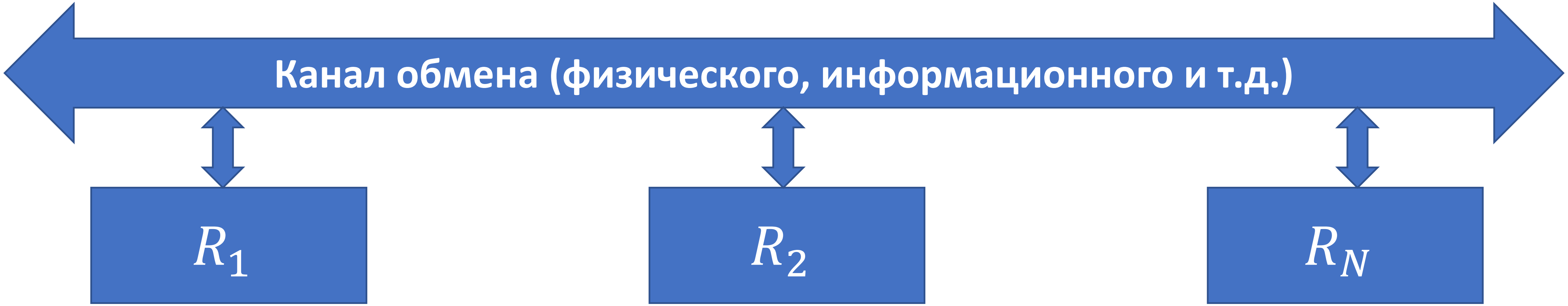
САМООРГАНИЗУЮЩИЕСЯ РАСПРЕДЕЛЕННЫЕ СИСТЕМЫ

Академик РАН
И.А. Каляев

РАСПРЕДЕЛЕННЫЕ СИСТЕМЫ



$\boldsymbol{O_1} = \langle O_1^1, O_2^1, \ldots O_{B_1}^1 \rangle$ $\boldsymbol{O_2} = \langle O_1^2, O_2^2, \ldots O_{B_2}^2 \rangle$ $\boldsymbol{O_N} = \langle O_1^N, O_2^N, \ldots O_{B_N}^N \rangle$

Множество операций, выполняемых РС $\boldsymbol{O} = \bigcup_{i=1}^{N} \boldsymbol{O_i}$.

Время, затрачиваемое ресурсами R_i на выполнение операции O_j

$$t_i(O_j) = \frac{v(O_j)}{D_i(O_j)}$$

где $v(O_j)$ – трудоемкость операции O_j;

$D_i(O_j)$ – производительность ресурса R_i при выполнении операции O_j.

КЛАССИФИКАЦИЯ РАСПРЕДЕЛЕННЫХ СИСТЕМ

1 Гомогенные РС первого типа

Все ресурсы $R_1, R_2, \dots, R_N$ выполняют одинаковые множества операций, т.е. $\boldsymbol{O}_i = \boldsymbol{O}_j = \boldsymbol{O}\ \ N, (i = 1, 2, \dots, N, j = 1, 2, \dots, N)$ причем производительность всех ресурсов $R_i\ (i = 1, 2, \dots, N)$ при выполнении одной и той же операции $O_s \in \boldsymbol{O}$ также одинакова, т.е. $D_i(O_s) = D_j(O_s)\ (i = 1, 2, \dots, N, j = 1, 2, \dots, N)$.

2 Гомогенные РС второго типа

Ресурсы $R_1, R_2, \dots, R_N$ выполняют одинаковые множества операций, т.е. $\boldsymbol{O}_i = \boldsymbol{O}_j = \boldsymbol{O}\ (i = 1, 2, \dots, N, j = 1, 2, \dots, N)$, но при этом производительность различных ресурсов $R_i \in \boldsymbol{R}$ и $R_j \in \boldsymbol{R}\ (i \neq j)$ при выполнении одной и той же операции $O_s \in \boldsymbol{O}$ различна, т.е. $D_i(O_s) \neq D_j(O_s)\ \ (i = 1, 2, \dots, N, j = 1, 2, \dots, i-1, i+1, \dots, N)$.

3 Гетерогенные РС первого типа

Множества операций, выполняемых различными ресурсами, различны (т.е. каждый ресурс имеет свою функциональную специализацию) $\boldsymbol{O}_i \neq \boldsymbol{O}_j\ \ (i = 1, 2, \dots, N, j = 1, 2, \dots, i-1, i+1, \dots, N)$, хотя они могут и пересекаться, т.е. $\boldsymbol{O}_i \cap \boldsymbol{O}_j \neq \varnothing$. При этом производительность различных ресурсов $R_i \in \boldsymbol{R}$ и $R_j \in \boldsymbol{R}$ при выполнении одной и той же операции $O_s \in \boldsymbol{O}$ одинаковая, т.е. $D_i(O_s) \neq D_j(O_s)\ (i = 1, 2, \dots, N, j = 1, 2, \dots, N)$.

4 Гетерогенные РС второго типа

Все ресурсы $R_1, R_2, \dots, R_N$ выполняют различные множества операций, т.е. $\boldsymbol{O}_i \neq \boldsymbol{O}_j$ и $\boldsymbol{O}_i \cap \boldsymbol{O}_j \neq \varnothing\ (i = 1, 2, \dots, N, j = 1, 2, \dots, i-1, i+1, \dots, N)$, причем производительность различных ресурсов $R_i \in \boldsymbol{R}$ и $R_j \in \boldsymbol{R}\ (i \neq j)$ при выполнении одной и той же операции $O_s \in\ \boldsymbol{O}_i$ также различна, т.е. $D_i(O_s) \neq D_j(O_s)$.

ЗАДАНИЯ, ВЫПОЛНЯЕМЫЕ РАСПРЕДЕЛЕННОЙ СИСТЕМОЙ

Поток заданий $\boldsymbol{Z} = \langle Z_1, Z_2, \dots, Z_L \rangle$

Ветвь графа задания Z_l

$$\boldsymbol{H_f} = \langle q_1^f, q_2^f, \dots, q_k^f \rangle$$

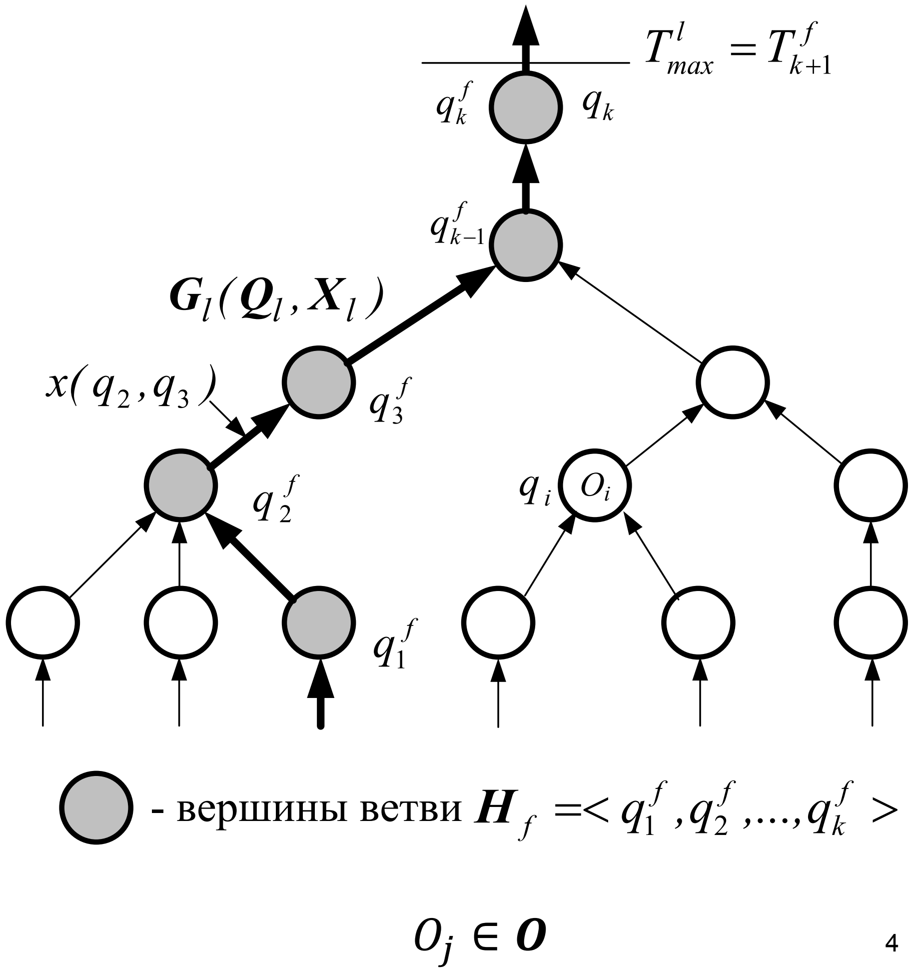
Длина ветви H_f

$$t_f = \sum_{i=1}^{k} \left(t_p(O_i^f) + t_п(O_i^f, O_{i+1}^f) \right)$$

где $t_p(O_i^f) = \frac{v(O_j^f)}{D_i(O_j^f)}$ – время затрачиваемое ресурсом R_p на выполнение операции O_i^f

$t_п(O_i^f, O_{i+1}^f)$ – время затрачиваемое ресурсом R_p на передачу результатов операции O_i^f ресурсу R_c, выполняющему следующую операцию O_{i+1}^f

$$t_п(O_i^f, O_{i+1}^f) = \begin{cases} 0, \text{если } R_p \text{ и } R_c \text{ один и тот же ресурс} \\ t_п, \text{если } R_p \text{ и } R_c \text{ разные ресурсы} \end{cases}$$



ЗАДАЧА САМООРГАНИЗАЦИИ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ

Самоорганизация – это свойство системы без дополнительного внешнего воздействия формировать свою внутреннюю пространственную, временную и функциональную структуры для оптимального выполнения поставленного перед ним задания.

