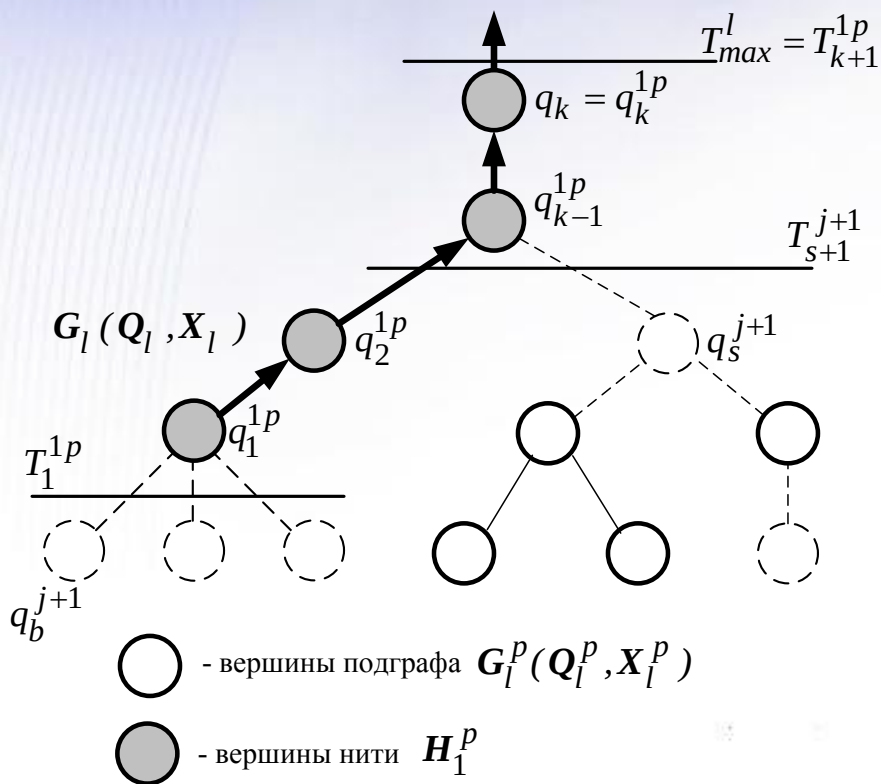
19. Задача Z_l не может быть решена к требуемому моменту времени T_{max}^l . Дескриптор задачи Z_l удаляется с ДО, пользователю направляется сообщение о невозможности выполнения его задания, а всем агентам, номера которых записаны в списке участников сообщества по решению задачи Z_l , сообщается о прекращении ее решения. Переход к п.1.

Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 3

$$A_j \neq A_i, A_j \cap A_i \neq \emptyset (j = \overline{1, N}, i = 1, 2, \dots, j-1, j+1, \dots, N)$$

$$S_d^i = S_d^j \quad A_s \in A_j \cap A_i \quad (j = \overline{1, N}, i = \overline{1, N})$$



Агент R_p выделяет в графе $G_l(Q_l, X_l)$ подграф $G_l^p(Q_l^p, X_l^p)$ вершинам которого приписаны подзадачи подмножества $A_p \subseteq A$, и находит в нем наиболее длинную нить H_1^p , для которой установлено требуемое время исполнения T_{k+1}^{1p} и определяет необходимое время начала ее исполнения как

$$T_1^{1p} = T_{k+1}^{1p} - \left(\sum_{i=1}^k \frac{V_i^{1p}}{S_i^p} + t_{\pi} \right).$$

Если $T_1^{1p} < T_{\text{тек}}$, то задание невыполнимо.

Иначе агент R_p принимает на себя исполнение нити H_1^p и модифицирует граф задания $G_l(Q_l, X_l)$ на ДО.

