

УДК 004.272.43

## ТРАНЗАКТНО-ПРОЦЕССНЫЙ ПОДХОД К ФОРМАЛИЗАЦИИ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ИНФОРМАЦИИ В СЕТИ МНОГОПРОЦЕССОРНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ

С.Ф. Маслович

Гомельский государственный университет им. Ф. Скорины, Гомель

## TRANSACTIONAL-PROCESS APPROACH TO FORMALIZATION SIMULATION OF DISTRIBUTED DATA PROCESSING IN THE NETWORK OF MULTIPROCESSOR COMPUTING SYSTEM

S.F. Maslovich

F. Scorina Gomel State University, Gomel

Описывается транзактно-процессный подход к формализации распределенной обработки информации в сети многопроцессорной вычислительной системы. Задача распределенной обработки представляется в виде сетевого графика. Динамика взаимодействия компонентов сети по использованию ресурсов ее узлов исследуется с помощью двух типов имитационных моделей: модель оборудования сети и модель рабочей нагрузки, имеющая вид сетевого графика.

**Ключевые слова:** имитационная модель, рабочая нагрузка, распределенная обработка информации, многопроцессорная вычислительная система.

Transactional-process approach to the formalization of distributed information processing network in a multiprocessor computing system is described. The problem of distributed processing is represented as a network schedule. Dynamics of the interaction of network components with the use of the resources of its nodes is studied with the help of two types of simulation models: a model of network equipment and model of the workload, which has the type of a network schedule.

**Keywords:** simulation model, workload, distributed processing, multiprocessor computer system.

### Введение

В статье предлагаются формальные модели вычислительного процесса в многопроцессорных вычислительных системах (МВС) и локальных вычислительных сетях (ЛВС), позволяющие отобразить динамику расхода ресурсов узлов МВС запросами рабочей нагрузки (РН) для распределенной обработки информации (РОИ). При формализации данного рода объекта используется *транзактно-процессный способ формализации* вычислительного процесса (ВП) и РН. Транзакты сложной структуры представляются в виде сетевого графика запросов задач для обработки на узлах сети. Причем задачи РОИ используются наравне с существующими ранее в сети МВС задачами диалогового и пакетного типов.

### 1 Формализация модели оборудования многопроцессорной вычислительной системы

Если вычислительный процесс в МВС представляет собой взаимодействие двух составляющих (РН и устройств узла, предоставляющих свои ресурсы для этой РН), то формализацию ВП в МВС представим тремя группами формальных описаний: поведение устройств узла МВС; запросов РН; отображение динамики их взаимодействия. Таким образом, узел МВС представим набором характеристик аппаратных и

программных компонент [1]. В качестве аппаратных компонентов выступают: устройства  $CPU_i$ , имитирующие обработку задач на процессорах узла, и устройство  $MEM$ , имитирующее обработку задач на устройстве внешней памяти. Перед каждым из этих устройств узла имеется дополнительная очередь, предназначенная только для хранения запросов задач, поступающих на обработку к  $CPU_i$  и  $MEM$ . Таким образом, аппаратные устройства узла МВС представляются парами устройство и очередь к ним. В качестве программного компонента выступает устройство диспетчер ( $DISP$ ), обеспечивающее выбор задач на обработку из очереди к устройствам  $CPU_i$ . Аппаратная составляющая узла МВС состоит из статической и динамической частей (статическими частями являются устройства  $MEM$  и  $DISP$ , а динамической составляющей является устройство  $CPU_i$ ). Количество этих устройств может меняться в зависимости от преследуемых исследователем-проектировщиком целей моделирования. Устройство  $DISP$  является местом анализа запросов, поступающих на устройства  $CPU_i$  или  $MEM$  задач, и принятия решений по их рациональному распределению по ресурсам узлам.

Рабочая нагрузка, поступающая на узел МВС, описывается набором запросов аппаратных средств узла МВС. Структура РН была

определена при постановке НЭ. В терминах используемого способа формализации запросам пользователей соответствуют транзакты. Транзакты выступают в качестве динамических компонент, использующих ресурсы узла МВС. В зависимости от типа каждый транзакт имеет уникальную структуру. Например, транзакты диалогового типа ( $TR_{\text{ДИАЛ}}$ ) характеризуются: порядковым номером  $j$ , количеством квантов использования ресурса  $CPU$  ( $n_j$ ), временем обдумывания ( $t_{\text{обд}}$ ), приоритетом  $\pi_{1j}$ . Поэтому тело транзакта диалогового типа имеет следующий вид:  $TR_{\text{ДИАЛ}} = (j, n_j, t_{\text{обд}}, \pi_{1j})$ .

