

Моделирование параллельных бизнес-процессов

В статье рассматривается новый раздел современной теории бизнес-процессов, посвященный их параллельной обработке. Актуальность работы обусловлена необходимостью реорганизации бизнес-процессов с целью улучшения ряда их ключевых показателей, таких как скорость, стоимость и т.д.

Следует отметить, что известные модели и методы современной теории бизнес-процессов практически не затрагивают вопросы параллелизма в них, хотя в общей теории процессов такое направление занимает значительное место и, помимо собственно методов описания, анализа и верификации параллельных конструкций, включает также и методы распараллеливания последовательных конструкций.

В работе предлагается классификация параллельных бизнес-процессов, базирующаяся на типах параллельных вычислительных систем и полностью покрывающая рассматриваемую предметную область. Приведены примеры параллельных бизнес-процессов каждого типа. Для каждого из выделенных классов бизнес-процессов дается краткий обзор существующего положения дел в части параллелизма, выделяются и анализируются направления соответствующих работ. Даны краткие описания моделей и методов конвейерного и синхронного параллелизма.

В части асинхронного параллелизма предлагается модель и языковые средства организации параллельных независимых ветвей (процессов). В качестве соответствующей модели используется смешанный граф, содержащий узлы и ветви различных типов. Формализуется сценарий выполнения параллельного

бизнес-процесса. Для оценки времени выполнения параллельного бизнес-процесса адаптируются основополагающие теоремы теории вычислительных процессов с целью их ориентации на бизнес-процессы, даются рекомендации по правилам построения параллельных бизнес-процессов. Формулируются требования к языковым средствам описания асинхронных параллельных бизнес-процессов.

Далее рассматриваются методы распараллеливания бизнес-процессов, касающиеся его линейных участков и циклов. Вводятся определения линейного участка и цикла в бизнес-процессе. Для их распараллеливания ключевой является задача выявления информационных зависимостей между функциями бизнес-процесса. Внимание обращается на последовательность определений/использований информационных объектов, проявляющихся через входящие и исходящие информационные потоки для функциональных объектов соответствующего визуального языка моделирования. В терминах вышеупомянутого графа бизнес-процесса формализуются понятия информационной зависимости различных видов.

Предлагаемая работа является одной из первых попыток построения формальной базы организации параллелизма в бизнес-процессах, играющего важное значение для улучшения ключевых показателей бизнес-процессов в задачах их реорганизации.

Ключевые слова: бизнес-процесс, параллельный бизнес-процесс, конвейерный, синхронный и асинхронный параллелизм, распараллеливание бизнес-процесса.

Georgiy N. Kalyanov

Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia

Parallel Business Processes Modeling

The article considers a new section of the modern theory of business processes dedicated to their parallel processing. The relevance of the work is due to the need to reorganize business processes in order to improve a number of their key indicators, such as speed, cost, etc.

It should be noted that the well-known models and methods of the modern theory of business processes practically do not address the issues of parallelism in them, although this direction occupies a significant place in the general theory of processes and, in addition to the actual methods of description, analysis and verification of parallel structures, also includes methods of parallelization of sequential structures.

The paper proposes a classification of parallel business processes based on the types of parallel computing systems and completely covering the subject area under consideration. Examples of parallel business processes of each type are given. For each of the selected classes of business processes, a brief overview of the current state of affairs in terms of parallelism is given; the directions of relevant work are identified and analyzed. Brief descriptions of models and methods of pipelined and synchronous parallelism are given.

In terms of asynchronous parallelism, a model and language tools for organizing parallel independent branches (processes) are proposed. A mixed graph containing nodes and branches of various types is used as the corresponding model. The scenario of executing a parallel business process is formalized. To estimate the execution time of a parallel

business process, the basic theorems of the theory of computational processes are adapted in order to focus them on business processes, recommendations are given on the rules for building parallel business processes. The requirements for the language tools for describing asynchronous parallel business processes are formulated.

Next, the methods of parallelization of business processes related to its linear sections and cycles are considered. The definitions of the linear section and the cycle in the business process are introduced. For their parallelization, the key task is to identify information dependencies between the functions of the business process. Attention is drawn to the sequence of definitions/uses of information objects, manifested through incoming and outgoing information flows for functional objects of the corresponding visual modeling language. In terms of the above-mentioned business process graph, the concepts of information dependence of various types are formalized.

