

Е.А. Титенко, И.Е. Чернецкая, М.А. Титенко, Э.В. Мельник, Д.А. Трокоз

РАСШИРЕННАЯ ПРОДУКЦИОННАЯ МАШИНА ВЫВОДА ДЛЯ РЕАЛИЗАЦИИ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Актуальность. В работе развивается теоретический подход организации параллельных вычислений на основе производственной модели управления потоком данных. Производственная парадигма параллельных вычислений имеет необходимые условия построения новых архитектур и организации высокопроизводительных параллельных вычислений. Рассматриваются производственные системы, управляющие наборами левых частей производств (образцами). **Цель** – повышение эффективности параллельного вывода решений за счет сокращения непродуктивных затрат времени на перебор возможных альтернатив в пространстве графа вывода. Метод решения основан на создании расширенной машины символического вывода для реализации параллельных вычислений. Машина символического вывода – это абстрактная система, систематизирующая производственный вывод как последовательность четырех вычислительно-поисковых этапов. Машина вывода задает общий вид однородной вычислительной системы. Главное отличие – декомпозиция базы производственных правил на отдельные подмножества на основе алгебры производств и структуризации отношений между производствами. Вместо единой «плоской» структуры предлагается базу производств декомпозировать на части – ввести систему независимых подмножеств производств. Параллельный вывод реализуется по отдельным подмножествам без потери общности, при этом перебор возможных альтернатив является сокращенным. Каждое подмножество производств имеет специальное слово-маркер, по значению которого активизируется только одно подмножество производств. Оно загружается в операционную часть однородной вычислительной системы для параллельного исполнения. **Результаты.** Показано, что количественные оценки сокращения времени вывода зависят от общего числа производств, количества образуемых подмножеств и их размера. Моделирование показало, что даже простейшая декомпозиция на два подмножества (одно подмножество состоит из 2-х производств) дает временной выигрыш (1,07-1,52) раз, пропорциональный общему числу производств. **Выводы.** Построенная расширенная машина символического вывода является основой для последующего создания архитектуры однородной вычислительной системы с комбинацией централизованного и локального управления, что позволяет вычислительным блокам однородной операционной части параллельно работать без избыточного обращения к общей памяти.

Производственная система; параллельные вычисления; независимые подмножества производств; конфликтные слова.

Е.А. Титенко, И.Е. Чернецкая, М.А. Титенко, Э.В. Мельник, Д.А. Трокоз

ADVANCED PRODUCTION OUTPUT ENGINE FOR IMPLEMENTING PARALLEL COMPUTING

Relevance. The paper discusses a theoretical approach to organizing parallel computing based on a production model of data flow control. The production paradigm of parallel computing has the necessary conditions for building new architectures and organizing high-performance parallel computing. We consider production (mathematical) systems that control sets of left-hand sides of productions (samples). The goal is to increase the efficiency of parallel inference of solutions by reducing unproductive time spent searching through possible alternatives in the inference graph space. The research is based on the creation of an extended symbolic computation machine for implementing parallel steps. A symbolic computing machine is an abstract system that systematizes production output as a sequence of four computational and search stages. The inference engine defines the general appearance of a homogeneous computing system. The main difference is the decomposition of the base of production rules into separate subsets based on the algebra of production and the structuring of relations between products. Instead of a single “flat” structure, it is proposed to decompose the product base into parts - to introduce a system of independent subsets of products. Parallel inference is implemented for individual subsets without loss of generality, while the search for possible alternatives is reduced. Each subset of productions has a special marker word, the value of which activates only one subset of productions. It is loaded into the operating part of a homogeneous computing system for parallel execution. **Results.** It is shown that quantitative estimates of the reduction in output time depend on the total number of productions, the number of subsets formed and their size. Simulation has shown that even the simplest decomposition into two subsets (one

subset consists of 2 productions) gives a time gain of (1.07-1.52) times, proportional to the total number of productions. Conclusions. The created extended symbolic computing machine is the basis for the subsequent creation of the architecture of a homogeneous computing system with a combination of centralized and local control. This property allows computational units of a homogeneous operating part to work in parallel without excessive access to shared memory.

Production system; parallel computing; independent subsets of production; conflict words.

Введение. Перспективные однородные вычислительные системы (ОВС) в значительной степени ориентируются на новый класс проблемно-поисковых задач, связанных с выполнением расчетно-поисковых, переборных, оптимизационных вычислений [1–4]. Значимыми примерами таких задач являются задачи вычислительной химии, физики, лингвистики, алгебраической комбинаторики слов, символьного кодирования/декодирования временных рядов, организации вычислений на графах, обеспечения реконфигурации многопроцессорных систем, вычислительно трудоемкие задачи структуризации Big Data и другие проблемно-поисковые задачи и модельные задачи, использующие модели и технологии искусственного интеллекта [5–9]. Их отличительная особенность – ориентация на обработку высокоуровневой информации (числовые, символьные, реляционные, смысловые данные), комбинация числовых и символьных преобразований, необходимость генерации множества вариантов для последующего отбора наиболее подходящих из них. Другая отличительная особенность таких задач – существенная доля временных затрат операций обращения к памяти в общем объеме вычислений, нерегулярный характер обращений к ячейкам памяти, что не позволяет использовать конвейеризацию аппаратных схем декодирования адресов, принятую при обработке массивов числовых данных.