Задача самоорганизации РС состоит в динамическом формировании в ее структуре множества сообществ ресурсов $\boldsymbol{R_l}, \dots, \boldsymbol{R_f}$, обеспечивающих выполнение потока поступающих заданий $Z_l, \dots, Z_f$ с заданным уровнем качества

КРИТЕРИЙ КАЧЕСТВА ВЫПОЛНЕНИЯ ПОТОКА ЗАДАНИЙ В РС

Время выполнения задания Z_l

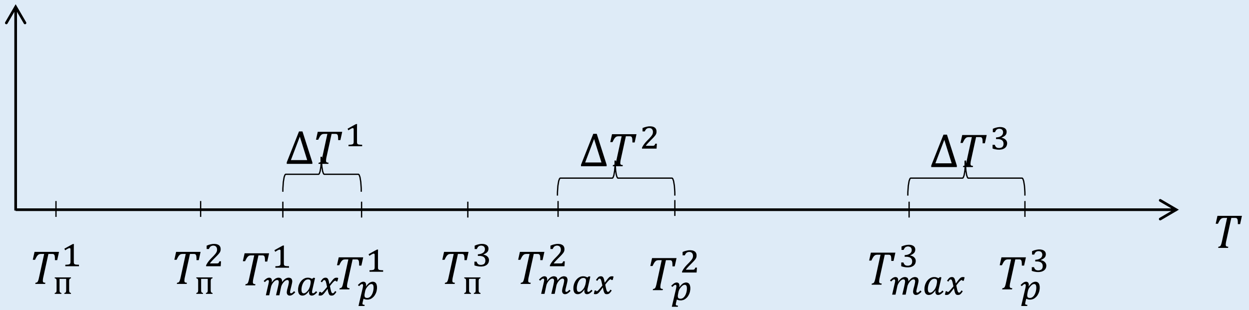
$$t^l = t_p^l + t_B^l$$

где t_p^l – время затрачиваемое на создание сообщества $\boldsymbol{R_l}$

t_B^l – время выполнения задания сообществом $\boldsymbol{R_l}$

Задержка выполнения задания Z_l

$$\Delta T^l = \begin{cases} 0, \text{если } T_\Pi^l + t^l \le T_{max}^l \\ \left(T_\Pi^l + t^l\right) - T_{max}^l, \text{иначе} \end{cases}$$



T_Π^l - момент времени поступления задания Z_l

T_{max}^l - требуемый момент времени выполнения задания Z_l;

T_p^l - реальный момент времени завершения выполнения задания Z_l.

$$\Delta T = \frac{\sum_{l=1}^{L} \Delta T^l}{L} = \frac{\sum_{l=1}^{L} \left(T_p^l - T_{max}^l\right)}{L}$$

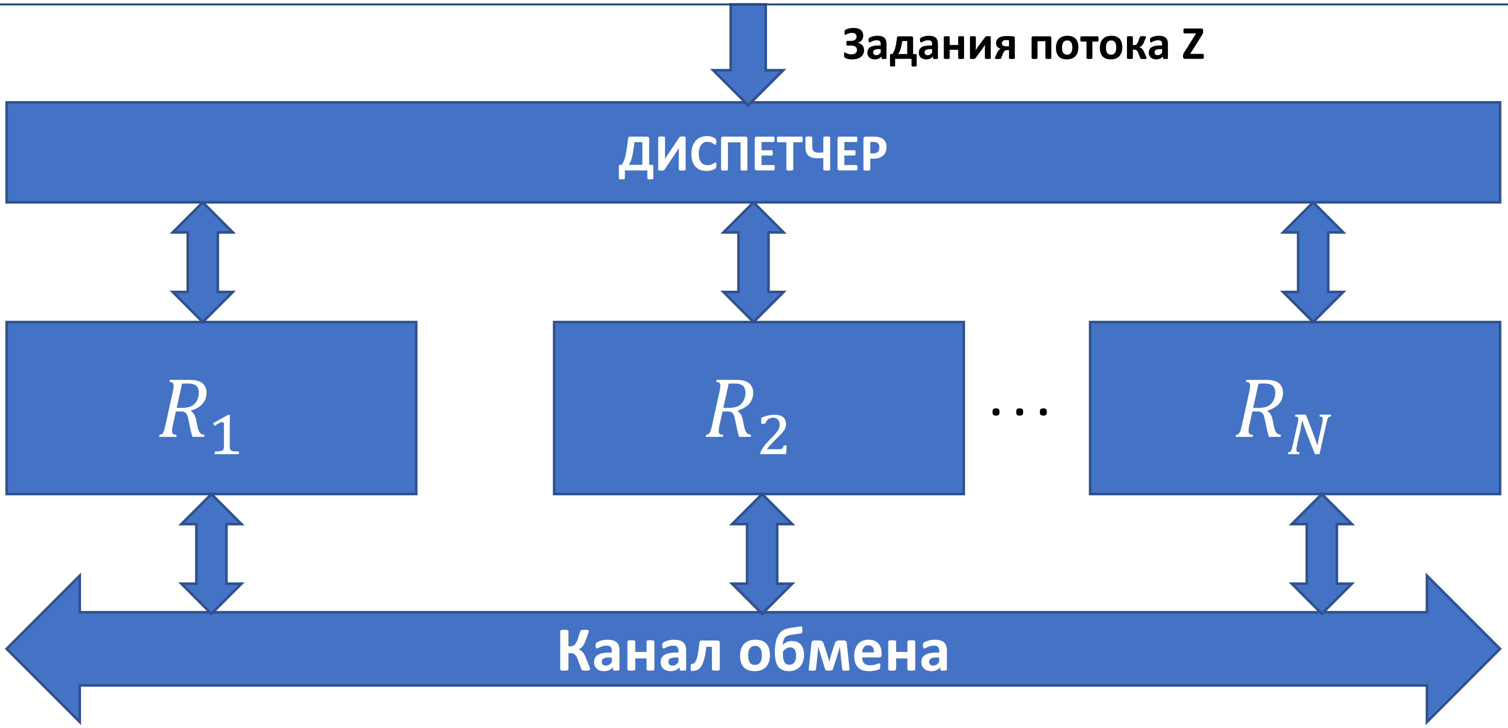
где L – число заданий, выполненных РС.

ЗАДАЧА САМООРГАНИЗАЦИИ РС

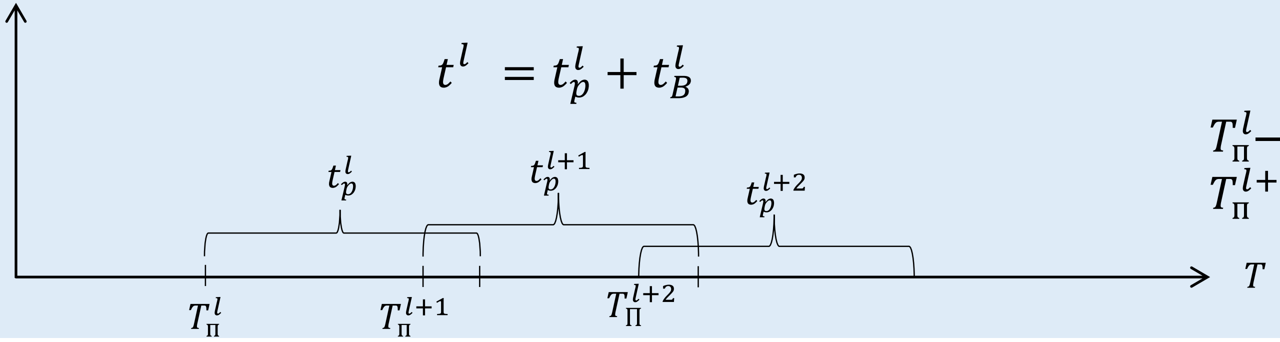
Задача самоорганизации РС состоит в динамическом формировании в ее структуре множества сообществ ресурсов $\boldsymbol{R_l}, \dots, \boldsymbol{R_f}$, каждое из которых решает свою задачу из потока поступающих заданий $Z_l, \dots, Z_f$, обеспечивая при этом минимум средней задержки выполнения всех заданий потока.

Или, иначе говоря, необходимо обеспечить такое распределение множества операций $O_j \subseteq \boldsymbol{O}$ заданий $Z_l \in \boldsymbol{Z}$, поступающих в произвольные моменты времени и описываемых графом $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$, по ресурсам $\boldsymbol{R_l}, \dots, \boldsymbol{R_f}$ которое бы минимизировало среднее время ΔT задержки выполнения всех заданий потока $\boldsymbol{Z} = \langle Z_1, Z_2, \dots, Z_L \rangle$.

СТРУКТУРА САМООРГАНИЗУЮЩЕЙСЯ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ



$$t^l = t_p^l + t_B^l$$



Должно выполняться условие

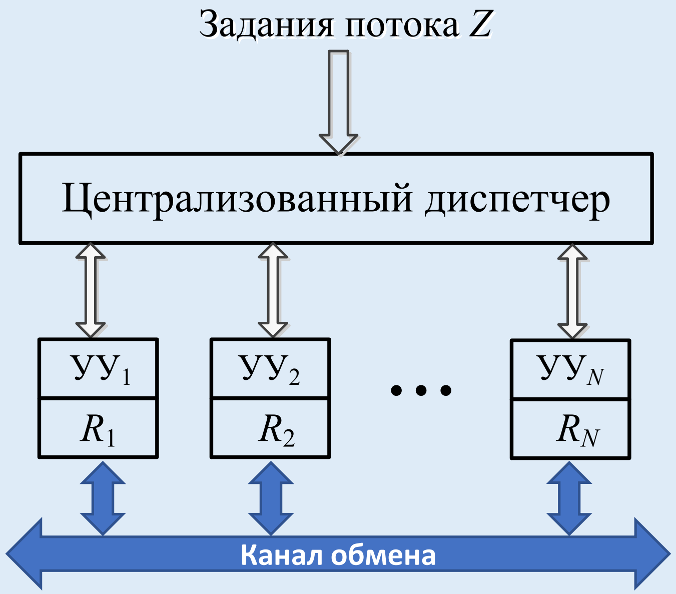
$$t_p^l < T_\text{п}^{l+1} - T_\text{п}^l$$

$T_\text{п}^l$ – момент времени поступления задания Z_l;