The proposed work is one of the first attempts to build a formal basis for the organization of parallelism in business processes, which plays an important role in improving the key indicators of business processes in the tasks of their reorganization.

Keywords: business process, parallel business process, pipelined, synchronous and asynchronous parallelism, business process parallelization.

Введение

Современная теория бизнес-процессов (БП) [1] является одним из разделов теории процессов и базируется на таких направлениях математической теории программирования как формальные грамматики и языки; методы тестирования программно-обеспечения (ПО); методы оптимизации, верификации, анализа и оценки качества ПО; теория баз данных и баз знаний; структурные и объектно-ориентированные методы анализа и проектирования и др. Основные направления и разделы теории бизнес-процессов рассмотрены в работах [2, 3], приведем здесь лишь их краткое перечисление:

- визуальные языки моделирования и графические модели БП,
- технологии моделирования,
- методы структурирования/декомпозиции,
- методы инжиниринга/реинжиниринга,
- методы анализа и верификации,
- методы перехода от моделей БП к требованиям по автоматизации БП.

Следует отметить, что включенные в вышеперечисленные разделы теории БП модели и методы практически не затрагивают вопросы параллелизма в БП, хотя в общей теории процессов такое направление занимает значительное место и, помимо собственно методов описания, анализа и верификации параллельных конструкций, включает также и методы распараллеливания последовательных конструкций.

Необходимо также заметить, что еще Хаммер и Чампи в [4] отмечали эффективность распараллеливания БП, например, отказ от линейности выполнения БП и переход к «естественному» (по сути – параллельному) порядку выполнения с целью резкого улуч-

шения их характеристик.

С другой стороны, параллелизм заложен в самой природе БП. Более того, такие механизмы синхронизации как семафоры Дейкстры, условные (критические) интервалы, мониторы и управляющие выражения [5] неформально присутствуют, например, в регламентах работы исполнителей с документами. Другое дело, что в языках моделирования БП подобные конструкции практически не используются, что исключает саму возможность исследования специфики параллельных БП.

Классификация видов параллелизма в БП

Для решения задач, связанных с параллелизмом, были выделены следующие классы БП, отражающие характер реализуемого параллелизма и основанные на известной классификации параллельных систем, предложенной Флинном [6]:

- ОКОД (Одиночный поток Команд и Одиночный поток Данных) – последовательные БП;
- МКМД (Множественный поток Команд и Множественный поток Данных) – асинхронные БП;
- ОКМД (Одиночный поток Команд и Множественный поток Данных) – синхронные (векторные) БП;
- МКОД (Множественный поток Команд и Одиночный поток Данных) – конвейерные или магистральные БП.

Данная классификация строится на понятиях потока данных и потока управления, являющихся основополагающими в современной теории БП, в частности, эти объекты являются ключевыми в языках моделирования БП.

Очевидно, что первый и второй классы в комментариях не нуждаются, соответствующие фрагменты имеются практически в любом БП. Конвей-

ерные и синхронные БП также встречаются достаточно часто. Классический пример конвейерного БП представляет собой процесс подготовки самолета (борта) к вылету: большое количество операций (техническое обслуживание, погрузка багажа, посадка пассажиров и т.д.) осуществляется над каждым экземпляром одного объекта. При этом исполнитель каждой из операций (бригада грузчиков, бригада механиков и т.д.) после выполнения своей операции переходит к следующему экземпляру объекта. Таким образом, над каждым из экземпляров одновременно могут выполняться различные операции, что и отражает суть конвейерного параллелизма.

Примером синхронного процесса является процесс написания отчета в крупной научной организации, которая выполняет государственные заказы, именуемые проектами. Каждый из проектов разбивается на темы, каждая из тем, в свою очередь, разделяется на работы, которые выполняются в рамках подразделений организации. По приказу руководства к определенному сроку в каждом из таких подразделений синхронно формируются отчеты по запланированным работам, далее руководители тем на их основе формируют (также синхронно) отчеты по темам, а затем руководители проектов – отчеты по проектам. Следует отметить, что могут иметься связи как между работами, так и между темами. Однако, это не противоречит концепции ОКМД-параллелизма, более того, такие возможности прямо заложены в архитектурные решения ОКМД-процессоров.

Направления работ в области параллелизма БП

Работы по проектированию и исследованию параллельных БП на современном этапе развиваются в двух направлениях:

создание визуальных языков, позволяющих описывать параллельные конструкции, и разработка методов распараллеливания описаний на традиционных языках визуального моделирования. В свою очередь, очевидно, что для решения первой задачи возможны два подхода – расширение существующих языков с целью их ориентации на параллельные БП и создание новых языков моделирования параллельных БП.