Алгоритм работы агента. Вариант 3

1. Агент R_p свободного объекта опрашивает ДО.
2. При обнаружении на ДО задачи Z_i агент R_p считывает ее дескриптор и анализирует граф задачи $G_l^j(Q_l^j, X_l^j)$. Если $G_l^j(Q_l^j, X_l^j) = \emptyset$, то переход к п.1, иначе
3. Агент R_p оценивает экономическую целесообразность своего участия в решении задачи Z_l , сравнивая предлагаемую пользователем относительную стоимость вычислений $D = O/V$ с установленным для него порогом D_p . Если $D < D_p$, то переход к п.1.
4. В графе $G_l^j(Q_l^j, X_l^j)$ выделяется подграф $G_l^{jp}(Q_l^{jp}, X_l^{jp})$, вершинам которого приписаны подзадачи множества A_p , выполняемые ресурсом R_p .
5. Если $G_l^{jp}(Q_l^{jp}, X_l^{jp}) = \emptyset$ или ни одной из вершин графа $G_l^{jp}(Q_l^{jp}, X_l^{jp})$ не приписано требуемое время T_{k+1}^j ее исполнения, то перейти к п.1, иначе
6. Агент R_p выделяет в графе $G_l^{jp}(Q_l^{jp}, X_l^{jp})$ наиболее длинную нить $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, конечной вершине которой приписано требуемое время исполнения T_{k+1}^{jp} (в момент размещения задания Z_i на ДО требуемое время $T_{k+1}^{jp} = T_{max}^l$ приписано только конечной вершине q_k графа $G_l(Q_l, X_l)$), и определяет допустимый момент времени, когда необходимо начать ее выполнение как:

Алгоритм работы агента. Вариант 3

$$T_1^{jp} = T_{k+1}^{jp} - \left(\sum_{i=1}^k \frac{V_i^{jp}}{S_i^p} + t_{\pi} \right).$$

7. Если $T_1^{jp} < T_{\text{тек}}$, где $T_{\text{тек}}$ – текущий момент времени, то переход к п.16, иначе

8. Агент R_p принимает на себя исполнение нити $\mathbf{H}_j^p$ для чего модифицирует дескриптор задачи Z_l на ДО: в список участников сообщества записывается его идентификатор p ; из графа $\mathbf{G}_l^j(\mathbf{Q}_l^j, \mathbf{X}_l^j)$ исключаются вершины нити $\mathbf{H}_j^p$, в результате чего формируется новый граф $\mathbf{G}_l^{j+1}(\mathbf{Q}_l^{j+1}, \mathbf{X}_l^{j+1}) = \mathbf{G}_l^j(\mathbf{Q}_l^j, \mathbf{X}_l^j) / \mathbf{H}_j^p$; вершинам q_b^{j+1} модифицированного графа $\mathbf{G}_l^{j+1}(\mathbf{Q}_l^{j+1}, \mathbf{X}_l^{j+1})$, инцидентным вершинам q_d^{jp} нити $\mathbf{H}_j^p$, приписывается требуемое время их исполнения, определяемое как

$$T_{b+1}^{j+1,p} = T_{k+1}^{jp} - \left(\sum_{i=d}^k \frac{V_i^{jp}}{S_i^p} + t_{\pi} \right),$$

а также идентификатор ресурса R_p , которому необходимо передать результаты решения подзадачи, приписанной вершине q_b^{j+1} .

Алгоритм работы агента. Вариант 3

9. Агент R_p приступает к исполнению подзадачи, приписанных вершинам нити $\mathbf{H}_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, $d = 1$.

10. Если $T_d^{jp} < T_{\text{тек}}$, где $T_d^{jp} = T_{k+1}^{jp} - \left(\sum_{i=d}^k \frac{V_i^{jp}}{S_i^p} + t_{\Pi} \right)$ $T_d^{jp} = T_{k+1}^j - \left(\sum_{i=d}^k t_p(A_i) + t_{\Pi} \right)$ –

требуемое время начала решения подзадачи A_d , приписанной вершине $q_d^{jp} \in \mathbf{H}_j^p$, то перейти к п.16, иначе

11. Если имеются не все исходные данные, необходимые для решения подзадачи A_d , приписанной вершине $q_d^f \in \mathbf{H}_j^p$, то агент R_p переходит в режим ожидания поступления данных от агентов, выполняющих сложные подзадачи, перейти к п.9, иначе

12. Агент R_p решает подзадачу A_d , приписанную вершине $q_d^{jp} \in \mathbf{H}_j^p$.