$T_\text{п}^{l+1}$ – момент времени поступления задания Z_{l+1}

Способы организации диспетчера СРС

СРС с централизованным диспетчером



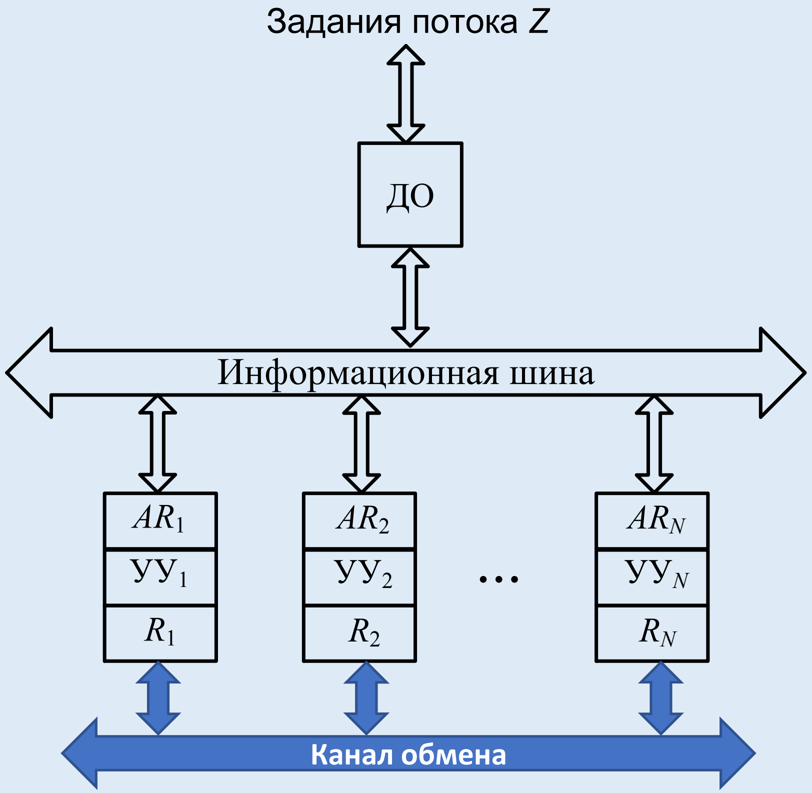
$УУ_i$ (i = 1, 2, …, N) – устройство управления ресурса R_i

$$t_p^l \sim N^M$$

N – число ресурсов R

M – число вершин в графе $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ задания Z_l

СРС с мультиагентным диспетчером



AR_i $(i = 1, 2, \dots, N)$ – агент ресурса R_i

$УУ_i$ $(i = 1, 2, \dots, N)$ – устройство управления ресурса R_i

$$t_p^l \sim N \cdot M$$

БАЗОВАЯ ПРОЦЕДУРА МУЛЬТИАГЕНТНОГО РАСПРЕДЕЛЕНИЯ РЕСУРСОВ

1 Граф $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ очередного задания $Z_l \in \boldsymbol{Z}$ размещается на ДО.

2 Агенты ресурсов $R_1, R_2, \dots, R_N$, входящих в состав СРС, периодически в некоторой последовательности (например в порядке нумерации) опрашивают ДО поисках работы для «своих» ресурсов.

3 Агент AR_1 ресурса R_1 при обнаружения на ДО некоторого задания $Z_l \in \boldsymbol{Z}$ делает попытку войти в состав сообщества $\boldsymbol{R_1}$ по его выполнению. Для этого агент AR_1 выделяет в графе $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ задания $Z_l \in \boldsymbol{Z}$ наиболее длинную ветвь $\boldsymbol{H}_1 =< q_1^1, q_2^1, \dots, q_k^1$, операции которой он может выполнить с помощью «своего» ресурса R_1 с минимальной задержкой относительно требуемого момента времени T_{k+1}^1, приписанного ее конечной вершине q_k^1. Здесь следует отметить, что при размещении задания Z_l на ДО, требуемый момент времени исполнения будет установлен только для конечной вершины q_k всего графа $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$, который определяется требуемым моментом времени T_{max}^l выполнения всего задания Z_l. Если агент AR_1 обнаруживает ветвь $\boldsymbol{H}_1$ графа $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$, удовлетворяющую данным условиям, то он вступает в сообщество $\boldsymbol{R_1}$ по выполнению задания $Z_l \in \boldsymbol{Z}$.

БАЗОВАЯ ПРОЦЕДУРА МУЛЬТИАГЕНТНОГО РАСПРЕДЕЛЕНИЯ РЕСУРСОВ

4

При этом агент AR_1 осуществляет следующие модификации графа $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ задания Z_l:

- выбранная им для исполнения ветвь $\boldsymbol{H}_1$ удаляется из графа задания $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$, т.е. формируется новый граф $\boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1) = \boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)/\boldsymbol{H}_1$;
- всем вершинам этого модифицированного графа $\boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1)$, инцидентным вершинам удаленной ветви $\boldsymbol{H}_1$ приписываются моменты времени, когда соответствующие им операции должны быть выполнены с тем, чтобы «вписаться» в график выполнения ветви $\boldsymbol{H}_1$, закрепленной за его ресурсом R_1;
- номер ресурса R_1 записывается в список участников сообщества $\boldsymbol{R}_l$ по выполнению задания Z_l;
- значение времени ΔT^l задержки увеличивается на величину задержки выполнения ветви $\boldsymbol{H}_1$ ресурсом R_1.

5

Далее аналогичный выбор делает агент AR_2, представляющий ресурс R_2 – в модифицированном графе $\boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1)$ он выбирает такую ветвь $\boldsymbol{H}_2 =< q_1^2, q_2^2, \ldots, q_r^2 >$, которую с помощью «своего» ресурса R_2 он может выполнить к моменту времени, приписанному ее конечной вершине q_r^2 с минимальной задержкой (в результате модификации, произведенной агентом AR_1, в графе $\boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1)$ будет иметься некоторое множество вершин, которым приписаны требуемые моменты времени их исполнения). После этого агент AR_2 осуществляет модификацию графа задания на ДО:

- из графа $\boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1)$ исключается ветвь $\boldsymbol{H}_2$, в результате чего формируется новый граф $\boldsymbol{G}_l^2(\boldsymbol{Q}_l^2, \boldsymbol{X}_l^2) = \boldsymbol{G}_l^1(\boldsymbol{Q}_l^1, \boldsymbol{X}_l^1)/\boldsymbol{H}_2$;
- всем вершинам этого модифицированного графа $\boldsymbol{G}_l^2(\boldsymbol{Q}_l^2, \boldsymbol{X}_l^2)$, инцидентным вершинам удаленной ветви $\boldsymbol{H}_2$ приписываются моменты времени, когда соответствующие им операции должны быть выполнены с тем, чтобы «вписаться» в график выполнения ветви $\boldsymbol{H}_2$;
- номер ресурса R_2 записывается в список участников сообщества $\boldsymbol{R}_l$ по выполнению задания Z_l;
- значение времени ΔT^l задержки увеличивается на величину задержки выполнения ветви $\boldsymbol{H}_2$ ресурсом R_2 .

6

Далее аналогичным образом в процесс распределения вступает агент AR_1, представляющий ресурс R_3 и т.д., до тех пор, пока не будут разобраны все ветви графа задания $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$, т.е. пока он окажется, что после очередной модификации граф задания на ДО становится пустым.

АЛГОРИТМЫ РАБОТЫ АГЕНТОВ РЕСУРСОВ СРС ПРИ НАЛИЧИИ СТИМУЛИРУЮЩИХ ФАКТОРОВ

Премия, получаемая сообществом $\boldsymbol{R_l}$ за выполнение задания $\boldsymbol{Z_l}$.

$$S^l = \begin{cases} S^l_{max} \cdot \dfrac{\Delta T^l_{max} - \Delta T^l}{\Delta T^l_{max}}, \text{если } \Delta T^l < \Delta T^l_{max} \\ 0, \qquad\qquad \text{если } \Delta T^l \geq \Delta T^l_{max} \end{cases}$$

где S^l – величина премии, получаемой за выполнение задания Z_l;
S^l_{max} – максимальная величина премии, получаемая при условии, что задание Z_l выполнено к моменту T^l_{max};
ΔT^l – задержка выполнения задания Z_l относительно момента T^l_{max};
ΔT^l_{max} – максимально допустимая задержка выполнения задания Z_l относительно момента T^l_{max}.

Премия, начисляемая ресурсу $\boldsymbol{R_p} \subseteq \boldsymbol{R_l}$, участвовавшему в выполнении задания $\boldsymbol{Z_p}$.

$$S^l_p = \frac{S^l \cdot \sum_{i=1}^{K} v(O^p_i)}{\sum_{i=1}^{M} v(O_i)}$$

где S^l_p – количество премиальных баллов, начисляемых ресурсу R_p;
$\sum_{i=1}^{M} v(O_i)$ – суммарная трудоемкость всех операций O_i задания Z_l;
$\sum_{i=1}^{K} v(O^p_i)$ – суммарная трудоемкость операций O^p_i задания Z_l, выполненных ресурсом R_p;
$M = |\boldsymbol{Q_l}|$ – число вершин (операций) в графе $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ задания Z_l;
K – число операций задания Z_l, выполненных ресурсом R_p.

Агент AR_p должен выбирать для исполнения такую ветвь $\boldsymbol{H_f}$ задания Z_l, для которой величина

$$S^{lf}_p = \begin{cases} \dfrac{\sum_{i=1}^{k} v(O^f_i) \cdot S^l_{max} \cdot \left(\Delta T^l - T_{\text{тек}} + T^f_{k+1} - \sum_{j=1}^{k} t_p\left(O^f_j\right) - t_{\Pi}\right)}{\sum_{j=1}^{M} v(O_j) \cdot \Delta T^l}, \text{если } \Delta T^{lf}_p < \Delta T^l; \\ 0, \qquad\qquad \text{если } \Delta T^{lf}_p \geq \Delta T^l. \end{cases}$$

максимальна.

АЛГОРИТМЫ РАБОТЫ АГЕНТОВ РЕСУРСОВ СРС ПРИ НАЛИЧИИ ИНДИВИДУАЛЬНЫХ ШТРАФНЫХ САНКЦИЙ

Штраф, начисляемый сообществу $\boldsymbol{R_l}$ за задержку выполнения задания $\boldsymbol{Z_l}$ на время ΔT^l относительно требуемого момента времени T^l_{max} будет определяться величиной

$$\Delta S^l = S^l_{max} - S^l = S^l_{max} - S^l_{max} \cdot \frac{\Delta T^l_{max} - \Delta T^l}{\Delta T^l_{max}} = S^l_{max} \cdot \frac{\Delta T^l}{\Delta T^l_{max}}$$

причем

$$\Delta T^l \leq \Delta T^l_1 + \Delta T^l_2 + \ldots + \Delta T^l_n,$$

где ΔT^l_i (*i* = 1, 2, …, *n*) – задержка, вносимая *i*-м агентом сообщества $\boldsymbol{R}_l$;

$n = |\boldsymbol{R}_l|$.

Индивидуальный штраф, налагаемый на агента ресурса $\boldsymbol{R_p}$

$$\Delta S^l \leq S^l_{max} \cdot \frac{\Delta T^l_1}{\Delta T^l_{max}} + S^l_{max} \cdot \frac{\Delta T^l_2}{\Delta T^l_{max}} + \cdots + S^l_{max} \cdot \frac{\Delta T^l_n}{\Delta T^l_{max}}$$

Величина $\Delta S^l_p = S^l_{max} \cdot \frac{\Delta T^l_i}{\Delta T^l_{max}}$ определяет штраф, начисляемый на *i*-го агента AR_p сообщества $\boldsymbol{R_l}$ (*p* = 1, 2, …, *n*).