Решение проблемно-поисковых задач преимущественно имеет недетерминированный характер. Вычислительно-поисковые действия связаны с генерацией, анализом, перебором множества промежуточных вариантов, отбором наиболее приоритетных путей в пространстве альтернатив и их проверкой на предмет совпадения с желаемыми или эталонными значениями. Такие вычисления традиционно выполняются в рамках моделей обработки знаний (логическая, фреймовая, продукционная модели, модель семантических сетей и др.) [10–13].

Традиционный путь разработки однородных ВС для таких задач считается экстенсивным. Он основан на методе «грубой силы»: введение структурной избыточности в архитектуру ОВС, повышение производительности за счет повышения тактовой частоты процессоров и устройств, распараллеливание преимущественно потока команд, создание системы макрокоманд, выполняющих крупноблочные числовые операции. Тем не менее используемая модель вычислений не имеет связи со структурой ОВС и функциональными блоками в ее составе. Как следствие, возникает семантический разрыв, порождающий набор избыточных вычислительных, коммуникационных, пересылочных операций, операций перезаписи данных между блоками памяти вместо их содержательной обработки. Как правило, такие операции не меняют структуру обрабатываемых данных, но входят в общее время решения задачи. Данный семантический разрыв позволяет достигать ОВС лишь около 10-15% от пиковой производительности на задачах реального уровня сложности [14]. Особенно ощутимы временные потери на задачах с преобладанием существенной доли обменных операций, что характерно для задач искусственного интеллекта (ИИ).

В противовес экстенсивному пути однородные ОВС целесообразно развивать на основе «сквозной» технологии, совмещающей методы и средства проектирования и реализации параллельных вычислений. «Сквозная» технология вычислений основана на преемственности математических, алгоритмических, архитектурных и схемотехнических решений в составе ОВС. Она является своеобразным «интеллектуальным мостиком» между теоретическими и аппаратно-программными средствами параллельных вычислений. Интеллектуализация вычислений является эффективным инструментом повышения производительности ОВС для проблемно-поисковых задач реального уровня сложности (количество анализируемых элементов $\approx 10^{14}$ - 10^{15}) за счет извлечения дополнительной информации о свойствах задачи и ходе вычислительного процесса.

В общем случае, под интеллектуализацией вычислений подразумеваются процессы автоматической (автоматизированной):

- ◆ адаптации структуры ОВС к информационному графу задачи;
- ◆ выполнения вычислительного процесса по мере готовности данных;
- ◆ эволюционного изменения алгоритма работы в зависимости от достигнутых состояний;
- ◆ аппаратной реализации крупноблочных символьных операций;
- ◆ динамической реконфигурации ОВС в зависимости от информационного графа задачи;
- ◆ выделения и накопления типовых (шаблонных) описаний и их последующего применения и др.

Модели и методы обработки знаний являются основой интеллектуализации вычислений в перспективных однородных ВС с параллельной обработкой высокоуровневой информации [10, 12, 15, 16].

Продукционные системы (ПС) – это общепризнанный и достаточно исследованный математический аппарат, использующий вычислительную схему «условие – реакция» для управления вычислительным процессом. Целесообразность выбора ПС для повышения производительности работы современных ОВС определяется следующими свойствами:

- ◆ однородность правил;
- ◆ легкость расширения базы продукций;
- ◆ гибкость реализации различных схем управления;
- ◆ естественный параллелизм исполнения набора продукций на основе потока данных (левых частей продукций);
- ◆ преемственность математических и алгоритмических операций;
- ◆ возможность расширения структуры правила типовыми числовыми обработчиками.

ПС предназначены для обработки символьной информации (ОСИ), под которой понимаются вычисления от словесного и выше уровня абстракции данных [17, 18]. Формальной основой ОСИ является алгебра продукций. Она позволяет исследовать соотношения между строковыми операндами, выполнять эквивалентные преобразования продукций, декомпозировать систему продукций, вести параллельные вычисления на основе стратегий выводов. Главное преимущество таких продукционных систем – это естественная ориентация на управление потоком данных, т.е. на параллельное исполнение подмножества правил (продукций) по мере их готовности. Условие готовности для продукций из ПС считается унифицированным – активация продукции, т.е. положительный результат поиска по образцу в описании внешней среды. Семантическая связь ПС с ОВС реализуется путем отображения продукций на вычислительные блоки, аппаратно реализующие базовые операции над строковыми данными. В этом случае создаются необходимые условия для параллельного исполнения множества продукций на основе принципа управления потоком данных.

Таким образом, актуальность исследования определяется применением продукционных моделей и стратегий параллельного вывода для их последующей аппаратной реализации на ОВС.

Материалы и методы. Пусть заданы $n \in \mathbb{N}$, рабочий алфавит A , метасимволы $\rightarrow \notin A$, $* \notin A$. Для описания продукционных систем принято вводить расширение M над рабочим алфавитом A . Расширение необходимо, чтобы выделять служебные символы и особые (предметные) свойства в данных и шагах работы продукционной системы

$$B = \{ A, \rightarrow, \bullet, \xi, \eta, \dots, \mu, \alpha, \beta, \dots, \delta, \nabla, \diamond \dots \}, \quad (1)$$

где ξ, η, \dots, μ – алфавитные (символьные) переменные, $\alpha, \beta, \dots, \delta$ – переменные-челноки, $\nabla, \diamond \dots$ – переменные-указатели.