13. Если агенту R_p поступило сообщение о прекращении решения задачи Z_l , то переход к п.1, иначе

14. $d = d + 1$, если $d \leq k$, то переход к п.9, иначе

Алгоритм работы агента. Вариант 3

15. Агента R_p сообщает на ДО об успешном завершении исполнения нити $\mathbf{H}_j^p$ задачи Z_l , а результат ее исполнения передается агенту R_c , идентификатор которого приписан последней вершине $q_k^{jp} \in \mathbf{H}_j^p$, либо пользователю, если $q_k^{jp} = q_k$ – конечная вершина графа $\mathbf{G}_l(\mathbf{Q}_l, \mathbf{X}_l)$, задачи Z_l . Идентификатор агента R_p исключается из списка $\mathbf{R}_l$ членов сообщества по решению задачи Z_l . Агенту R_p начисляются премиальные баллы $O_p = O \cdot V^p / V$. Переход к п.1.

16. Задача Z не может быть решена к требуемому моменту времени T_{max}^l . Дескриптор задачи Z_l удаляется с ДО, пользователю направляется сообщение о невозможности решения его задачи, а всем агентам, идентификаторы которых записаны в списке участников сообщества по решению задачи Z_l , передается сообщение о прекращении ее решения. Переход к п.1

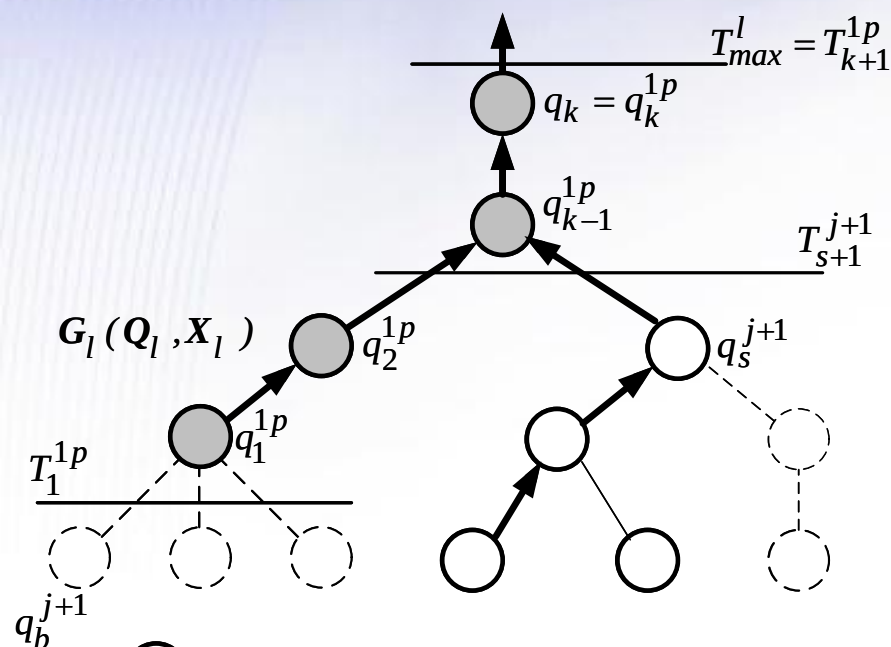
Принципы мультиагентного диспетчирования ресурсов в ОВС