Учитывая, что $\Delta T^l_p = \begin{cases} T_{\text{тек}} - T^f_1, \text{если } T_{\text{тек}} > T^f_1 \\ 0, \quad \text{если } T_{\text{тек}} \leq T^f_1 \end{cases}$, (*p* = 1, 2, …, *n*)

где $T^f_1 = T^f_{k+1} - \left(\sum_{j=1}^{k} t_p(O^f_j) + t_{\text{п}}\right)$, получаем

$$\Delta S^l_p = S^l_{max} \frac{T_{\text{тек}} - T^f_{k+1} + \sum_{j=1}^{k} t_p(O^f_i) + t_{\text{п}}}{\Delta T^l_{max}}.$$

Агент AR_p должен выбрать для исполнения такую ветвь $\boldsymbol{H}_f$ задания Z_l, для которой величина $S^l_p = \begin{cases} S^l_{max} \cdot \frac{V^p_l}{V_l} - \Delta S^l_p, \text{если } \Delta T^l < \Delta T^l_{max}; \\ 0, \quad \text{если } \Delta T^l \geq \Delta T^l_{max}. \end{cases}$

где $V^p_l = \sum_{j=1}^{K} v(O^i_j)$ – суммарная трудоемкость операций задания Z_l, выполненных ресурсом R_p;

$V_l = \sum_{j=1}^{M} v(O_i)$ – суммарная трудоемкость всех вершин графа $\boldsymbol{G}_l(\boldsymbol{Q}_l, \boldsymbol{X}_l)$ задания Z_l;

ΔT^l – общая задержка выполнения задания Z_l относительно требуемого момента времени T^l_{max},

максимальна.

АЛГОРИТМ РАБОТЫ АГЕНТА РЕСУРСА В УСЛОВИЯХ ОГРАНИЧЕННОГО ЗАПАСА

Ресурс $\boldsymbol{R_i}$ (*i* = 1, 2, …, *N*) обладает некоторым ограниченным запасом $\boldsymbol{B_i}$ и при выполнении операции $O_j \in \boldsymbol{O_i}$ ресурс R_i тратит $b_i(O_j)$ своего запаса, причем

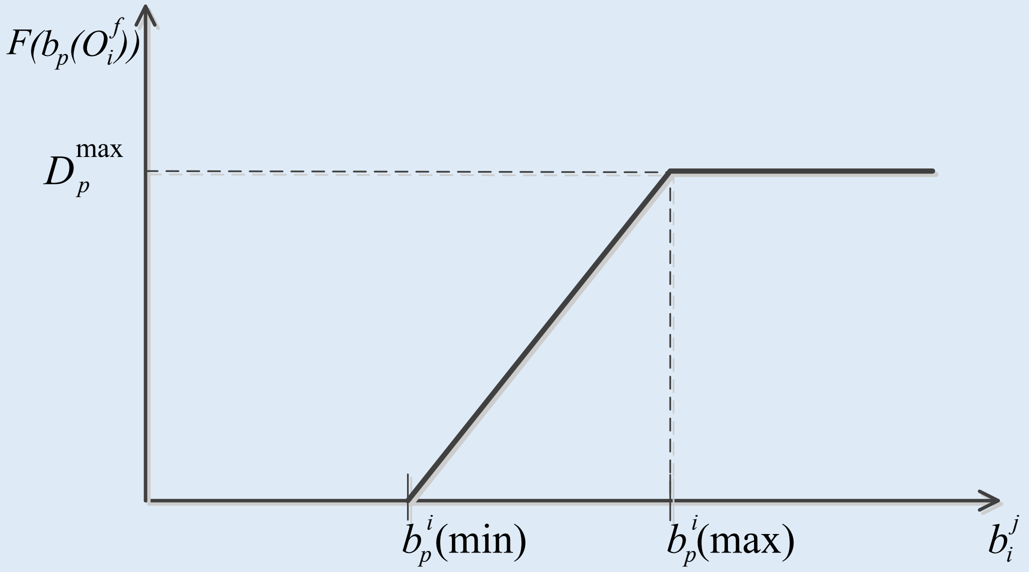
$$\sum_{i=1}^{k} b_p(O_i^f) \le B_p,$$

где k – общее число операций ветви $\boldsymbol{H_f}$ задания Z_l, выполняемых ресурсом R_p.

Будем считать, что производительность $D_p(O_i^f)$ ресурса R_p при выполнении операции O_i^f ветви $\boldsymbol{H_f}$ задания Z_l зависит от выделяемого им для этого запаса b_p^i, т.е.

$D_p(O_i^f) = F(b_p(O_i^f))$,

где F – некоторая функция.



Вид функции $F(b_p(O_i^f))$

т.е.

$$D_p(O_i^f) = \begin{cases} 0, & \text{если } b_p(O_i^f) < b_{min}(O_i^f) \\ K_i b_p(O_i^f), & \text{если } b_{min}(O_i^f) \le b_p(O_i^f) \le b_{max}(O_i^f) \\ D_p^{\max}(O_i^f), & \text{если } b_p(O_i^f) > b_{max}(O_i^f), \end{cases}$$

где K_i – некоторый коэффициент.

АЛГОРИТМ РАСПРЕДЕЛЕНИЯ ЗАПАСА В МЕЖДУ ОПЕРАЦИЯМИ ВЕТВИ H_f, ВЫПОЛНЯЕМОЙ РЕСУРСОМ R_p

1. Для всех операций O_i^f $(i = 1,2,\dots,k)$ ветви $\boldsymbol{H}_f =< q_1^f, q_2^f, \dots, q_k^f >$, принимается, что $b_p\left(O_i^f\right) = b_{max}(O_i^f)$ $i = 1,2,\dots,k)$.

2. Если $\sum_{i=1}^{k} b_p\left(O_i^f\right) \le B_p$, где B_p – суммарный запас, имеющийся в распоряжении ресурса R_p, то переход к 9, иначе

3. Для всех операций O_i^f $(i = 1,2,\dots,k)$ ветви $\boldsymbol{H}_f$ вычисляется значение

$$\Delta t_p\left(O_i^f\right) = \frac{v\left(O_i^f\right)}{K_i\left(b_p\left(O_i^f\right) - 1\right) \cdot b_p\left(O_i^f\right)}.$$

4. Операции O_i^f $(i = 1,2,\dots,k)$ ветви $\boldsymbol{H}_f$ ранжируются в порядке возрастания величины $\Delta t_p\left(O_i^f\right)$.

5. $i = 1$.

6. Если $b_p\left(O_i^f\right) - 1 > b_{min}(O_i^f)$, то перейти к 8, иначе

7. $i = i + 1$, если $i > k$, то перейти к 10,иначе перейти к 6.

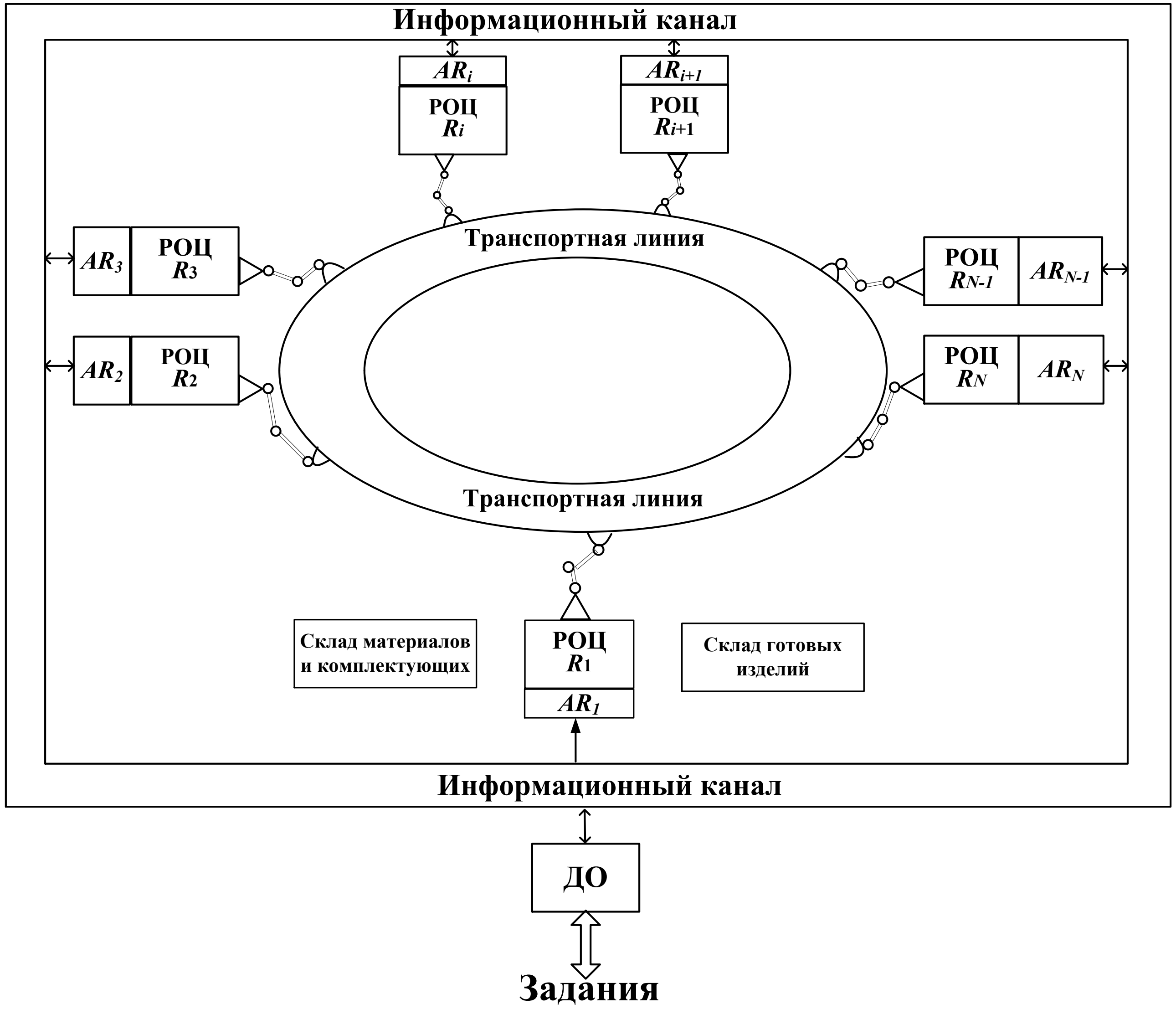
8. $b_p\left(O_i^f\right) = b_p\left(O_i^f\right) - 1$, перейти к 2.