В качестве примеров параллельных расширений можно привести соответствующие конструкции в DFD-технологии [7] и языке UML [8]. В обоих случаях моделирование параллельного поведения сводится к описанию параллельных потоков управления и способов взаимодействия между ними с использованием аппарата конечных автоматов. В DFD-технологии такие потоки являются объектами диаграмм потоков управления CFD (Control Flow Diagrams) на верхнем уровне и диаграмм переходов состояний STD (State Transition Diagrams) на нижних уровнях, соответственно. В UML предусмотрен целый спектр специальных средств, предназначенных для более тонкого, детального и явного моделирования поведения при параллельном выполнении процесса (в данном случае – вычислительного). Однако и здесь применяются диаграммы переходов состояний за счет их расширения с использованием параллельных составных состояний и введения ряда дополнительных конструкций, таких как активное состояние, конфигурация активных состояний, область параллельного составного состояния, составной переход, синхронизирующее состояние [8].

Современные языки моделирования БП класса EML (Enterprise Modeling Languages) органично включают в себя средства моделирования па-

раллельных процессов, имеющие тот же уровень абстракции, что и все другие языковые средства. Так один из наиболее часто используемых языков данного класса BPMML, базирующийся на нотации BPMN [9], содержит такие конструкции как параллельное исполнение без синхронизации, параллельное исполнение с синхронизацией, координацию исполнения, а также ряд паттернов завершения, в совокупности реализующих соответствующие механизмы синхронизации.

В части создания новых языков для описания МКОД-параллелизма в [10] разработан язык моделирования конвейерных БП, описывающий процесс в виде ациклического ориентированного графа, вершинами которого могут быть функциональные операции или спусковые функции. Набор функциональных примитивов, позволяющих описывать конвейерные БП, включает такие конструкции как операция, линейный конвейер, функции мультиплицирования-редуцирования, конъюнкции-дизъюнкции, задачи-приема и др.

Необходимо отметить, что конструкции всех без исключения вышеперечисленных языков моделирования параллельных БП имеют значительно более низкий уровень абстракции по сравнению с широко используемыми диаграммными техниками, такими как IDEF0, DFD и другие, и уже не способны играть роль языков «для передачи понимания», фактически являясь классическими языками проектирования и находясь с языками первого типа в соотношении «ассемблер – язык высокого уровня».

Заметим, что для ОКМД-параллелизма (синхронных БП) в настоящее время отсутствуют какие-либо языковые средства его организации. Однако, в качестве основы такого параллелизма могут быть ис-

пользованы результаты теории программирования в части организации векторных и матричных структур данных и методов их размещения и обработки, а также языков программирования векторных вычислений [11]. Для решения данной задачи необходимо соответствующее расширение информационных моделей БП (в частности, ERD), что выходит за рамки настоящей статьи.

Методы распараллеливания БП, включая распараллеливание их линейных участков и повторяющихся конструкций, представляют несомненный интерес для решения задач, связанных с их реорганизацией. Эффективность применения таких методов может превысить эффект от «горизонтального и вертикального уплотнения» [4] – двух наиболее эффективных механизмов реинжиниринга бизнес-процессов по Хаммеру и Чампи.

Параллельные языки

Модель и языковые средства организации параллельных независимых ветвей (процессов)

В качестве основы организации асинхронного параллелизма используем общую модель бизнес-процесса, приведенную в [12], которая представляет собой следующий граф управления бизнес-функциями:

$$G(N, \{n_0\}, \{n_{\phi}\}, E, M, EM, EN, R, ER), \text{ где}$$

- N – множество узлов, каждый из которых соответствует бизнес-функции;
- $\{n_0\}$ и $\{n_{\phi}\}$ – множества входных и завершающих узлов, соответственно;
- E – множество управляющих ребер такое, что $\forall n_i, n_j \in N \cup \{n_0, n_{\phi}\}: (n_i, n_j) \in E$, если возможна ситуация, когда за выполнением бизнес-функции n_i будет выполняться бизнес-функция n_j ;

M – множество узлов, соответствующих структурным единицам предприятия (вплоть до роли конкретного исполнителя);

EM – множество ребер подчиненности такое, что $\forall m_i, m_j \in M: (m_i, m_j) \in EM$, если структурная единица m_i подчинена структурной единице m_j ;

EN – множество ребер исполнения бизнес-функции такое, что $\forall m_i \in M, n_j \in N: (m_i, n_j) \in EN$, если бизнес-функция n_j может быть выполнена в подразделении m_i ;

R – множество информационных объектов (ИО) предприятия;

ER – множество ребер использования ИО такое, что $\forall r_i \in R, n_j \in N: (r_i, n_j) \in ER$, если бизнес-функция n_j использует при своем выполнении ИО r_i или $(n_j, r_i) \in ER$, если бизнес-функция n_j определяет при своем выполнении ИО r_i .