Объекты $O_i, P_i \in A^*$ – слова, составленные по правилам конкатенации, где $i=1-n$.

Пусть задано обрабатываемое слово $S \in A^*$.

Продукция – это математическое выражение в алфавите M вида [12]

$$O \xrightarrow{\Theta} P, \quad (1)$$

где O – левая часть (слово-условие) в алфавите B ; P – правая часть (слово-реакция) в алфавите B , Θ – индикатор типа продукции (текущая или заключительная).

Тогда продукционная система (*Rule System Based – RSB*) задается как система-тройка:

$$RSB = \{ B, \Theta, C \}, \quad (2)$$

где C – стратегия (схема) вывода, Θ – определяющее множество продукций, управляемых множеством образцов продукций (поток данных):

$$\Theta : \begin{cases} O_1 \rightarrow P_1 \\ \dots \\ O_i \rightarrow P_i \\ \dots \\ O_n \rightarrow P_n \end{cases} \quad (3)$$

Содержательно ПС представляется множеством продукций, объединенных стратегией вывода C , которая задает количество и порядок срабатывания продукций на каждом шаге вывода. Согласно принципу управления потоком данных шаг вывода состоит в срабатывании подмножества продукций (активированные продукции) и формировании набора промежуточных слов. ПС завершает работу на основе двух условий:

- ◆ срабатывание заключительной продукции;
- ◆ отсутствие вхождения левых частей продукций в слово S .

Известно, что в архитектурах и машинах, управляемых потоком данных (*DataFlows Machine*) [15], используется принцип потенциального распараллеливания вычислений по данным. Это означает, что операционная часть однородной ВС содержит множество однотипных вычислительных блоков, реализующих все базовые операции. Тем не менее лишь некоторое подмножество продукций из выражения (3) будет потенциально активно для вывода на текущем ярусе графа вывода. Соответственно из операционной части ОВС необходимо динамически выбирать только такие продукции, число которых изменяется по шагам вывода. В результате, в операционной части однородной ВС возникают конфликты на размещение подмножеств продукций.

Для снижения ресурсных затрат однородной ВС на работу ПС (объем памяти, время перебора продукций) предлагается ее декомпозиция на обособленные подмножества, выделяемые на основе операций алгебры продукций [19]. В этом случае работа ПС выполняется по каждому подмножеству с сохранением корректности конечного результата.

Декомпозиция ПС включает операции конструктивной логики пересечения, объединения, дополнения слов и др. Эти операции попарно устанавливают свойства структурной связи между образцами и подстановками продукций вида (3) [12]. Над ПС выполняются подготовительные вычислительные действия, позволяющие исследовать соотношения между продукциями и выполнить разбиение их на t независимых подмножеств D_1, D_2, \dots, D_t , обеспечивающие параллельный вывод по каждому подмножеству отдельно.

Тем не менее характер подготовительных действий имеет комбинаторную сложность. Количество проверок на парные пересечения левых и правых частей из выражения (3) имеет квадратичную зависимость, что при больших n или частых изменениях состава ПС является достаточно критичным для эффективности работы ПС.

Обеспечение параллельной работы ПС вида (3) обеспечивается поиском такого подмножества продукций D_y ($y \in 1 \dots t$), которое имеет максимальный размер. Соответственно целевая функция подготовительных действий имеет вид

$$f\{\Theta\} = D, |D| \rightarrow \max. \quad (4)$$

Построенное подмножество максимального размера позволяет задать в операционной части ОВС минимально необходимое число функциональных блоков, реализующих продукции, и обеспечить тем самым параллельное исполнение множества путей на текущем уровне графа вывода.

С точки зрения теоретического проектирования параллельный вывод в системе продукции описывается системой вида [20]

$$\{R, \Theta, I_M\}, \quad (5)$$

где R – исходные данные в алфавите M ; Θ – определяющее множество продукции (база продукции); I_M – машина вывода.

Машина вывода – это абстрактная система, имеющая циклическую вычислительную схему на основе управления потоком данных (активированных продукции). Будучи впервые созданной для проектирования экспертных систем, машина вывода в дальнейшем стала инструментом для проектирования и программирования однородных ВС под класс задач на основе продукционной парадигмы вычислений [21, 22].

Основу работы машины вывода составляет вычислительный цикл, содержащий четыре вычислительных этапа: выборка данных, активация подмножества продукционных правил, разрешение конфликта и срабатывания продукции, изменения состояний внешней рабочей памяти R .

Теоретико-множественное описание машины вывода I_M имеет вид

$$I_M = \{V, SMP, K, W\}, \quad (6)$$

где V – блок выборки из R и B множества активных данных и множества активных продукции; SMP – блок сопоставления, определяющий множество вхождений образов продукции во входные данные; K – блок разрешение конфликтов; W – блок выполнение выбранных продукции.

Структура стандартной машины вывода I_M представлена на рис. 1 в виде циклической последовательности 4-х вычислительных-поисковых блоков, собственно базы продукции и рабочей памяти для записи исходных данных и результатов вывода.