9. Оптимальное распределение запаса B_p ресурса R_p по операциям O_i^f $i = 1,2,\dots,k)$ ветви $\boldsymbol{H}_f$ найдено, перейти к 11.

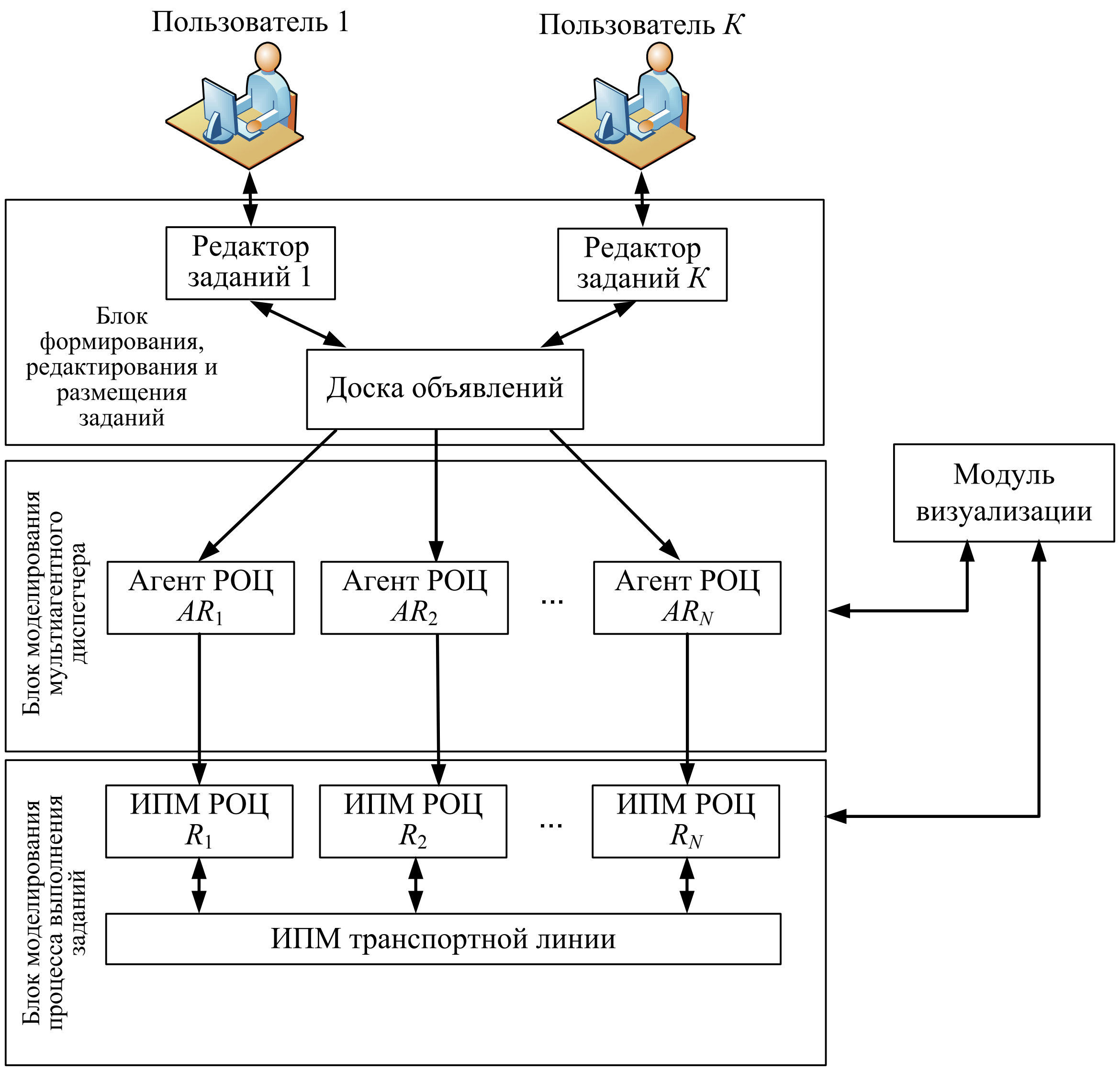
10. Ресурс R_p не может выполнить ветвь $\boldsymbol{H}_f$ вследствие недостаточности имеющегося у него запаса B_p.

11. Конец.

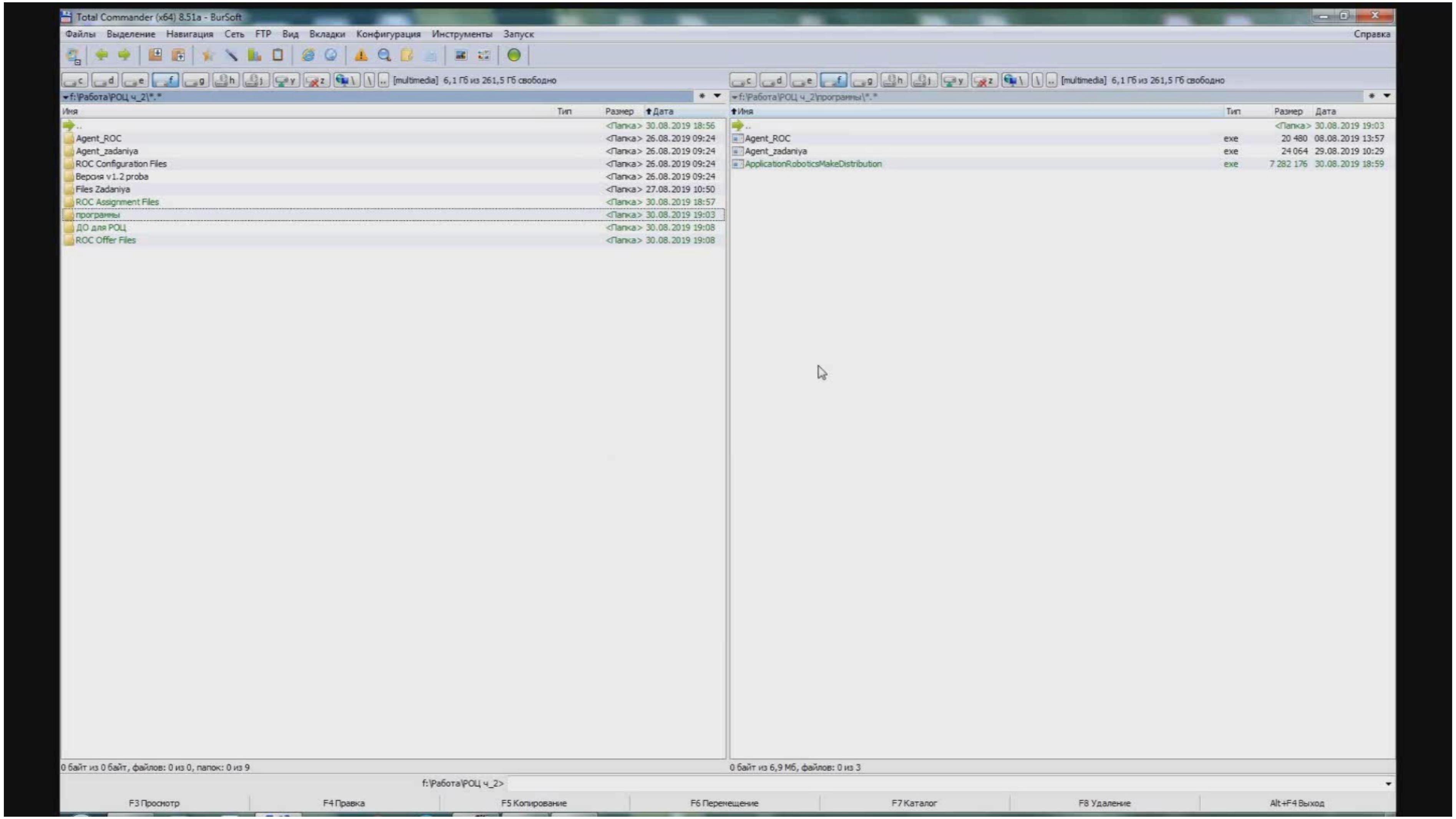
САМООРГАНИЗУЮЩИЕСЯ БЕЗЛЮДНОЕ РОБОТИЗИРОВАННОЕ ПРОИЗВОДСТВО



КОМПЛЕКС ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ САМООРГАНИЗУЮЩЕГОСЯ БРП