Транзакты отложенного счета ( $TR_{\text{ОТЛ.СЧЕТ}}$ ) имеют более сложную структуру, проецируемую на устройства узла МВС. Данный тип транзактов имеет следующий вид:  $TR_{\text{ОТЛ.СЧЕТ}} = (j, k, (m_{1j}^1; m_{2j}^1), \dots, (m_{1j}^k; m_{2j}^k), \pi_{2j})$  (где  $j$  – номер транзакта,  $k_j$  – количество циклов CPU-МЕМ,  $m_{1j}^l$  – количество квантов длительностью  $\Delta_1 t$  использования ресурса CPU на  $l$ -ом цикле ( $l=1, \dots, k$ ),  $m_{2j}^l$  – количество квантов длительностью  $\Delta_2 t$  использования ресурса МЕМ  $l$ -ом цикле ( $l=1, \dots, k$ ),  $\pi_{2j}$  – приоритет  $j$ -ой задачи).

Каждый тип транзакта генерируется соответствующим его типу источником-генератором ( $GEN_{\text{ДИАЛ}}$  и  $GEN_{\text{ОТЛ.СЧЕТ}}$ ) с соответствующими частотами  $\lambda_{\text{ДИАЛ}}$  и  $\lambda_{\text{ОТЛ.СЧЕТ}}$ . Однако в генерации транзактов первого и второго типов имеются отличия: задачи диалогового типа поступают в систему группами величина ( $v_{\text{group}}$ ) которой является параметром модели. Транзакты режима отложенного счета поступают в систему по одному последовательно. При этом транзактам диалогового типа присваивается больший приоритет, чем транзактам отложенного счета. Между очередями к устройствам  $CPU_i$  и самими устройствами используется модель программного блока  $DISP$ , которая осуществляет их распределения по  $CPU_i$ .

## 2 Модель распределенной обработки информации в сети многопроцессорной вычислительной системы

Описание РОИ в МВС основывается на временных диаграммах выполнения задач РОИ на узлах и в сети МВС. Эта модель РН РОИ представляет собой вероятностный сетевой график (ВСГР) (рисунок 1). Узлами ВСГР являются свершения событий в этом графике, а ветвями отображаются запросы на ресурсы  $j$ -го узла МВС, необходимые для выполнения  $S_j$ -го модуля. Модель задачи РОИ представим вероятностным сетевым графиком, конструируемых двумя типами элементов:  $SOB_{mj}$  и  $SOB_{nj}$  (номера свершения событий  $m$  и  $n$  после выполнения запросов ресурсов на  $j$ -ом узле МВС).  $MTXO_{mnh}$  представляют собой микротехнологические

операции события ВСГР, инициируемые пользователем  $h$ -го узла МВС, выполняемые после свершения  $m$ -го события и активизирующие  $n$ -е событие, соответствующее очередному модулю использования ресурсов  $j$ -го узла. Значения времен запросов ресурсов  $CPU_i$  и  $MEM$  узла МВС подзадачами ( $p$ ) РОИ известны заранее на основе ранее проведенных НЭ и задаются тройкой значений:  $\{j_p, t_{p0CPU}, t_{p0MEM}\}$  (где  $j_p$  – номер узла выполнения подзадачи  $p$  (значение может быть как случайным, так и заранее заданным);  $t_{p0CPU}$  – заказ времени счета на подзадачи  $p$  на  $CPU$ ;  $t_{p0MEM}$  – заказ времени счета на подзадачи  $p$  на  $MEM$ ).

Из-за наличия в узлах МВС собственного случайного потока запросов на ресурсы задач диалогового типа и задачи отложенного счета, выполнение  $MTXO_{mnh}$  расхода ресурсов узла МВС  $CPU_i$  и  $MEM$  также представляют собой случайную последовательность:  $\{(\tau_{mnjh}, V_{mnjh}, P_{jh})\}$ , (где  $\tau_{mnjh}$ ,  $V_{mnjh}$  – соответственно время использования  $CPU_i$  и места на  $MEM$ ;  $P_{jh}$  – вероятность обмена  $CPU_i$  с  $MEM$  в квантах использования ресурсов узла  $j$ ). При этом предполагается, что исследователю известны регрессионные зависимости между временем использования  $MEM$   $j$ -го узла  $\tau_{mnjh}$  и размером используемого места  $V_{mnjh}$ :  $\tau_{mnjh} = \varphi_j(V_{mnjh})$ .