Цикл работы машины вывода состоит в последовательном выполнении выше обозначенных модулей: 1 – выборка, 2 – сопоставление, 3 – разрешение конфликтов, 4 – выполнения продукции и модификация рабочей памяти.

Использование продукционной модели приводит к управлению данными (образцами активированных продукции) в общем цикле работы. Машина вывода на каждом цикле формально порождает новые слова (промежуточные или конечные решения), которые записываются в рабочую память. Циклы вывода повторяются до тех пор, пока не будет получен целевой результат или ПС будет не применима к обрабатываемым словам.

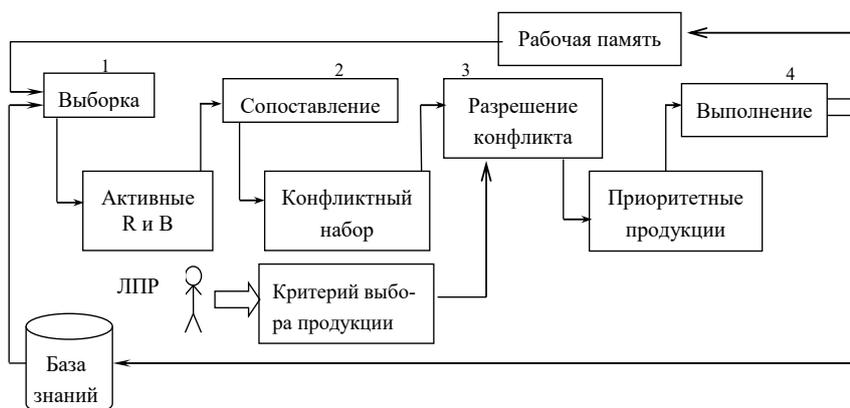


Рис. 1. Структура стандартной машины вывода

Разнообразие алгоритмических, технических вариантов реализаций блоков выборки, сопоставления, разрешения конфликтов и выполнения определяет стратегию вывода решений. При этом блоки сопоставления и разрешения конфликтов являются определяющими во временных затратах работы машины вывода.

Недостаток машины вывода со стандартной структурой заключается в последовательном характере вычислений как по блокам, так и внутри блоков, а формируемый при этом граф вывода, понимается как лабиринт альтернативных путей. Разрешение конфликта определяется лицом, принимающим решения (ЛПР), директивно. Выбор ЛПР приоритетных продуктов из конфликтного набора преимущественно проводится в рамках имеющихся вычислительных ресурсов и не зависит от машины вывода. Он проводится без содержательного анализа множества структурных пересечений образцов между собой, что является системным ограничением для обеспечения параллельного вывода.

Для систем продуктов с недетерминированным характером их срабатывания на этапе сопоставления, в общем случае, формируется некоторое подмножество активированных продуктов и активных данных, готовых к применению и модификации соответственно. Тем не менее, блок разрешения конфликта, в основном, осуществляет выбор из конфликтного множества единственной приоритетной продукции для выполнения. Остальные активированные продукты и активные данные будут рассмотрены на следующем шаге вывода или отложены по исполнению.

Сущность предлагаемого подхода к реализации недетерминизма в работе машины вывода заключается в понимании графа вывода не как лабиринта, а как сетевого пространства равных возможностей, в котором все пути в пределах текущего уровня могут быть сгенерированы параллельно.

Для формализации данных рассуждений выполняется модификация машины вывода путем добавления в ее состав новых блоков: блока синтеза конфликтных слов и блока обеспечения параллельных выводов.

Блок синтеза конфликтных слов основан на циклическом анализе парных пересечений образцов системы продуктов (3) и полученных конфликтных слов K . Он обеспечивает формирование полного списка слов, для которых складывается конфликтная ситуация выбора. Проверка на конечность такого списка слов является логическим индикатором возможности параллельного вывода. Эту работу выполняет блок 5 обеспечения параллельных выводов.

Основу формирования списка конфликтных слов составляют конструктивные дизъюнкции вида

$$\left\{ \begin{array}{l} (O_i^H = O_j^K) \vee (O_i^K = O_i^H) \vee (O_i \subset O_j) \vee (O_j \subset O_i) \\ (K_{i1}^H = K_{j1}^K) \vee (K_{i1}^K = K_{i1}^H) \vee (K_{i1} \subset K_{j1}) \vee (K_{j1} \subset K_{i1}) \end{array} \right. , \quad (7)$$

где O_i^H, O_i^K – собственные начало и окончания образца O_i ; O_j^H, O_j^K – собственные начало и окончания образца O_j ; K_{i1}^H, K_{i1}^K – собственные начало и окончания $i1$ -го конфликтного K_{i1} ; K_{j1}^H, K_{j1}^K – собственные начало и окончания $j1$ -го конфликтного K_{j1} ; $i \neq j$, ($i=1 \dots n, j=1 \dots n$), $i1 \neq j1$, ($i=1 \dots t, j=1 \dots t$).

Истинность конструктивных дизъюнций (7) позволяет структурировать продукции из базы, разбить их на обособленные подмножества.