Для описания сценария выполнения параллельного бизнес-процесса обозначим через P общее количество его исполнителей. Здесь и далее под исполнителями будем понимать нижний уровень подчиненности элементов множества M , которые пронумеруем от 1 до $m \leq |M|$, например, все инженеры получают номера от 1 до κ , все бухгалтера – от $\kappa + 1$ до n и т.д. Обозначим сценарий выполнения БП через $H_p = \{(n_i, P_i, t_i)\}$, где $n_i \in N$, P_i – номер исполнителя, t_i – время начала бизнес-функции.

Для уменьшения сложности задачи, не нарушая ее общности можно предположить, что

1) время выполнения любой бизнес-функции является одинаковым и равняется условной 1;

2) передача данных между ИО и использующими их функциями осуществляется без временных затрат.

Рассматриваемый сценарий H_p должен обладать следующими свойствами:

1) $\forall n_i, n_j \in N: t_i = t_j \Rightarrow P_i \neq P_j$, т.е. один и тот же исполнитель

не должен назначаться разным функциям в один и тот же момент времени;

2) $\forall (n_i, n_j) \in E \Rightarrow t_j \geq t_i + 1$, т.е. к назначаемому моменту выполнения функции предшествующая функция должна быть завершена;

По аналогии с [13] (где в качестве модели вычислений используется граф «операции-операнды») граф G бизнес-процесса вместе с множеством сценариев его выполнения $\{H_p\}$ может рассматриваться как модель БП, реализуемого P исполнителями – $A_p(G, H_p)$.

Время исполнения сценария определяется максимальным значением времени, применяемым в нем – $T_p(G, H_p) = \max_{n_i \in N} (t_i + 1)$. Минимальное время исполнения сценария задается как $T_p(G) = \min_{H_p} (G, H_p)$.

Для анализа максимально возможного параллелизма можно определить оценку наиболее быстрого исполнения (при использовании неограниченного количества исполнителей) – $T_\infty(G) = \min_{p \geq 1} (T_p)$. Оценка T_1 определяет время выполнения бизнес-процесса при использовании одного исполнителя (последовательный вариант решения задачи) – $T_1(G) = |N - \{n_0\}|$.

Ниже приведен ряд теорем из [14, 15], адаптированных для оценки времени выполнения бизнес-процесса.

Теорема 1. Минимально возможное время выполнения параллельного бизнес-процесса определяется длиной максимального его сценария, т.е. $T_\infty(G) = d(G)$, где d – диаметр (длина максимального пути) графа.

Теорема 2. Пусть для некоторого завершающего узла n_ϕ существует путь из каждого входного узла n_0 . Кроме того, пусть входная степень узлов графа (количество входящих ребер) не превышает 2. Тогда минимально возможное время выполнения бизнес-процесса ограничено снизу значением

$T_\infty(G) = \log_2(n)$, где n – количество входных узлов.

Теорема 3. При уменьшении количества исполнителей время выполнения бизнес-процесса увеличивается пропорционально коэффициенту уменьшения количества исполнителей, т.е. $\forall q = cp, 0 < c < 1 \Rightarrow T_q \leq cT_p$.

Теорема 4. Для любого количества исполнителей справедлива следующая верхняя оценка для времени выполнения параллельного бизнес-процесса: $\forall p \Rightarrow T_p < T_\infty + T_1/p$.

Теорема 5. Времени выполнения бизнес-процесса, которое сопоставимо с минимально возможным временем T_∞ , можно достичь при количестве исполнителей порядка $p \sim T_1/T_\infty$, а именно $p \geq T_1/T_\infty \Rightarrow T_p \leq 2T_\infty$. При меньшем количестве исполнителей время выполнения не может превышать более чем в 2 раза наилучшее время выполнения при имеющемся количестве исполнителей, т.е. $p < T_1/T_\infty \Rightarrow T_1/p \leq T_p \leq 2T_1/p$.