Каждая  $MTXO_{mnh}$  является операцией (подзадачей) обработки части общей распределенной задачи в МВС. Фактическое время, затраченное на выполнения  $MTXO_{mnh}$  ( $\tau_{MTXO_{mnh}}$ ), определяется как разность между временем прибытия ( $t_{inmnh}$ ) и временем отправки ( $t_{outmnh}$ ) управляющего транзакта ( $UT_{mnh}$ ) на узлы МВС, которые в данный момент времени могут обрабатывать иные задачи, и поэтому  $UT_{mnh}$  приходится ожидать в очереди к  $j$ -му узлу длительностью ( $\tau_{ожетmnjh}$ ). Таким образом, время обработки  $h$ -ой реализации  $MTXO_{mnh}$  на узле МВС определяется по формуле:  $\tau_{MTXO_{mnh}} = t_{inmnh} - t_{outmnh}$ .

Времена выполнения  $MTXO_{mnh}$  ( $\tau_{MTXO_{mnh}}$ ) являются случайными величинами из-за случайного характера обработки информации на узлах различного класса собственных задач в МВС. Поэтому для определения среднего времени выполнения  $MTXO_{mnh}$  необходимо использование метода Монте-Карло [3].

Методика расчета параметров ВСГР реализуется следующей последовательностью шагов.

**Шаг 1.** На  $s$ -ой реализации ВСГР ( $s=1, \dots, N$ ) фиксируются значения времен  $MTXO_{mn}$  ( $\tau_{MTXO_{mns}}$ ). В каждой  $s$ -ой реализации ВСГР заменяется обычным сетевым графиком с постоянными значениями параметров  $MTXO_{mns}$ , согласно известной методике расчета и анализа статистик реализации операций  $\{MTXO_{mn}\}$  и свершения событий  $\{SOB_m\}$ .