Блок 5 обеспечения параллельных выводов представлен набором параллельных стратегий: И- параллельный вывод, ИЛИ- параллельный вывод и др. [23, 24]. Данные стратегии обеспечивают эффективное движение в сетевом пространстве поиска на основе анализа текущей конфликтной ситуации, состава обрабатываемых слов в рабочей памяти, что создает предпосылки для динамического выбора наиболее эффективной стратегии.

На рис. 2 представлена структура модифицированной машины вывода. Практическая ценность данной машины вывода определяется тем, что вычисления в блоке синтеза конфликтных слов осуществляются однократно и в дальнейшем используются параллельными стратегиями для параллельной генерации текущих состояний.

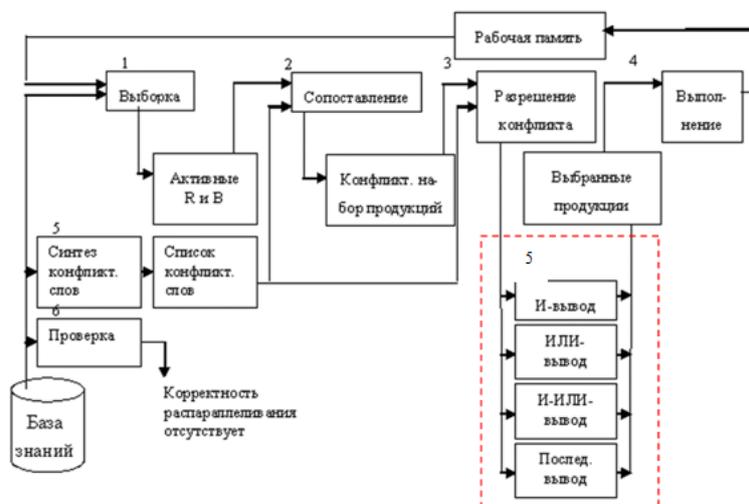


Рис. 2. Структура модифицированной машины вывода

Новизна модифицированной машины вывода заключается в формировании множества конфликтных слов на основе анализа пересечений пар образцов продукции и выборе приоритетных продукции из конфликтного множества без участия оператора (ЛПР). Комбинаторно-вычислительный процесс формирования множества конфликтных слов позволит сократить число возвратных переходов и тем самым сократить время построения графа вывода.

Дополнительные блоки проверки и синтеза конфликтных слов собственно в цикл работы модифицированной машины вывода не входят, поскольку обеспечивают необходимые вычисления однократно без использования обрабатываемых данных из рабочей памяти.

Вместе с тем модифицированная машина вывода имеет «плоскую» структуру базы продукции. Это означает, что в каждом цикле работы машины вывода на этапе сопоставления анализируются все продукции из базы, хотя фактически потенциально активным будет лишь некоторое подмножество продукции.

Для сокращения временных затрат до этапа сопоставления обрабатываемых слов из рабочей памяти и продукции из базы продукции выполняется анализ и декомпозиция базы продукции на независимые подмножества. Рис. 3 демонстрирует структуру расширенной машины вывода.

Теоретической основой декомпозиции базы продукции является конструктивные дизъюнкции вида (7). На основе пар номеров продукции, имеющих непустое пересечение, составляется список так называемых связанных продукции, составляющих подмножество D продукции и имеющих общий дескриптор, идентифицирующий данное подмножество. Этим дескриптором является расширенное конфликтное слово, построенное на основе конструктивных дизъюнкций (7).

Коммутативность пересечений образцов продукции по (7) позволяет разбить исходную базу на t независимых подмножеств D_1, D_2, \dots, D_t , что обеспечит в дальнейшем выполнение этапа сопоставления только по этим подмножествам D_y , ($y=1 \dots t$).

Следующая возможность повышения гибкости стратегии вывода для ПС – расширительная модификация состава продукции. Известный подход [24] состоит во введении пред- и постусловий применения продукции. В отличие от традиционного подхода в предлагаемых производственных системах для уменьшения количества рассматриваемых вариантов вводятся не условия, а временные и количественные оценки самих продукции. Эта информация позволяет ранжировать продукции и при неопределённости выбора руководствоваться только наиболее приоритетной продукцией.

В качестве таких оценок в состав продукции вводятся следующие показатели:

- ◆ номер продукции (z_0);
- ◆ частота активации продукции (z_1);
- ◆ временной отсчет последнего срабатывания продукции (z_2);
- ◆ частота попадания в конфликтный набор продукции (z_3);
- ◆ длина левой части (z_4);
- ◆ длина правой части (z_5);
- ◆ количество переменных в левой части (z_6);
- ◆ приоритет заключения (z_7);
- ◆ условие модификации левой части (z_8);
- ◆ условие модификации право части (z_9);

Тогда продукция вида (2) приобретает вид

$$z_0 (z_1, \dots, z_6) O \rightarrow P(z_7, \dots, z_9). \quad (8)$$

Новизна модификации структуры продукции вида (8) связана с введением в ее состав элементов, инвариантных к типу решаемых задач, не только учитывающих предысторию состояний в графе вывода, но и условия модификации самих продукции в процессе выполнения, что уменьшает размер конфликтного множества продукции и повышает тем самым временную эффективность вывода решений в ПС

В итоге, расширенная машина вывода (рис. 3) представляет собой объект-семерку вида:

$$I_M = \{D, CW, AR, V, CMP, K, W\}, \quad (6)$$

где D – конфигуратор для декомпозиции на независимые подмножества продукции (слоты);

CW – набор конфликтных слов, маркирующих независимые подмножества продукции;

AR – арбитр, осуществляющий выбор подмножества продукции и стратегии параллельного вывода.