Приведенные теоремы позволяют дать следующие рекомендации по правилам построения параллельных бизнес-процессов:

1) При выборе сценария должен использоваться граф с минимально возможным диаметром (теорема 1).

2) Для параллельного выполнения бизнес-процесса целесообразное количество исполнителей определяется величиной $p \sim T_1/T_\infty$ (теорема 5).

3) Время выполнения параллельного сценария ограничивается сверху величинами, приведенными в теоремах 4 и 5.

Языковые средства описания асинхронных параллельных бизнес-процессов семантически эквивалентны соответствующим средствам для вычислительных процессов данного класса. С позиции синтаксиса они должны лишь отличаться от других графических символов используемого

визуального языка моделирования, а также быть включены в множество терминальных символов порождающей сценарии грамматики бизнес-процесса и присутствовать в ее порождающих правилах. Эти средства разделяются на две группы [5]:

1) Языковые средства организации параллельных независимых ветвей — объявления ветвей, их инициации и завершения, а также механизмы их взаимодействия. Каждая ветвь должна быть описана как независимый подпроцесс, имеющий специальные конструкции для указания возможности его одновременного выполнения с другими подпроцессами.

2) Языковые средства синхронизации доступа к разделяемым ресурсам (семафоры Дейкстры, условные критические интервалы, мониторы, сентинелы и др.)

В качестве примера реализации этого механизма в грамматику бизнес-процесса [7, 16] было включено специальное множество символов, определяющих параллелизм, в простейшем случае состоящее из двух символов — апострофа над левой частью порождающего правила, означающего начало параллельных ветвей, и пустого символа над левой частью порождающего правила, означающего линейный переход к правой части правила.

Язык моделирования конвейерных БП

Структура конвейерного процесса описывается параллельно-последовательной схемой. Ступени такого конвейера выполняются последовательно, а операции одной ступени параллельно. Разработанная на основе такой схемы теория [17] позволяет вычислять основные характеристики конвейерного процесса и составлять его расписание. Расширение данного формализованного класса процессов приведено в [10],

где описаны конвейерные процессы, определяемые рекурсивными функциями. Процесс описывается расписанием выполнения операций, в котором время завершения каждой операции вычисляется с помощью рекурсивной функции вида

$$f_k^n = R(f_{k-1}^n, f_k^p, \dots, f_k^q, t_n),$$

где

n — номер операции конвейера,

k — номер выполняемого цикла,

t_n — время выполнения операции с номером n ,

f_k^n — время завершения выполнения операции n на k -м цикле,

p, \dots, q — номера операций, непосредственно предшествующих операции с номером n ,

R — некоторая рекурсивная функция.

Модель конвейерного процесса [10] представляет собой ациклический ориентированный граф, в котором вершины ассоциируются с исполняемыми операциями или спусковыми функциями, определяющими порядок выполнения операций, а дуги определяют отношения предшествования. Основными элементами соответствующего графического языка (диаграммы конвейера) являются следующие объекты:

- линейная операция (аналог объекта «активность» в IDEF0), идентифицируемая уникальным именем и имеющая атрибут, определяющий ее продолжительность во времени;

- спусковая функция AND, интерпретируемая следующим образом: выполнение следующей за ней операции начинается после того как завершится выполнение двух предшествующих ей операций;

- функция мультиплицирования операций, определяющая, что после выполнения предшествующей ей операции последующая операция выполняется m раз;

- функция редуцирования операций, определяющая, что последующая операция начинается выполняться после того как m раз выполнится предшествующая ей операция;

- функция раздачи, имитирующая одноименный процесс;

- функция приема, имитирующая одноименный процесс и являющаяся обратной по отношению к функции раздачи.

В [10] показано, что в общем случае конвейерный процесс сначала находится в переходном режиме, а затем принимает стационарный режим, в котором представляет собой колебательный процесс, т.е. описывается периодической функцией. Для таких процессов возможно вычисление следующих характеристик: времени завершения нулевого цикла конвейера; номера цикла, с которого начинается стационарный процесс; времени начала стационарного процесса; частоты колебаний интервала конвейера в стационарном режиме; амплитуды изменения интервала конвейера в стационарном режиме. При этом вычисление характеристик осуществляется за время $o(n)$, где n — количество объектов диаграммы модели.