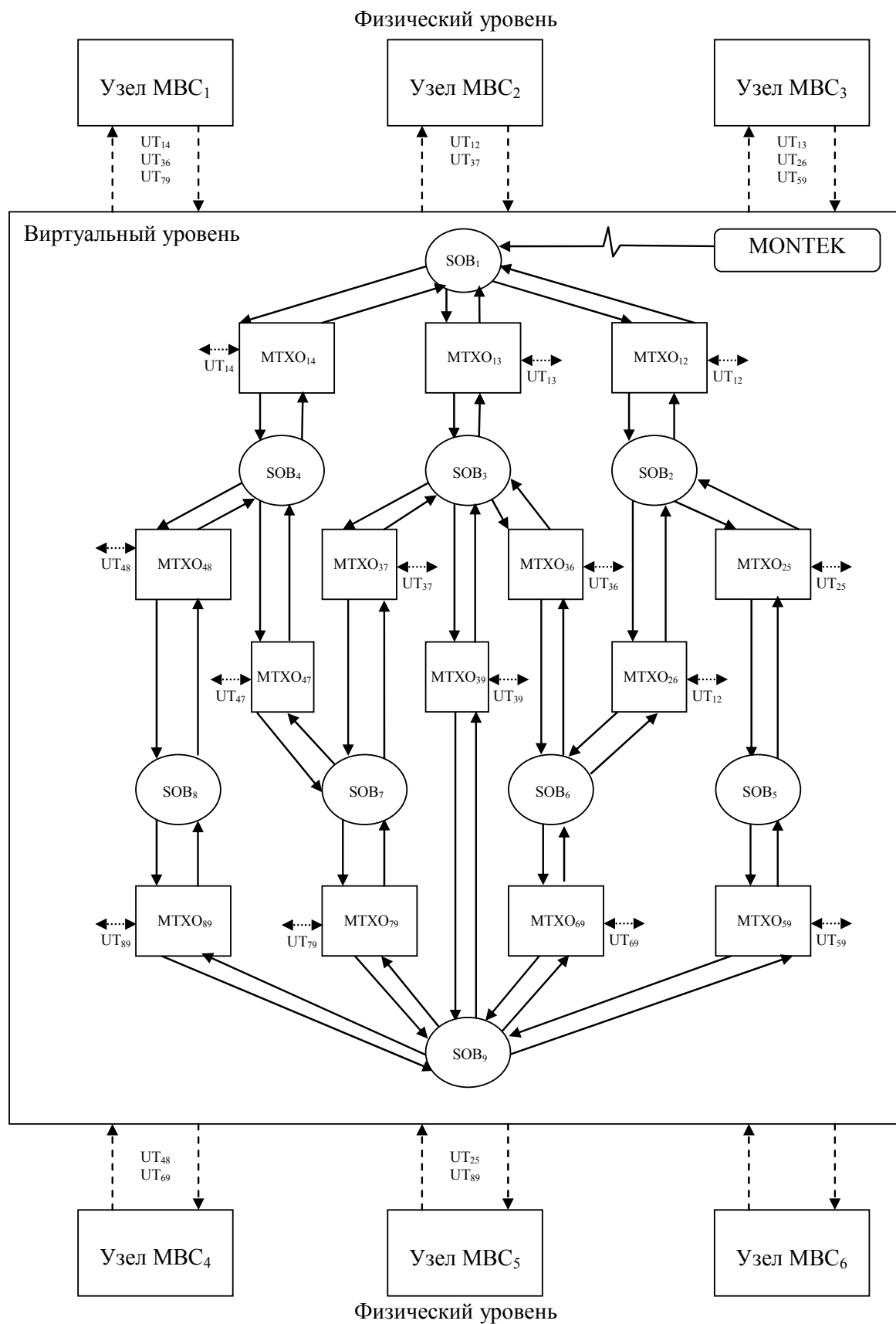


Рисунок 1 – Структура ИМ ВСГР РОИ в MBC

**Шаг 2.** Определяются ранние сроки свершения событий ( $t_{pms}$ ). Формируются статистики ( $\tau_{MTXO_{mn}}$ ) при реализации  $\{MTXO_{mn}\}$ . Вычисляются поздние сроки свершения событий ( $t_{nms}$ ), начиная с завершающего события в обратном порядке и кончая исходным событием.

**Шаг 3.** Рассчитываются резервы свершения событий  $\{R_{ms}\}$  и типовые статистики выполнения  $MTXO_{mns}$  при реализации сетевого графика ( $t_{pnmns}$ ,  $t_{nmmns}$ ,  $t_{pomns}$ ,  $t_{nomns}$ ). Завершается анализ  $s$ -ой реализации ВСГР нахождением критического пути ( $KPP_{ks}$ ) при реализации  $k$ -го запроса распределенной обработки на МВС.

**Шаг 4.** В результате имитации  $N$  реализаций ВСГР в базе данных комплекса формируются выборки: значений статистик реализации событий  $\{t_{pms}, t_{nms}, R_{ms}\}$ , статистик реализации  $MTXO_{mn}$   $\{t_{pnmns}, t_{nmmns}, t_{pomns}, t_{nomns}\}$ , критических путей  $\{KPP_{ks}\}$ . Каждой  $s$ -ой реализации ВСГР в этих выборках соответствуют  $s$ -ые номера статистик реализации  $\{SOB_m\}$ ,  $\{MTXO_{mn}\}$ ,  $\{KPP_{ks}\}$ .

Анализ ВСГР по данным ИЭ реализуется следующей последовательностью шагов.

**Шаг 1.** Формирование по выборкам математических ожиданий ( $M_Z$ ) и выборочных дисперсий  $D_Z$ . Здесь под  $Z$  понимаем обозначение перечисленных ранее статистик свершения  $\{SOB_m\}$ , выполнения  $\{MTXO_{mn}\}$  и длины критического путей в ВСГР ( $T_{KPPk}$ ).

**Шаг 2.** Анализ структуры  $KPP_k$ , представляет собой последовательность чередования  $MTXO_{mn}$  и  $SOB_m$  с нулевыми резервами времени их свершения ( $R_{ms}=0$ ). В общем случае для  $N$  реализаций ВСГР может существовать множество  $\{KPP_k\}$ , в котором только некоторые пары ( $SOB_m$ ,  $MTXO_{mn}$ ) различны, а остальные пары могут не отличаться друг от друга. Поэтому исследователю сообщается диапазон реализации  $SOB_m$ , возникающих в ВСГР в одно и тоже модельно время  $t_0$  при различных реализациях  $\{KPP_k\}$  в ВСГР.