САМООРГАНИЗУЮЩИЕСЯ БЕЗЛЮДНОЕ РОБОТИЗИРОВАННОЕ ПРОИЗВОДСТВО

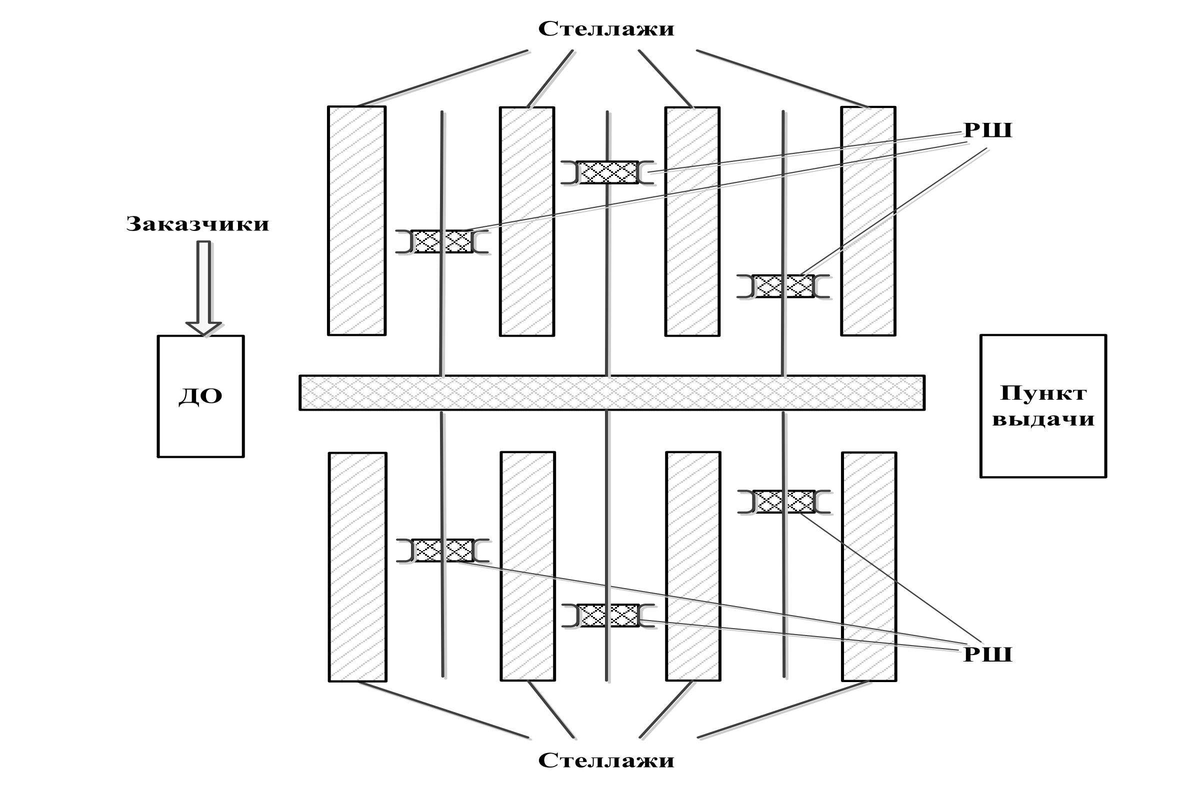


РОБОТИЗИРОВАННЫЕ СКЛАДСКИЕ КОМПЛЕКСЫ

Внешний вид РСК

Внешний вид робота-штабелера стеллажного

РОБОТИЗИРОВАННЫЕ СКЛАДСКИЕ КОМПЛЕКСЫ

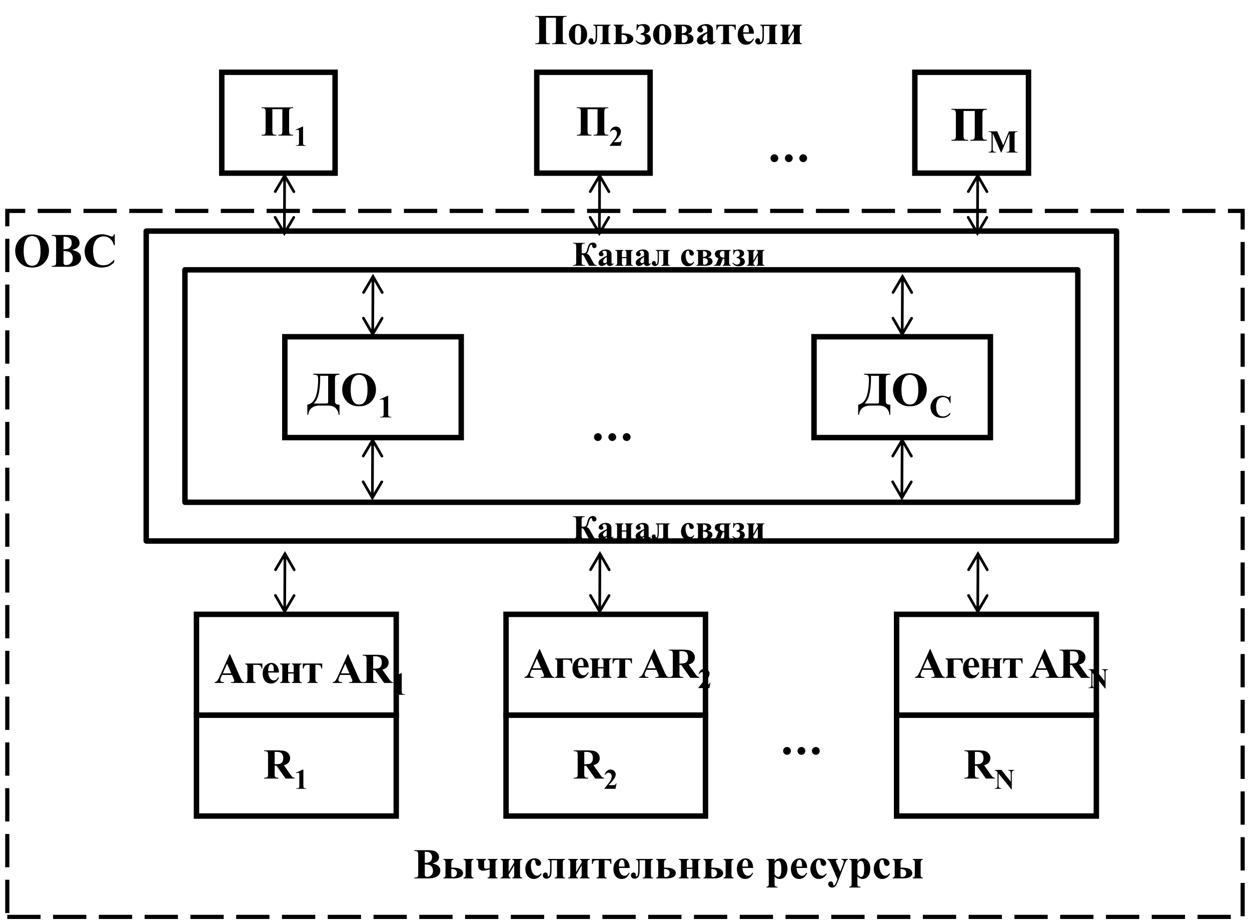


Каждый РШ обладает своим локальным устройством управления (УУ), обеспечивающим выполнение операций 4-х типов.

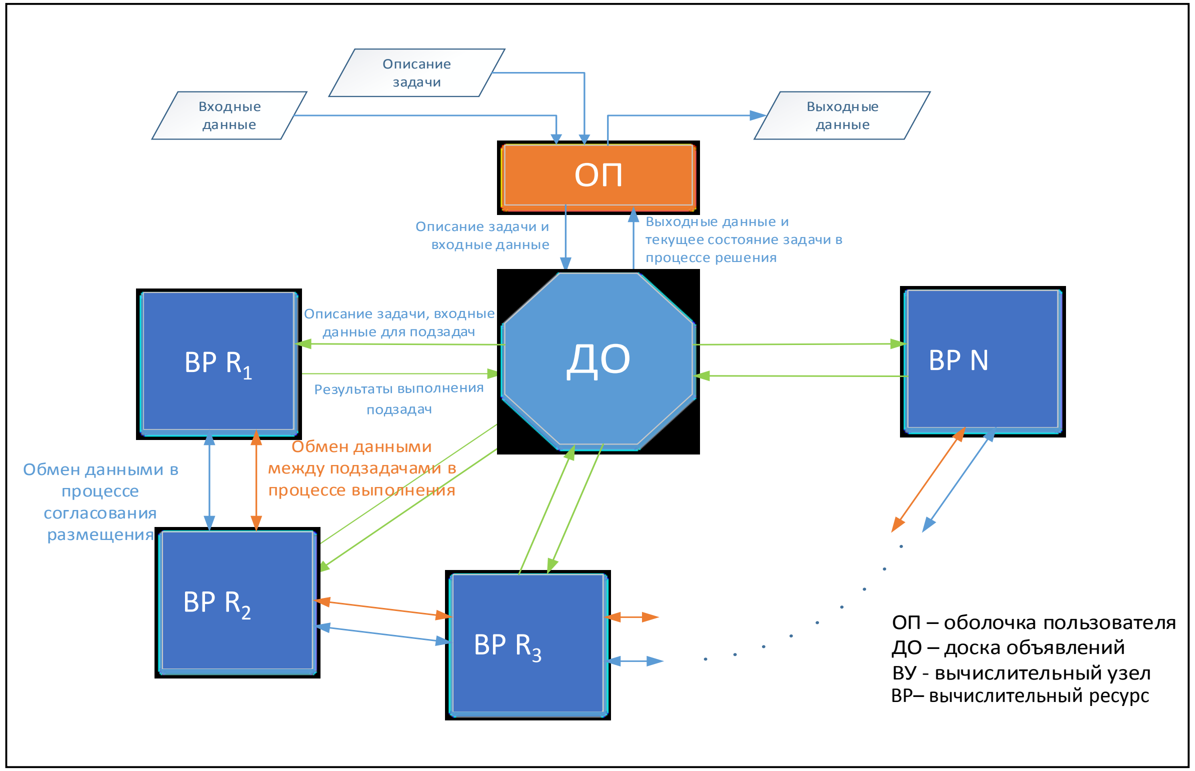
1. Переместить захват РШ к ячейке Y стеллажа M.
2. Взять захватом из ячейки Y стеллажа M деталь D_i.
3. Переместить деталь D_i к транспортной линии.
4. Поместить деталь D_i на транспортную линию.

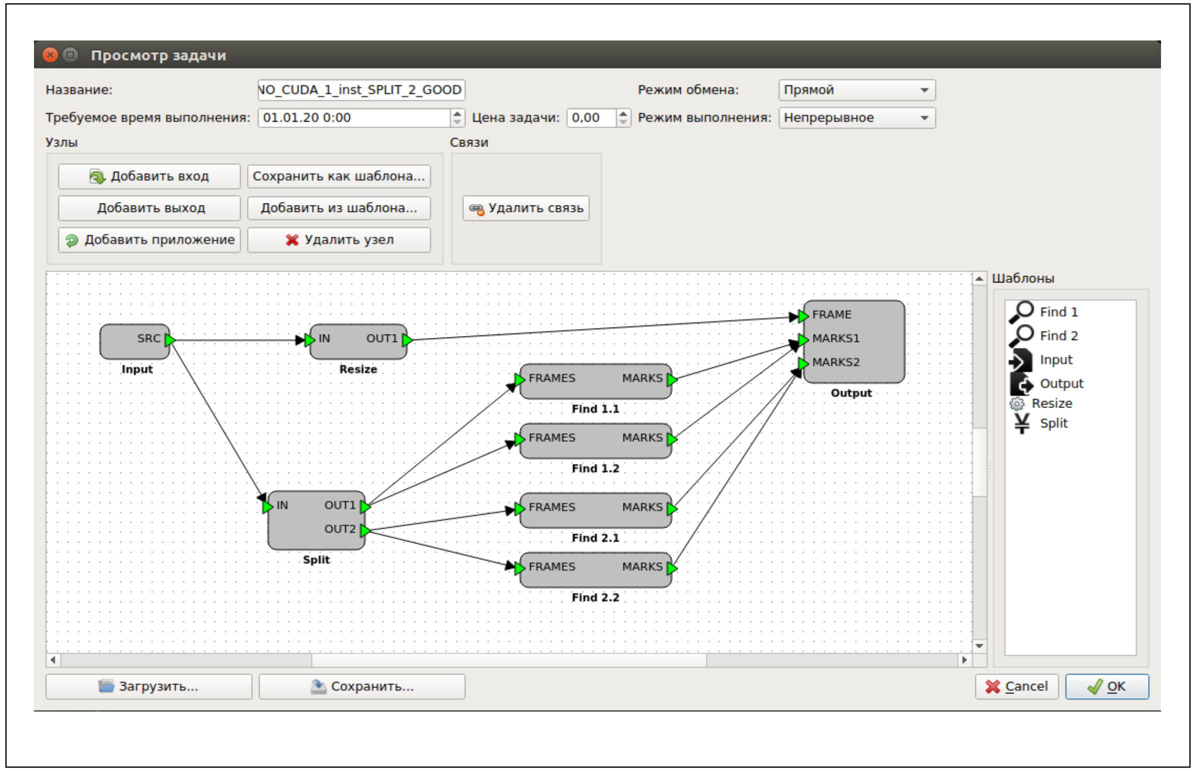
Внедрение метода мультиагентного диспетчирования позволило до 15% сократить среднее временя задержки доставки Заказчикам требуемых деталей роботами-штабелерами, на 20% уменьшить время простоя РШ при выполнении множества заданий, на 30% увеличить максимальное число отказов РШ, при котором гарантировано сохранение работоспособности системы.