Вариант 4

$$A_j \neq A_i \quad (j = \overline{1, N}, i = 1, 2, \dots, j-1, j+1, \dots, N)$$

$$S_d^i \neq S_d^j \quad A_s \in A_j \cap A_i$$

$$(i = \overline{1, N}, j = 1, 2, \dots, i-1, i+1, \dots, N)$$



○ - вершины подграфа $G_l^p(Q_l^p, X_l^p)$

● - вершины нити H_1^p

○ - вершины графа $G_l(Q_l, X_l)$ которым принесены подзадачи, не входящие в множество A_p

Агент R_p выделяет в графе $G_l(Q_l, X_l)$ подграф $G_l^p(Q_l^p, X_l^p)$, вершинам которого приписаны подзадачи подмножества $A_p \subseteq A$, и находит в нем нить H_1 , конечной вершине которой приписано требуемое время исполнения T_{k+1}^{1p} и определяет необходимое время начала ее исполнения как

$$T_1^{1p} = T_{k+1}^{1p} - \left(\sum_{i=1}^k \frac{V_i^{1p}}{S_i^p} + t_{\pi} \right).$$

Если $T_1^{1p} < T_{\text{тек}}$, то агент R_p выделяет следующую по длине нить и т.д. до тех пор, пока не обнаружит нить, которую он может выполнить к заданному моменту времени.

Алгоритм работы агента. Вариант 4

1. Агент R_p свободного объекта опрашивает ДО.
2. При обнаружении на ДО задачи Z_i агент R_p считывает ее дескриптор и анализирует граф задачи $G_l^j(Q_l^j, X_l^j)$. Если $G_l^j(Q_l^j, X_l^j) = \emptyset$, то переход к п.1, иначе
3. Если $T_{\text{тек}} \geq T_l^{\text{max}}$, где $T_{\text{тек}}$ – текущий момент времени, то переход к п.21, иначе
4. Агент R_p оценивает экономическую целесообразность своего участия в решении задачи Z_l , сравнивая предлагаемую пользователем относительную стоимость вычислений $D = O/V$ с установленным для него порогом D_p . Если $D < D_p$, то переход к п.1.
5. В графе $G_l^j(Q_l^j, X_l^j)$ выделяется подграф $G_l^{jp}(Q_l^{jp}, X_l^{jp})$, вершинам которого приписаны подзадачи множества A_p , выполняемые ресурсом R_p .
6. Если $G_l^{jp}(Q_l^{jp}, X_l^{jp}) = \emptyset$ или ни одной из вершин графа $G_l^{jp}(Q_l^{jp}, X_l^{jp})$ не приписано требуемое время T_{k+1}^j ее исполнения, то перейти к п.1, иначе
7. $i = 1$; $G_i(Q_i, X_i) = G_l^{jp}(Q_l^{jp}, X_l^{jp})$.

Алгоритм работы агента. Вариант 4

8. Агент R_p выделяет в графе $G_i(Q_i, X_i)$ наиболее длинную нить $H_i^p = \langle q_1^{ip}, q_2^{ip}, \dots, q_k^{ip} \rangle$, для конечной вершины q_k^{ip} которой определено требуемое время исполнения T_{k+1}^{ip} (если $H_i = H_1$, то $T_{k+1}^{ip} = T_{max}^l$), а для начальной вершины есть все исходные данные и определяет момент времени, когда он должен приступить к ее выполнению как

$$T_1^{ip} = T_{k+1}^{ip} - \left(\sum_{m=1}^k \frac{V_m^{ip}}{S_m^p} + t_{\pi} \right).$$

9. Если $T_1^{ip} \geq T_{\text{тек}}$, где $T_{\text{тек}}$ – текущий момент времени, то переход к п.12, иначе

10. Нить H_i^p исключается из графа $G_i(Q_i, X_i)$, т.е.

$$G_{i+1}(Q_{i+1}, X_{i+1}) = G_i(Q_i, X_i) / H_i^p.$$

11. Если $G_{i+1}(Q_{i+1}, X_{i+1}) = \emptyset$, то перейти к п.1, иначе

12. $i = i + 1$, перейти к п.7.