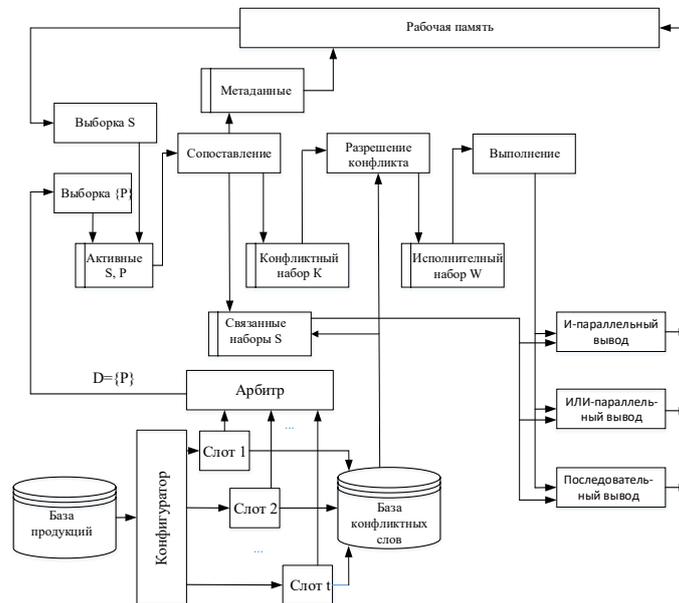


Рис. 3. Структура расширенной машина вывода

Результаты и их обсуждение. В качестве анализируемых ПС рассматриваются типовые системы продукционной обработки символьных данных: удвоение слова, обращения слова [25].

Пусть в рабочем алфавите A задано слово S в виде $S = \xi_1 \dots \xi_i \xi_{i+1} \dots \xi_L$, где $\xi_i \in A$ ($i=1-L$), $L=|S|$ – длина слова.

Тогда результат работы алгоритма U удвоения слова представляется как

$$U(S) = SS. \tag{9}$$

При удвоении S в качестве служебных метасимволов выделяются:

- ♦ метасимвол-челнок α , реализующий движение и выделяющий конечную область обработки рядом с челноком;

- ♦ признак удвоения текущей буквы ξ слова S – метасимвол-маркер β .

Определяющее множество Θ имеет вид

$$\left\{ \begin{array}{l} \alpha\xi \rightarrow \xi\beta\xi\alpha \\ \beta\xi\eta \rightarrow \eta\beta\xi \\ \beta \rightarrow \lambda \\ \alpha \rightarrow \bullet\lambda \\ \lambda \rightarrow \alpha \end{array} \right., \tag{10}$$

где λ – обозначение пустого символа (слова).

Первая продукция из (10) выполняет удвоение текущей буквы из S , расположенной справа от челнока α . Копия буквы ξ помечается маркером β , располагаемым слева от нее. Вторая продукция (10) выполняет движение копии буквы ξ вправо, реализуя тем самым посимвольное удвоение слова. Третья продукция из (10) удаляет растиражированные по обрабатываемому слову маркеры β . Четвертая продукция (10) заключительная, она удаляет челнок α , формируя тем самым окончательный результат и завершая работу алгоритма удвоения. Пятая продукция (10) срабатывает первой, она присоединяет слева к слову S челнок α , иницируя тем самым этап алгоритма посимвольного удвоения. Этап посимвольного удвоения реализуется собственно первой продукцией (10), которая срабатывает L раз.

Трассировка работы алгоритма для $L=3$ приведена ниже в табл. 1. Анализ системы продукций удвоения слова на пересечения по (7) показывает, что продукции первая и вторая продукции из (10) имеют общую часть. Это означает, что система продукций (10) может быть декомпозирована на 2 отдельных подмножества:

- $D_1 = \{1, 2\}$;
- $D_2 = \{3, 4, 5\}$.

Далее вычислительный процесс может быть реализован отдельно по каждому из подмножеств продукций, что существенно сократит количество холостых шагов на перебор продукций в каждом из подмножеств. Трассировочная табл. 1 подтверждает, что продукции из подмножества $D_1 = \{1, 2\}$ составляют единую последовательность, не имеющую вставок из подмножества D_2 .

Таблица 1

Трассировка шагов алгоритма $U(S) = SS$

Обрабатываемое слово	Номер сработавшей продукции	Номер шага ПС	Примечание
Processed word	Number of triggered production	Step number PS	Note
$\xi_1 \xi_2 \xi_3$	5	1	Исходное слово
$\alpha \xi_1 \xi_2 \xi_3$	1	2	
$\xi_1 \beta \xi_1 \alpha \xi_2 \xi_3$	1	3	
$\xi_1 \beta \xi_1 \xi_2 \beta \xi_2 \alpha \xi_3$	1	4	