Методы распараллеливания БП

Как упоминалось выше, методы распараллеливания БП включают распараллеливание их линейных участков и повторяющихся конструкций (циклов).

Назовем узел $n_i \in N$ непосредственным предшественником узла $n_j \in N$, если $\exists(n_i, n_j) \in E$. Соответственно, n_i будет являться непосредственным последователем n_j .

Определим *линейный участок* на графе бизнес-процесса G как маршрут (сценарий) (n_k, \dots, n_l) , где

- начальный узел n_k имеет более одного непосредствен-

ного предшественника или у его непосредственного предшественника имеется более одного непосредственного последователя;

- конечный узел n_i имеет более одного непосредственного последователя или у его непосредственного предшественника имеется более одного непосредственного предшественника;

- остальные узлы имеют ровно по одному непосредственному предшественнику и последователю.

Будем считать циклом БП любую повторяющуюся последовательность его бизнес-функций. Определим *цикл* на графе бизнес-процесса G как маршрут (сценарий) (n_k, \dots, n_l) , если $\exists(n_l, n_k) \in E$.

Примерами циклов в БП являются процесс обработки входящей корреспонденции, где в качестве итерации выступает последовательность функций обработки каждого из писем, или процесс согласования договора, где в качестве итерации выступает последовательность функций согласования конкретного договора с участниками согласования (юристом, бухгалтером и др.).

Основной задачей при распараллеливании линейных участков и циклов является задача выявления информационных зависимостей между его бизнес-функциями. Внимание обращается на последовательность определений / использований ИО, проявляющихся,

например, через входящие и исходящие информационные потоки на DFD – диаграмме потоков данных. При этом, не нарушая общности, будем считать, что исследуемые ИО являются элементарными объектами.

Обозначим через I_i множество всех входных потоков бизнес-функции, а через O_i – множество всех ее выходных потоков. Тогда требования *информационной независимости* двух бизнес-функций могут быть записаны следующим образом:

$$I_j \cap O_i = \emptyset$$

$$I_i \cap O_j = \emptyset$$

$$O_i \cap O_j = \emptyset$$

Или в терминах графа $G \forall k, m$:

$$\{(r_k, n_j) \in ER\} \cap \{(n_i, r_m) \in ER\} = \emptyset$$

$$\{(r_k, n_i) \in ER\} \cap \{(n_j, r_m) \in ER\} = \emptyset$$

$$\{(n_i, r_k) \in ER\} \cap \{(n_j, r_m) \in ER\} = \emptyset$$

В свою очередь вводятся следующие виды информационной зависимости:

1) Если $\exists x \in R: x \in O_i \wedge x \in I_j$, то говорят, что между функциями i и j существует *потоковая* или *истинная зависимость*, что обозначается как $i\delta_j$.

2) Если $\exists x \in R: x \in I_i \wedge x \in O_j$, то говорят, что между функциями i и j существует *антизависимость* ($i\bar{\delta}_j$).

3) Если $\exists x \in R: x \in O_i \wedge x \in O_j$, то говорят, что между функциями i и j существует *зависимость по выходу* ($i\delta_j$).

4) Если $\exists x \in R: x \in I_i \wedge x \in I_j$, то говорят, что между функ-

циями i и j существует *зависимость по входу* ($i\delta_{aj}$).

Все зависимости в БП можно представить в виде графа зависимостей по данным, вершины которого соответствуют функциям БП, а ребра различных типов – соответствующим отношениям зависимости (при этом зависимость по входу не играет существенной роли в анализе зависимостей и может быть исключена из данного графа).

Собственно, исходя из этих требований и строятся алгоритмы распараллеливания линейных участков и циклов (например, в работах [5], [18]), при этом для циклов анализируется информационная зависимость между его итерациями. Примером такой зависимости является вышеприведенный процесс согласования договора в случае, когда договор проходит ролевое согласование несколько раз.

Следует отметить, что распараллеливание цикла может быть осуществлено как в сторону его конвейеризации, так и в сторону векторизации. Так процесс обработки входящей корреспонденции может быть реализован как в виде конвейера, где каждый сотрудник выполняет одну функцию при обработке писем (прием, регистрация и т.д.), так и в виде векторной конструкции, где каждый сотрудник выполняет все функции процесса по одному письму.