**Шаг 3.** С помощью подпрограмм вторичной обработки результатов определяются статистики реализации:  $\{MTXO_{mn}\}$ ,  $\{SOB_m\}$ ,  $\{KPP_k\}$ . Формируется граф критических путей ( $GRKPP_k$ ) и множество оценок значений коэффициентов напряженности выполнения  $\{MTXO_{mn}\}$ . Составляется также список тех  $SOB_m$ , которые имеют большие значения оценок математического ожидания резервов времени их свершения ( $R_m$ ). Из этого списка выбираются те  $MTXO_{mn}$ , которые фигурируют затем в качестве кандидатов на изъятие от них резервов ресурсов для последующей передачи этих ресурсов в распоряжение  $MTXO_{mn}$ , находящихся на наиболее вероятном критическом пути с целью уменьшения времени его свершения.

**Шаг 4.** Если множество  $\{KPP_k\}$  достаточно большое, то на его основе формируется граф критических путей  $GRKPP_k$ . Этим завершается

очередная итерация поиска наиболее вероятного критического пути, когда вместо  $BCGP_k$  исследуется уже сам  $GRKPP_k$ . После нескольких итераций число вероятных критических путей существенно сокращается и далее исследователь на основе анализа содержимого ветвей оставшегося графа  $GRKPP_k$  может определить, какая из них является наиболее вероятной.

Состав параметров, статистик и откликов модели РОИ в МВС приведен в таблице 1.

### 3 Модели организации взаимодействия компонентов многопроцессорной вычислительной системы и рабочей нагрузки

Динамика взаимодействия компонентов МВС использования ресурсов узлов МВС исследуется с помощью двух типов имитационных моделей (ИМ). В МВС используются только два типа РН: РОИ и диалоговый режим [2].

**Первый тип ИМ** (рисунок 2) отображает расход запросами РН  $i$ -го типа ресурсов оборудования  $j$ -го узла МВС (ИМ  $ОБОР_j$ ) (здесь  $i$ -номер типа РН, а  $j$ -номер узла МВС). ИМ  $ОБОР_j$  является универсальной для обоих типов ИМ и состоит из следующих процессов: расхода ресурсов центрального процессора  $j$ -го узла МВС ( $PR.CPU_j$ ); расхода ресурсов внешней памяти ( $PR.MEM_j$ ). Эти процессы отображают алгоритмы расхода ресурсов узла МВС запросами РН каждого типа, которые обслуживаются процессами согласно следующим приоритетам. Запросы режима  $DR_i$  имеют более высокий приоритет, чем запросы задачи РОИ. Выделение ресурсов в процессе  $PR.CPU_j$  осуществляется квантами времени использования  $CPU_j$  длительности  $\Delta\tau_{cpi}$ . Появление более приоритетного запроса приводит к прерыванию выделения ресурса менее приоритетному запросу. Ресурс устройства МЕМ выделяется запросам РН полностью без прерываний на время  $\Delta\tau_{MEM}$ . В случае, когда задаче требуется больше времени, чем один квант, то она снова поступает в очередь к устройствам  $CPU_j$  для ожидания очередного кванта  $\Delta\tau_{cpi}$ .

Аналогичная схема организации обработки задач происходит на устройстве МЕМ. ИМ  $ОБОР_j$  является универсальной и не требует программирования на этапах эксплуатации со стороны руководства МВС. Из-за того, что программы процессов ИМ  $ОБОР_j$  являются реентерабельными, исследователям для задания структуры и состава узлов МВС достаточно указать только количественный состав МВС.

**Второй тип ИМ** (рисунок 3) отображает во времени последовательность  $i$ -го типа запросов РН на ресурсы  $j$ -го узла МВС (ИМ РОИ). ИМ рабочей нагрузки также является универсальной для обоих типов МВС и состоит из следующих типов процессов: генератор запросов диалоговых задач ( $GENDIAL_i$ ); генератор запросов на распределенную обработку информации ( $GENPOI_q$ ).