Структура ОВС с мультиагентным диспетчером



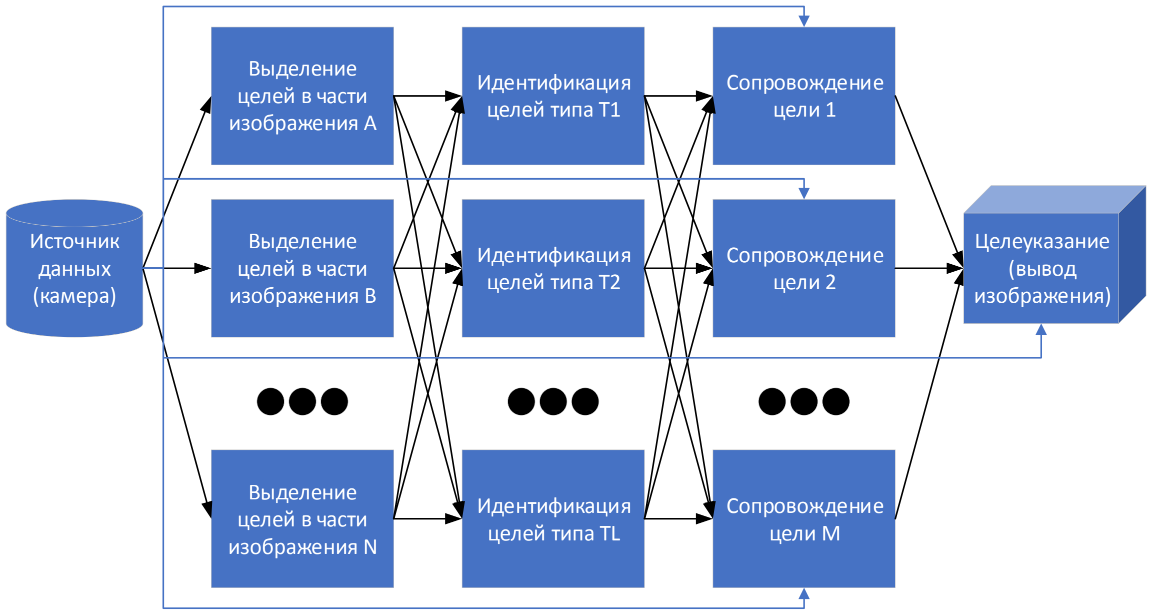
ПРОГРАММНАЯ МОДЕЛЬ САМООРГАНИЗУЮЩЕЙСЯ ОВС



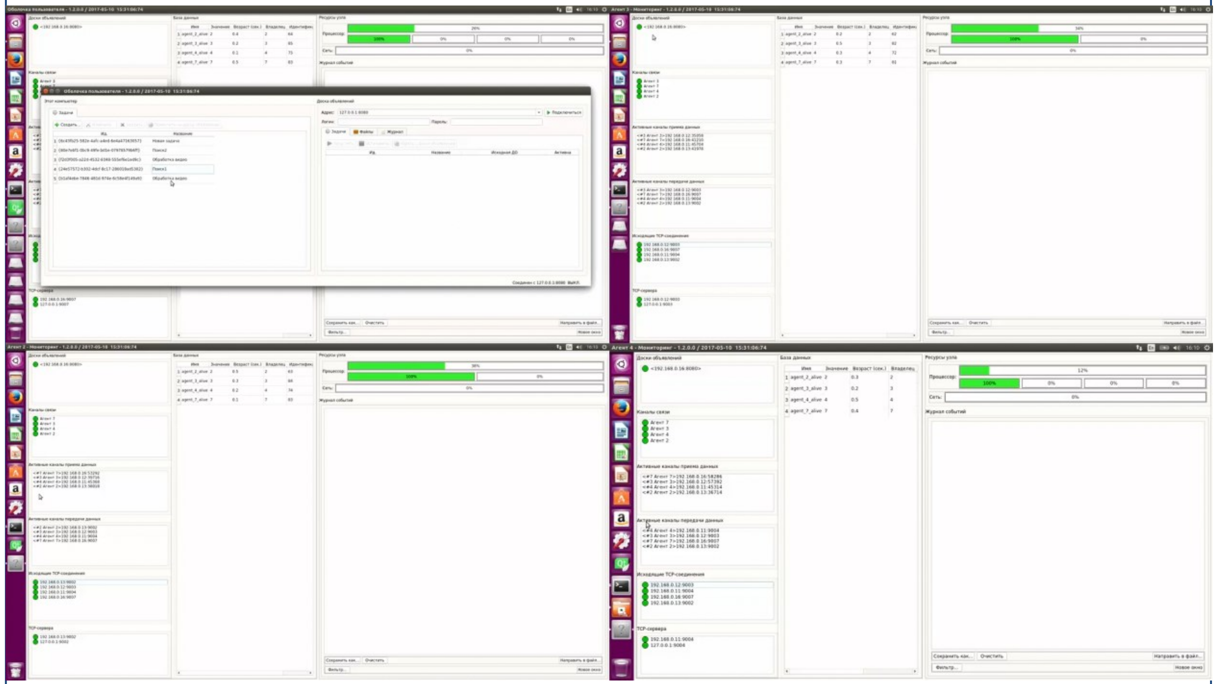


Результаты экспериментов с программной моделью показали, что среднее время задержки выполнения множества заданий в самоорганизующейся ОВС с мультиагентным диспетчером от 25 до 42% меньше, чем в ОВС с централизованным диспетчером.

ВЫЧИСЛИТЕЛЬНАЯ СРЕДА ДЛЯ РЕШЕНИЯ ЗАДАЧ ПОТОКОВОЙ ОБРАБОТКИ ВИДЕОИНФОРМАЦИИ ОТ МНОЖЕСТВА ВИДЕОКАМЕР

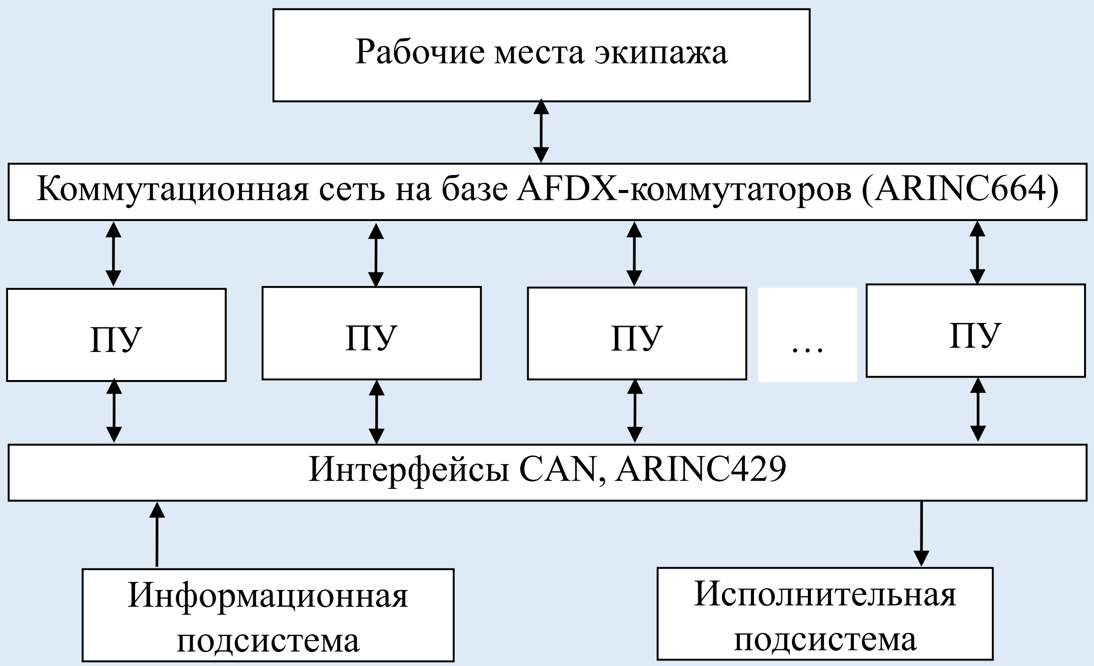


Обобщенная схема архитектуры мультиагентной распределенной системы автоматического видеообнаружения, идентификации и сопровождения группировок объектов.