Литература

1. Калянов Г.Н. О теории бизнес-процессов // Программная инженерия. 2018. Т. 9. № 3. С. 99–109.
2. Калянов Г.Н. Теория бизнес-процессов: формальные модели и методы // Экономика, статистика и информатика. 2016. № 4. С. 19–21.
3. Калянов Г.Н. Модели и методы теории бизнес-процессов // Открытое образование. 2015. № 6. С. 4–9.
4. Hammer M., Champy J. Reengineering the Corporation: A Manifesto for Business Revolution. N.Y.: Harper-Collins, 1993.

5. Трахтенгерц Э.А. Программное обеспечение параллельных процессов. М.: Наука, 1987.
6. Хокни Р., Джессхоуп К. Параллельные ЭВМ. М.: Радио и связь, 1987.
7. Калянов Г.Н. Концептуальная модель DFD-технологии // Открытое образование. 2017. № 4. С. 21–26.
8. Громов Ю.Ю., Иванова О.Г., Беляев М.П., Данилкин С.В. Методы и средства проектирования информационных систем. Объектно-ориентированный подход. Тамбов, 2013.
9. Федоров И.Г. Моделирование бизнес-процессов в нотации BPMN 2.0. М.: МЭСИ, 2013.

10. Куприянов Б.В. Метод эффективного анализа модели рекурсивного конвейерного процесса // Автоматика и телемеханика. 2017. № 3. С. 63–79.

11. Разбегин В.П., Калянов Г.Н., Куприянов Б.В. Система программирования векторных вычислений // Программирование. 1985. № 4. С. 25–32.

12. Калянов Г.Н. Теория и практика реорганизации бизнес-процессов. М.: СИНТЕГ, 2000.

13. Гергель В.П. Теория и практика параллельных вычислений. М.: ИНТУИТ, 2007.

14. Bertsecal D.P., Tsitsiklis J.N. Paralleled and Distributed Computation. Numerical Methods. N.J.: Prentice Hall, 1989.

15. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002.

16. Калянов Г.Н., Куприянов Б.В., Лукинова О.В. Адаптация DFD-технологии при моделировании бизнес-систем в среде РДС // Открытое образование. 2019. № 23(2). С. 61–68.

17. Воеводин В.В. Математическая модель конвейерных вычислений. М.: Наука, 1982.

18. Векторизация программ: теория, методы, реализация. М.: Мир, 1991.

19. Карпов В.Е. Введение в распараллеливание алгоритмов и программ // Компьютерные

исследования и моделирование. 2010. Т. 2. № 3. С. 231–272.

20. Марлей В.Е. и др. Автоматизация распараллеливания программ на основе информационных связей // Программные продукты и системы. 2005. № 1. С. 2–6.

21. Jordan H. F., Alaghand F. Fundamentals of Parallel Processing. Pearson Education Inc., Upper Saddle River, NJ 07452, 2003.

22. Wilkinson B., Allen M. Parallel programming techniques and applications using networked workstations and parallel computers. Pearson Education Inc., Upper Saddle River, NJ 07452, 2005.

23. Афанасьев К.Е. Многопроцессорные вычислительные системы и параллельное программирование. Кемерово: Кузбассвузиздат, 2003.

24. Немнюгин С.А., Стесик О.Л. Параллельное программирование для многопроцессорных вычислительных систем. СПб.: БХВ-Петербург, 2002.

25. Костенко В.А. К вопросу об оценке оптимальной степени параллелизма // Программирование. 1995. № 4. С. 24–28.

26. Бочаров Н.В. Технологии и техника параллельного программирования // Программирование. 2003. № 1. С. 5–23.

References

1. Kalyanov G.N. On the theory of business processes. Programmynaya inzheneriya = Software engineering. 2018; 9; 3: 99–109. (In Russ.)

2. Kalyanov G.N. Business Process Theory: Formal Models and Methods. Ekonomika, statistika i informatika = Economics, Statistics and Informatics. 2016; 4: 19–21. (In Russ.)

3. Kalyanov G.N. Models and methods of the theory of business processes. Otkrytoye obrazovaniye = Open education. 2015; 6: 4–9. (In Russ.)

4. Hammer M., Champy J. Reengineering the Corporation: A Manifesto for Business Revolution. New York: Harper-Collins; 1993.

5. Trakhtengerts E.A. Programmnoye obespecheniye parallel'nykh protsessov = Parallel process software. Moscow: Nauka; 1987. (In Russ.)