Таблица 1 – Состав параметров, статистик и откликов РОИ в МВС

Название вектора параметров	Обозначение	Назначение и сущность параметра
Входные параметры	$\lambda_{DIALi}$	Частота поступления задач диалогового типа
	$\lambda_{OTLi}$	Частота поступления задач отложенного счета
	$\lambda_{POIi}$	Частота поступления задач РОИ
	$\vartheta_{CPU}$	Скорость работы процессора узла сети МВС
	$\vartheta_{MEM}$	Скорость работы памяти узла сети МВС
	$BCGP_i$	Структура i-ой задачи РОИ
	$t_{pOCPU}$	Заказ времени счета подзадачи p на cpu i-ой задачи РОИ
	$t_{pOMEM}$	Заказ времени счета подзадачи p на mem i-ой задачи РОИ
Статистики и отклики реализации	$T_{проп}$	Среднее время пропуска задач через сеть МВС
	$\eta_{CPUi}, \eta_{MEMi}, \eta_{MBC}$	Коэффициенты загрузки устройств сети (процессора, память, вся сеть МВС)
	$\bar{t}_{ожCPUi}, \bar{t}_{ожMEMi}, \bar{t}_{MBC}$	Средние времена ожидания к устройствам сети (процессора, память, вся сеть МВС)
Статистики и отклики реализации	$\bar{l}_{CPUi}, \bar{l}_{MEMi}, \bar{l}_{MBC}$	Средние длины очередей к устройствам сети МВС (процессора, память, вся сеть МВС)
	$\bar{LT}_{CPUi} = \bar{t}_{ожCPUi} \cdot \bar{l}_{CPUi}$ $\bar{LT}_{MEMi} = \bar{t}_{ожMEMi} \cdot \bar{l}_{MEMi}$ $\bar{LT}_{MBC} = \bar{t}_{ожMBC} \cdot \bar{l}_{MBC}$	Коэффициенты литтла устройствам сети (процессор, память, всей сети МВС)
	$\tau_{MTXOij}$	Времена выполнения MTXO <sub>ij</sub>
	$L_i$	Последовательность вершин критических путей задачи рои
	$T_{крит}$	Длина критического пути
	$t_{pij}$	Ранние сроки свершения событий
	$t_{pij}$	Поздние сроки свершения событий
	$R_{ij}$	Резервы времен выполнения событий
	$t_{pniij}$	Ранние начальные времена реализации MTXO <sub>ij</sub>
	$t_{pniij}$	Поздние начальные времена реализации MTXO <sub>ij</sub>
	$t_{poiij}$	Ранние окончательные времена реализации MTXO <sub>ij</sub>
	$t_{poiij}$	Поздние окончательные времена реализации MTXO <sub>ij</sub>

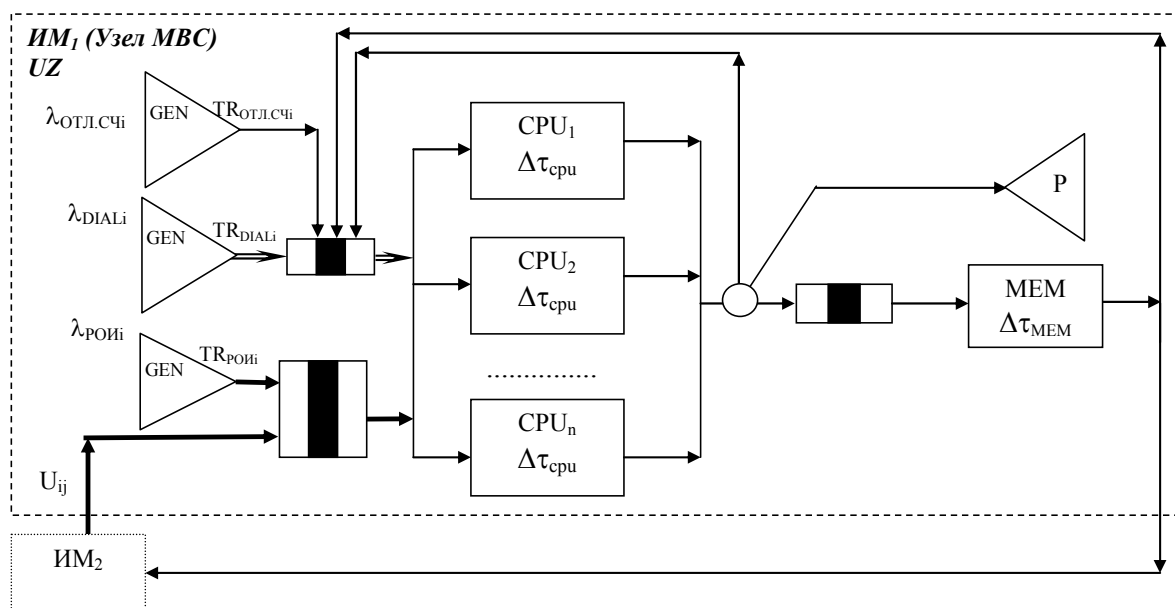


Рисунок 2 – Схема модели имитационной модели узла МВС (ИМ ОБОР)