Алгоритм работы агента. Вариант 4

13. Агент R_p принимает на себя исполнение нити $\mathbf{H}_i^p$ для чего модифицирует дескриптор задачи Z_i на ДО: в список участников сообщества записывается его идентификатор p ; из графа $G_l^j(Q_l^j, X_l^j)$ исключаются вершины нити $\mathbf{H}_i^p$, в результате чего формируется новый граф $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1}) = G_l^j(Q_l^j, X_l^j) / \mathbf{H}_i^p$; вершинам q_b^{j+1} модифицированного графа $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1})$, инцидентным вершинам q_d^{ip} нити $\mathbf{H}_i^p$, приписывается требуемое время их исполнения, определяемое как

$$T_{b+1}^{j+1,p} = T_{k+1}^{jp} - \left(\sum_{m=d}^k \frac{V_m^{ip}}{S_m^p} + t_{\pi} \right),$$

а также идентификатор агента R_p , которому необходимо передать результаты исполнения подзадачи, приписанной вершине q_b^{j+1} .

14. Агент R_p приступает к исполнению операций, приписанных вершинам нити $\mathbf{H}_i^p = \langle q_1^{ip}, q_2^{ip}, \dots, q_k^{ip} \rangle$, $d=1$.

15. Если $T_d^{ip} < T_{\text{тек}}$, где $T_d^{ip} = T_{k+1}^{ip} - \left(\sum_{m=d}^k \frac{V_m^{ip}}{S_m^p} + t_{\pi} \right)$ – требуемое время начала решения задачи A_d , приписанной вершине $q_d^{ip} \in \mathbf{H}_i^p$, то перейти к п.21, иначе

Алгоритм работы агента. Вариант 4

16. Если имеются не все исходные данные, необходимые для решения подзадачи A_d , приписанной вершине $q_d^f \in \mathbf{H}_j$, то агент R_p переходит в режим ожидания поступления данных от агентов, выполняющих сложные подзадачи, переход к п.15, иначе

17. Агент R_p решает подзадачу A_d , приписанную вершине $q_d^{jp} \in \mathbf{H}_j^p$.

18. Если агенту R_p поступило сообщение о прекращении выполнения задачи Z_l , то переход к п.1, иначе

19. $d = d + 1$, если $d \leq k$, то переход к п.15, иначе

20. Агент R_p сообщает на ДО об успешном завершении исполнения нити $\mathbf{H}_j^p$ задачи Z_l , а результат ее исполнения передается агенту R_c , идентификатор которого приписан последней вершине $q_k^{jp} \in \mathbf{H}_j^p$, либо пользователю, если $q_k^{jp} = q_k$ – конечная вершина графа $\mathbf{G}_l(\mathbf{Q}_l, \mathbf{X}_l)$, задачи Z_l . Идентификатор агента R_p исключается из списка $\mathbf{R}_l$ членов сообщества по решению задачи Z_l . Агенту R_p начисляются премиальные баллы $O_p = O \cdot V^p / V$. Переход к п.1.

21. Задача Z не может быть решена к требуемому моменту времени T_{max}^l . Дескриптор задачи Z_l удаляется с ДО, пользователю направляется сообщение о невозможности решения его задачи, а всем агентам, номера которых записаны в списке участников сообщества по выполнению задачи Z_l , передается сообщение о прекращении ее решения. Переход к п.1.