6. Khokni R., Dzhesskhoup K. Parallel'nyye EVM = Parallel computers. Moscow: Radio and communication; 1987. (In Russ.)

7. Kalyanov G.N. Conceptual model of DFD technology. Otkrytoye obrazovaniye = Open education. 2017; 4: 21–26. (In Russ.)

8. Gromov YU.YU., Ivanova O.G., Belyayev M.P., Danilkin S.V. Metody i sredstva proyektirovaniya informatsionnykh sistem. Ob'yektno-oriyentirovanny podkhod = Methods and tools for designing information systems. Object oriented approach. Tambov; 2013. (In Russ.)

9. Fedorov I.G. Modelirovaniye biznes-protsessov v notatsii BPMN 2.0 = Modeling business processes

in BPMN 2.0 notation. Moscow: MESI; 2013. (In Russ.)

10. Kupriyanov B.V. A method for effective analysis of a model of a recursive conveyor process. Avtomatika i telemekhanika = Automation and Remote Control. 2017; 3: 63–79. (In Russ.)

11. Razbegin V.P., Kalyanov G.N., Kupriyanov B.V. Vector computing programming system. Programirovaniye = Programming. 1985; 4: 25–32. (In Russ.)

12. Kalyanov G.N. Teoriya i praktika reorganizatsii biznes-protsessov = Theory and practice of business process reorganization. Moscow: SINTEG; 2000. (In Russ.)

13. Gergel' V.P. Teoriya i praktika parallel'nykh vychisleniy = Theory and practice of parallel computing. Moscow: INTUIT; 2007. (In Russ.)

14. Bertsecal D.P., Tsitsiklis J.N. Paralleled and Distributed Computation. Numerical Methods. New Jersey: Prentice Hall; 1989.

15. Voyevodin V.V., Voyevodin V.I.V. Parallel'nyye vychisleniya = Parallel computing. Saint Petersburg: BHV-Petersburg; 2002. (In Russ.)

16. Kalyanov G.N., Kupriyanov B.V., Lukinova O.V. Adaptation of DFD technology in modeling business systems in the RDS environment. Otkrytoye obrazovaniye = Open Education. 2019; 23(2): 61–68. (In Russ.)

17. Voyevodin V.V. Matematicheskaya model' konveyernykh vychisleniy = Mathematical model of pipeline calculations. Moscow: Nauka; 1982. (In Russ.)

18. Vektorizatsiya programm: teoriya, metody, realizatsiya = Vectorization of programs: theory, methods, implementation. Moscow: Mir; 1991. (In Russ.)

19. Karpov V.Ye. Introduction to Parallelization of Algorithms and Programs. Komp'yuternyye issledovaniya i modelirovaniye = Computer Research and Modeling. 2010; 2; 3: 231–272. (In Russ.)

20. Marley V.Ye. et al. Automation of parallelization of programs based on information links. Programmnyye produkty i sistemy = Software products and systems. 2005; 1: 2-6. (In Russ.)

21. Jordan H. F., Alagband F. Fundamentals of Parallel Processing. Pearson Education Inc., Upper Saddle River, NJ 07452, 2003.

22. Wilkinson B., Allen M. Parallel programming techniques and applications using networked workstations and parallel computers. Pearson

Education Inc., Upper Saddle River, NJ 07452, 2005.

23. Afanas'yev K.Ye. Mnogoprotsessornyye vychislitel'nyye sistemy i parallel'noye programmirovaniye = Multiprocessor computing systems and parallel programming. Kemerovo: Kuzbassvuzizdat; 2003. (In Russ.)

24. Nemnyugin S.A., Stesik O.L. Parallel'noye programmirovaniye dlya mnogoprotsessornykh vychislitel'nykh system = Parallel programming for multiprocessor computing systems. Saint Petersburg: BHV-Petersburg; 2002. (In Russ.)

25. Kostenko V.A. On the question of evaluating the optimal degree of parallelism. Programmirovaniye = Programming. 1995; 4: 24–28. (In Russ.)

26. Bocharov N.V. Technologies and techniques of parallel programming. Programmirovaniye = Programming. 2003; 1: 5–23. (In Russ.)

Сведения об авторе

Георгий Николаевич Калянов

Д.т.н., профессор, г.н.с.

Институт проблем управления РАН,

Москва, Россия

Эл. почта: Kalyanov@mail.ru

Information about the author

Georgiy N. Kalyanov

Dr. Sci. (Engineering), Professor, Chief Researcher

Institute of Control Sciences of Russian Academy of

Sciences Moscow, Russia

E-mail: Kalyanov@mail.ru