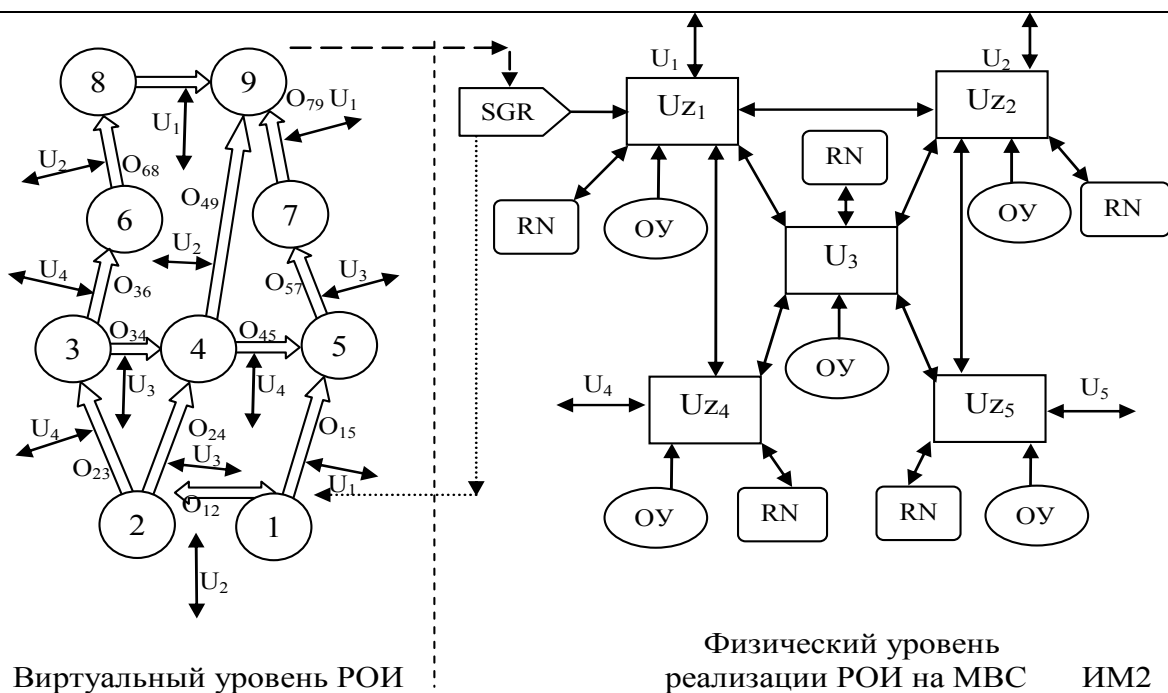


Рисунок 3 – Схема имитационной модели РОИ на MBC

Для задания состава и структуры РН на узлах MBC руководству достаточно задать число процессов-генераторов транзактов сложной структуры каждого типа ( $I, Q$ ). Все эти генераторы имеют стандартный алгоритм имитации вероятностных запросов ресурсов узлов MBC, что позволяет реализовать их в виде реентерабельных программ [1]. В информационной базе данных (ИБД) ИМ MBC для каждой версии генераторов отведено место для хранения рабочей информации и сбора статистики имитации. В итоге ИМ рабочей нагрузки состоит из двух реентерабельных программ-генераторов запросов ресурсов узлов MBC, которые в своей работе используют индивидуальные рабочие места в ИБД (соответственно по одной области в ИБД для каждой версии генераторов).

#### 4 Имитация динамики взаимодействия РН и оборудования MBC

Каждый  $i$ -ый генератор, используя рабочие места в ИБД по адресу  $\alpha_i$ , формирует транзакт сложной структуры ( $TRSS_i$ ) стандартного типа, а в одну из очередей оборудования MBC посылает триаду  $(i, \pi_i, \alpha_i)$ . Здесь  $i$ -номер запроса ресурсов;  $\pi_i$  – приоритет запроса;  $\alpha_i$  – адрес информационной части запросов  $i$ -го типа.

$GENPOI_i$  с интенсивностью  $\alpha_i$  формирует «подкрашенный» транзакт вида  $TRSSI=(i, \pi_i, \alpha_i)$ , у которого в информационной части находятся запросы ресурсов  $CPU_i$  и  $MEM_i$ , разыгранные по соответствующим функциям распределения, сформированным до имитационного эксперимента (ИЭ) по данным мониторинга поведения диалоговых запросов.