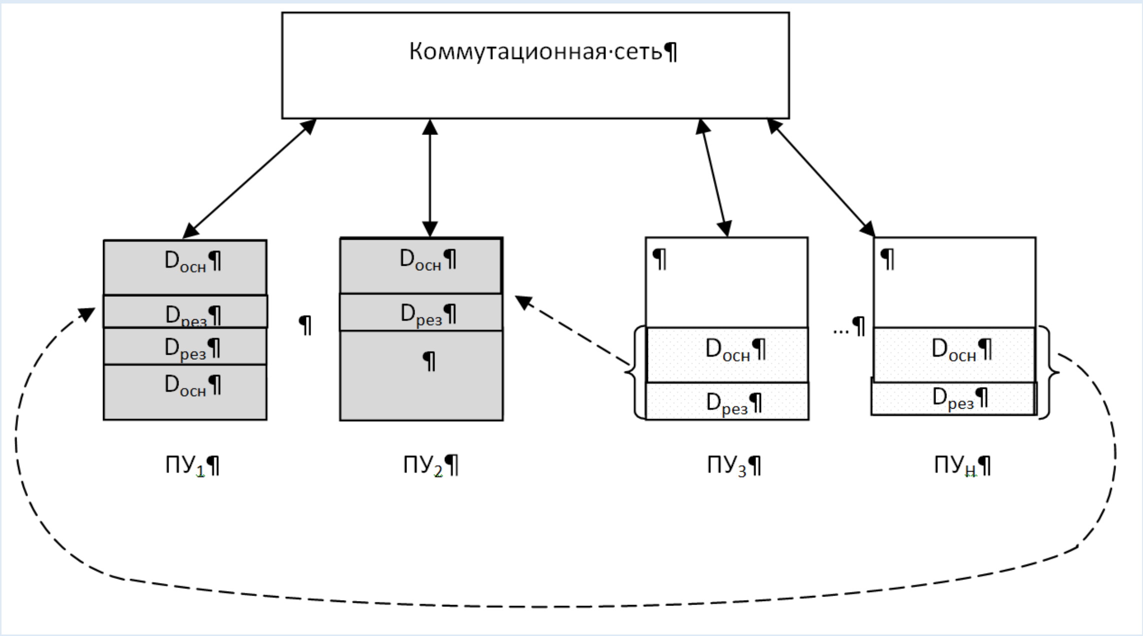


САМОРГАНИЗУЮЩАЯСЯ ГЕТЕРОГЕННАЯ ВЫЧИСЛИТЕЛЬНАЯ СРЕДА СКЦ «ПОЛИТЕХНИЧЕСКИЙ»

САМООРГАНИЗУЮЩИЕСЯ СИСТЕМЫ УПРАВЛЕНИЯ ВОЗДУШНЫХ СУДОВ

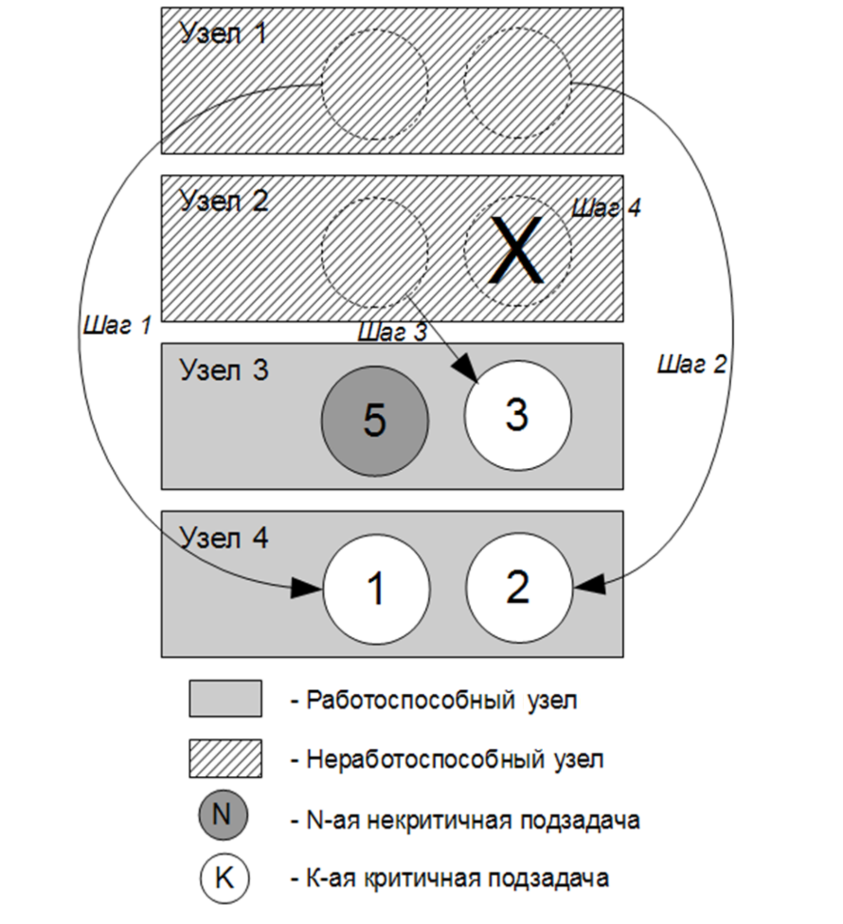


Обобщенная структура СРС с архитектурой ИМА

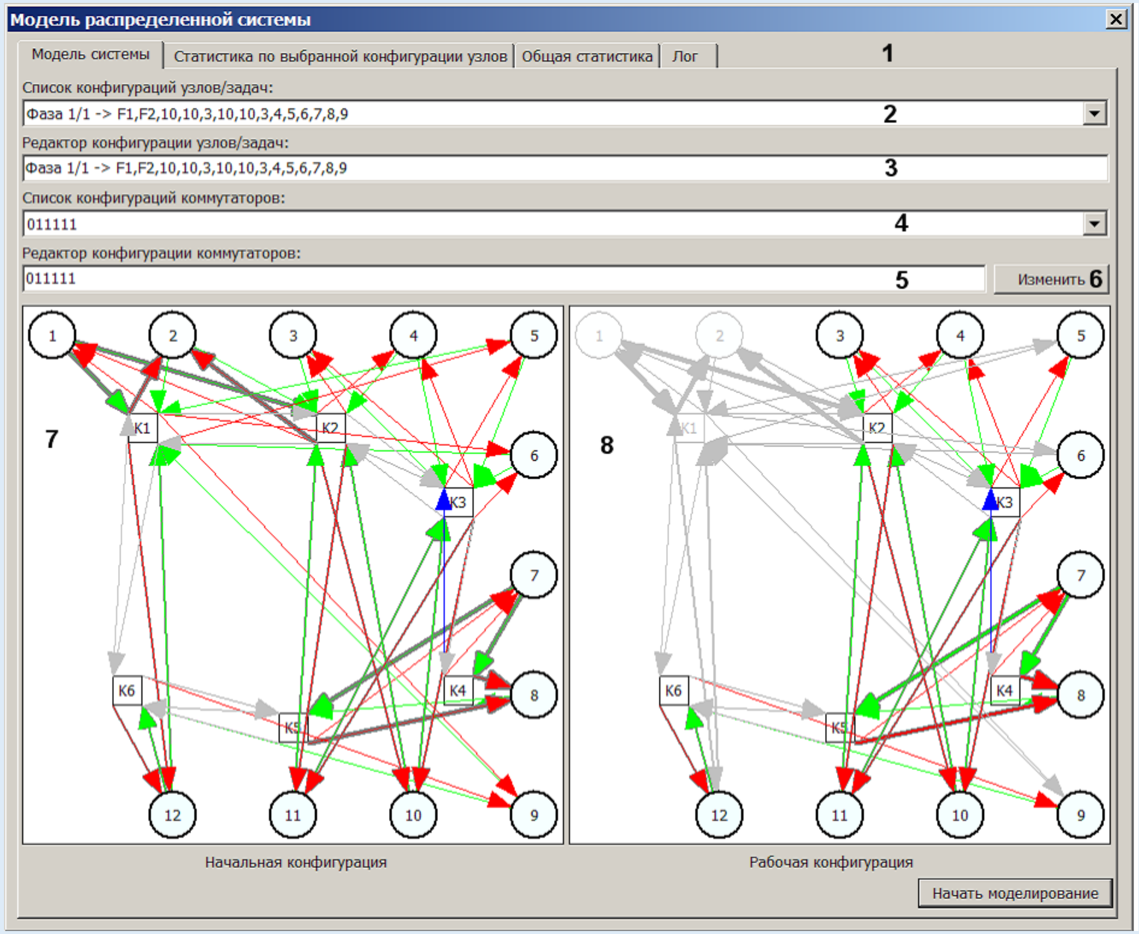


Мультиагентное перераспределение вычислительных ресурсов ИМА

САМООРГАНИЗУЮЩИЕСЯ СИСТЕМЫ УПРАВЛЕНИЯ ВОЗДУШНЫХ СУДОВ

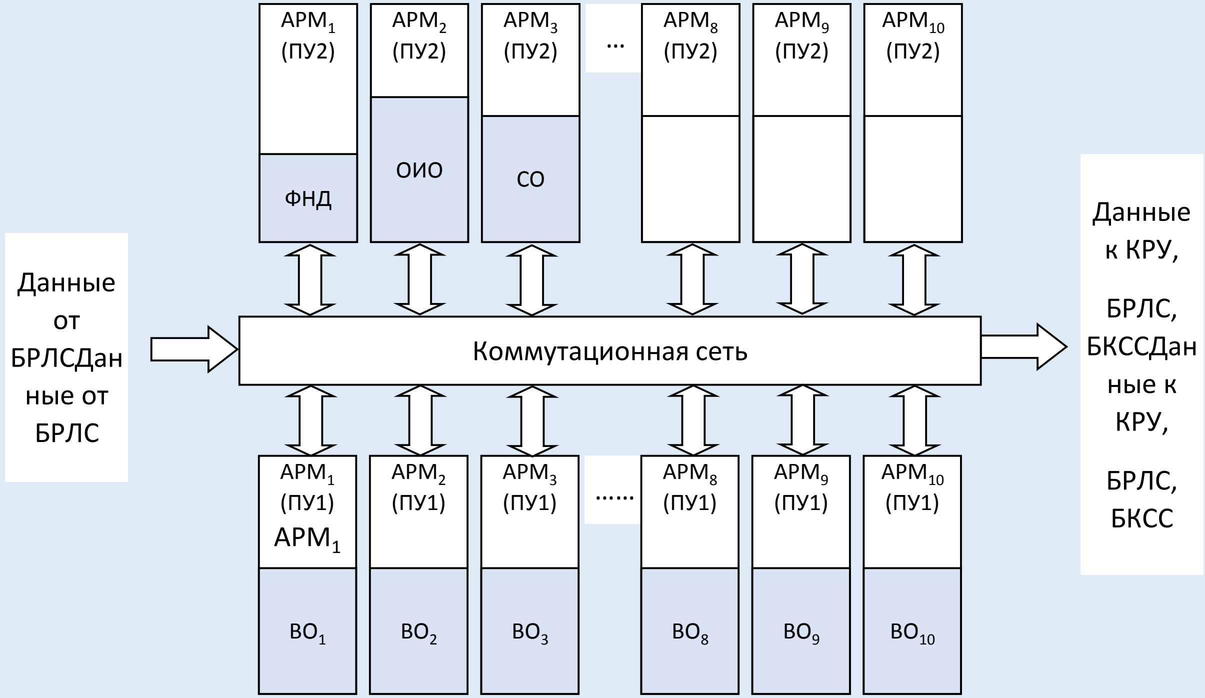


Реконфигурация ИМА после отказа части узлов



Окно вывода информации о реконфигурациях ИМА в случае отказа части системы

САМООРГАНИЗУЮЩИЕСЯ СИСТЕМЫ УПРАВЛЕНИЯ ВОЗДУШНЫХ СУДОВ



Применение метода самоорганизации в БИУС самолета ДРЛО позволило обеспечить его устойчивость к 5 отказам ПУ и повысить гамма-процентную наработку на 20% (при заданной вероятности безотказной работы 0,9999)

Спасибо за внимание!