$\xi_1 \beta \xi_1 \xi_2 \beta \xi_2 \xi_3 \beta \xi_3 \alpha$	2	5	Движение вправо
$\xi_1 \xi_2 \beta \xi_1 \beta \xi_2 \xi_3 \beta \xi_3 \alpha$	2	6	
$\xi_1 \xi_2 \beta \xi_1 \xi_3 \beta \xi_2 \beta \xi_3 \alpha$	2	7	
$\xi_1 \xi_2 \xi_3 \beta \xi_1 \beta \xi_2 \beta \xi_3 \alpha$	3	8	Удаление маркера
$\xi_1 \xi_2 \xi_3 \xi_1 \beta \xi_2 \beta \xi_3 \alpha$	3	9	
$\xi_1 \xi_2 \xi_3 \xi_1 \xi_2 \beta \xi_3 \alpha$	3	10	Удаление челнока
$\xi_1 \xi_2 \xi_3 \xi_1 \xi_2 \xi_3 \alpha$	4	11	
$\xi_1 \xi_2 \xi_3 \xi_1 \xi_2 \xi_3$	-		STOP

Известен оригинальный алгоритм обращения, предложенный А.А. Марковым [25]. Оригинальным алгоритмическим решением является отсутствие внешнего маркера конца слова. Для прекращения фазы движения символов слова вправо А.А. Марков ввел внутренний маркер, обозначаемый удвоенным челноком $\alpha\alpha$ как признак обработки всех символов слова S .

Основу ПС обращения слова составляют продукции

$$\alpha\xi\eta \rightarrow \eta\alpha\xi,$$

$$\lambda \rightarrow \alpha.$$

Первая продукция выполняет движение вправо текущего символа ξ слова с помощью челнока α . Вторая продукция осуществляет левую конкатенацию челнока α к слову S . Тогда многократные применения продукции движения вправо приводят к инверсному расположению символов исходного слова. Рабочие символы попутно размечаются метасимволом α , который теперь понимается как маркер обработанной буквы.

Для перехода к следующей фазе вычислений необходимо ввести новый челнок движения вправо β . Челнок β при движении будет удалять метасимвол α , пока не останется один. В этой ситуации необходимо заключительной продукцией завершить работу ПС $\beta \rightarrow \bullet\lambda$.

Итоговое определяющее множество ПС обращения слова имеет вид

$$\left\{ \begin{array}{l} \alpha\alpha \rightarrow \beta\alpha \\ \beta\alpha \rightarrow \beta \\ \beta \rightarrow \bullet\lambda \\ \alpha\xi\eta \rightarrow \eta\alpha\xi \\ \lambda \rightarrow \alpha \end{array} \right. \quad (12)$$

Пусть $A=\{a,b,c\}$ и пусть в A задано слово $S=abc$. Тогда работа (12) над S имеет вид

$$\begin{aligned} abc &\xrightarrow{5} \alpha abc \xrightarrow{4} \beta aac \xrightarrow{4} \beta caa \xrightarrow{5} \alpha \beta caa \xrightarrow{4} \alpha \beta aca \xrightarrow{5} \alpha \beta aca \xrightarrow{5} \\ \alpha \alpha c a b a a &\xrightarrow{1} \beta \alpha c a b a a \xrightarrow{2} \beta c a b a a \xrightarrow{3} c \beta a b a a \xrightarrow{2} c \beta b a a \xrightarrow{3} c b \beta a a \xrightarrow{2} \\ c b \beta a &\xrightarrow{3} c b a \end{aligned}$$

Анализ работы (12) по шагам показывает, что четвертая и пятая продукции имеют непустое пересечение, что позволяет объединить их в одно подмножество. Исходная система продукций разбивается на подмножества:

$$\begin{aligned} - D_1 &= \{1, 2, 3\}; \\ - D_2 &= \{4, 5\}. \end{aligned}$$

Трассировка сработавших продукций из (12) показывает, что продукции действительно выполняются отдельно по подмножествам D_1 и D_2 . Значит, предварительная декомпозиция (12) на части является новым рабочим шагом в расширенной машине вывода, позволяющим сократить количество холостых срабатываний и повысить эффективность вывода решений.

Оценочно, декомпозиция исходной продукционной базы на 2 подмножества позволяет сократить число холостых срабатываний продукции до $(n / d_{max} + f(i) \cdot k)$ раз, где k – число уровней в графе вывода решений, $f(i)$ – модельная оценка точки разрыва ПС на подмножества.

Для моделирования временных затрат работы ПС задаются следующие параметры:

- ◆ количество продукции – n ;
- ◆ количество уровней в графе вывода – k ;
- ◆ количество декомпозируемых подмножеств – d ;
- ◆ номера точек разрыва ПС – i_1, i_2, \dots, i_d ;
- ◆ временная пропорция операций сопоставления и подстановки $\tau_1 : \tau_2$.

Пусть рассматривается декомпозиция базы продукции на 2 подмножества, одно из которых состоит из 2-х продукции. Пусть принята пропорция сопоставления и подстановки $5\tau : \tau$. Единственная точка разрыва ПС будет иметь варьируемое значение. Минимальный номер точки разрыва – 3, максимальный – $(n-2)$.

На рис. 4 представлены результаты моделирования времени работы $T(n)$ продукционной системы при $i=3, i=n-2$, для количества уровней в графе вывода $k=10$. На рис. 5 представлены результаты моделирования времени работы $T(n)$ продукционной системы при $i=3, i=n-2$ для количества уровней в графе вывода $k=15$.