$GENPOI_q$  с интенсивностью  $\alpha_q$  формирует «подкрашенный» транзакт вида  $TRSS2_q=(q, \pi_q, \alpha_q)$ , у которого в информационной части находятся запросы ресурсов  $CPU_q$  и  $MEM_q$ , разыгранные по соответствующим функциям распределения по данным мониторинга поведения РОИ. Важным отличием информационной части этого транзакта является описание структуры РОИ, которая представляет собой вероятностный сетевой график ( $VSGR$ ). Узлами  $SGR$  являются свершение событий в  $SGR$ , а ветвями являются запросы на ресурсы  $q$ -го узла MBC, необходимые для выполнения  $S_q$ -го модуля задачи управления объектами реального времени.

Таким образом, в транзакте 2-го типа  $TRSS2_q$  указывается адрес возврата на  $S_q$ -ый модуль выполнения  $SGR$  после имитации счета этого модуля на  $q$ -ом узле MBC. Из-за вероятностного характера запросов ресурсов узлов MBC РОИ  $q$ -го узла используется процедура Монте-Карло. В ходе  $l$ -ой реализации РОИ  $q$ -го узла MBC по соответствующим функциям формируются  $CPU_{ql}$   $MEM_{ql}$ , что позволяет при имитации  $l$ -ой реализации  $SGR$  считать эти запросы детерминированными величинами. Поэтому  $GENPOI_q$  рассчитывает моменты свершения событий, начиная от начального события  $SGR_l$  и заканчивая завершающим событием  $SGR_l$ . На событиях  $SGR_l$  осуществляется синхронизация их свершения, и по завершении самого позднего момента выделения ресурсов MBC активизируются все модули РОИ, начинающиеся в только что завершеном событии. В моменты завершения  $l$ -ой реализации  $SGR_l$  фиксируется в ИБД статистика свершения всех событий в информационном

поле транзакта  $TRSS2_q$  по адресу  $\alpha_q$ . При этом, ведется контроль за окончанием числа реализаций процедуры Монте-Карло для РОИ  $q$ -го узла МВС. Когда  $l > N_m$  (где  $N_m$  – количество реализаций процедуры Монте-Карло) эта статистика усредняется, что является статистиками и откликами имитации РОИ в МВС. На основе полученных данных определяются необходимые параметры сетевого графика  $q$ -ой задачи РОИ такие, как критический путь  $\{l_{qкрит}\}$ , его длина ( $T_{qкрит}$ ), времена свершения ранних  $\{t_{pниq}\}$  и поздних сроков  $\{t_{пdq}\}$  выполнения задач, значения резервов времен каждого события  $\{t_{резервq}\}$ . По завершении  $N_m$  реализаций ИМ МВС также определяются усредненные значения откликов и статистик оборудования ИМ МВС: коэффициенты загрузки устройств  $CPU$   $\{\eta_{срj}\}$  и МЕМ  $\{\eta_{МЕМj}\}$ , длин очередей к этим устройствам  $\{l_{срj}, l_{МЕМj}\}$ , времена ожидания запросов к ним  $\{t_{ожсрj}, t_{ожМЕМj}\}$ .

### Заключение

В статье предложен транзактно-процессный способ формализации модели распределенной обработки информации в МВС на основе двухступенчатой реализации задач РОИ в МВС. На

первом (верхнем) уровне РН режима РОИ представлена вероятностным сетевым графиком (виртуальный уровень), а на втором (нижнем) уровне запросы каждого модуля РОИ решаются на оборудовании того узла (физический уровень), который предусмотрен в сетевом графике, но захватывает ресурсы узла уже в порядке приоритетов режимов обработки информации в узле МВС или ЛВС.

### ЛИТЕРАТУРА

1. Майоров, С.А. Основы теории вычислительных систем : учебное пособие / С.А. Майоров. – М. : Высшая школа, 1978. – 224 с.
2. Маслович, С.Ф. Имитация динамики взаимодействия оборудования многопроцессорной вычислительной системы с моделями рабочей нагрузки / С.Ф. Маслович // Известия Гомельского гос. ун-та им. Ф. Скорины. – Гомель, 2008. – № 5 (50). – С. 78–81.
3. Соболев, И.М. Метод Монте-Карло / И.М. Соболев. – Москва : Наука, 1978. – 64 с.

Поступила в редакцию 10.03.10.