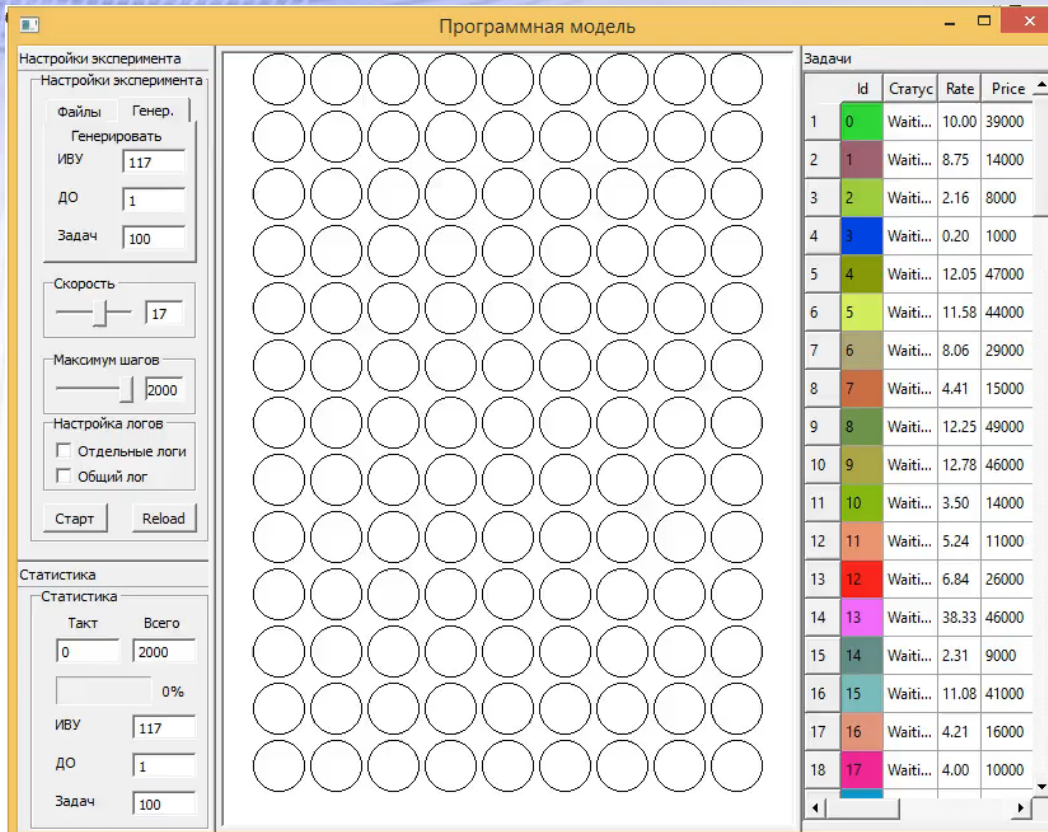
Результаты экспериментальных исследований

Варьируемые параметры модели

- количество гетерогенных вычислительных ресурсов в ОВС (до 1000 шт.);
- количество типов подзадач, выполняемых различными ресурсами ОВС (до 20);
- производительность ресурсов при решении подзадач различных типов;
- вычислительная сложность (количество операций) подзадач;
- требуемые моменты времени получения результатов решения пользовательских задач.

Критерии эффективности

- коэффициент полезного действия (КПД) – отношение времени, затраченного ресурсами ОВС на выполнение пользовательских задач, к общему времени их работы в системе;
- коэффициент гарантированности выполнения (КГВ) пользовательской задачи – отношение количества пользовательских задач, выполненных к требуемому моменту времени, к общему количеству задач, направленных пользователями на ДО.



Результаты экспериментов:

КПД $\geq 75\%$ (среднее значение 84%);
КГВ $\geq 91\%$ (среднее значение 97%).

Заключение

Реализация предлагаемого метода мультиагентного диспетчирования ресурсов гетерогенной ОВС позволяет:

- обеспечить квазиоптимальное адаптивное распределение ресурсов ОВС в процессе решения потока пользовательских задач с учетом их реальной производительности на той или иной подзадаче;
- обеспечить высокую полезную загрузку ресурсов ОВС;
- обеспечить возможность неограниченного наращивания (масштабируемости) числа гетерогенных вычислительных ресурсов в ОВС;
- обеспечить повышенную отказоустойчивость ОВС, поскольку в ней отсутствуют узлы, выход из строя которых приводит к катастрофическим последствиям для системы в целом.

Благодарю за внимание!