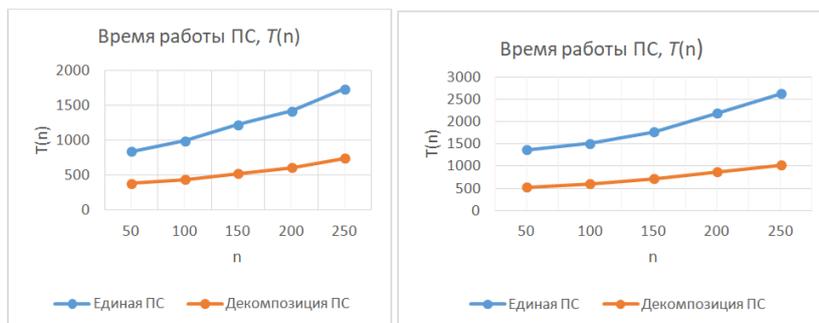


Рис. 4. Время зависимости времени работы ПС при $i=3, i=n-2$ для $k=10$

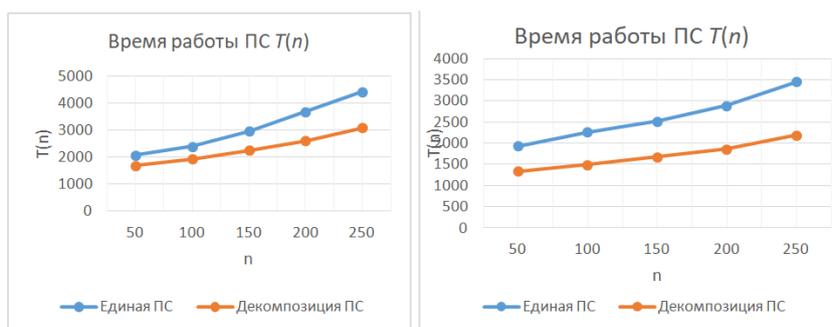


Рис. 5. Время зависимости времени работы ПС при $i=3, i=n-2$ для $k=15$

Графики демонстрируют существенное сокращение затрат времени при работе ПС, разбитой на независимые подмножества, так как в каждой итерации (уровень графа вывода) сокращается число рассматриваемых альтернатив в среднем на 25-30% от мощности наибольшего подмножества. При этом с увеличением количества продукции в 5 раз работа ПС при ее декомпозиции на 2 подмножества (одно из которых – две продукции) относительный выигрыш по времени описывается интервалами $(1,07 \div 1,52)$ и $(1,14 \div 1,74)$ раз, что подтверждает потенциальное преимущество работы ПС при ее разбиении на независимые подмножества.

Выводы:

1. Организация высокопроизводительных параллельных вычислений на основе продукционной парадигмы связывается с созданием теоретических и прикладных инструментов, направленных на сокращение временных затрат параллельного вывода.

2. В теоретическом аспекте перспективный путь организации параллельных вычислений – создание продукционной машины вывода, имеющей однородную исполнительную часть и средства декомпозиции базы продукции на подмножества с сохранением корректности результата. Созданная расширенная машина вывода отличается введением ее структуру блоков синтеза конфликтных слов, декомпозиции на независимые подмножества и параллельного выполнения. Анализ отношений пересечения продукции своими левыми частями позволяет логически разбить систему продукции на независимые подмножества и вести параллельный вывод по каждому из подмножеств с сохранением корректности результата. Продукционная база представлена независимыми подмножествами продукции, имеющими специальные слова-маркеры, которые участвуют в выборе стратегии вывода.

3. Временной выигрыш декомпозиции продукционной базы достигается за счет сокращения количества рассматриваемых вариантов (альтернатив) на этапах сопоставления активных продукции и данных, разрешения конфликтов, срабатывания продукции. Экспертная оценка показала, что в худшем случае декомпозиция исходной продукционной базы на два подмножества позволяет сократить число холостых срабатываний продукции на $(n / d_{max} + f(i) \cdot k)$ раз, где k – число уровней в графе вывода решений, d_{max} – мощность максимального подмножества. Моделированием установлено сокращение затрат времени при работе продукционной системы в среднем на 25-30% от мощности наибольшего подмножества. При этом с увеличением количества продукции до 5 раз работа продукционной системы при ее декомпозиции на 2 подмножества (одно из которых – две продукции) относительный выигрыш по времени в худшем случае составил $(1,07 \div 1,52)$ раз, что подтверждает преимущество создания расширенной машины вывода, которая имеет однородную структуру, и способна выполнять параллельные вычисления по независимым подмножествам продукции.

4. Структуризация вида продукции является прикладным инструментом повышения эффективности параллельного вывода за счет избирательной выборки активных продукции из базы. Для этого в дополнение к традиционным показателям выборки активных продукции вводятся условия изменения структуры продукции в процессе вывода, что позволяет зафиксировать в продукционной базе машины вывода общезначимые свойства и отношения, а в дальнейшем – использовать их для сокращения числа шагов работы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Воеводин В.В.* Математические модели и методы в параллельных процессах. – М.: Наука, 1986. – 296 с.
2. *Гузик В.Ф., Каляеа И.А., Левин И.И.* Реконфигурируемые вычислительные системы: учеб. пособие. – Таганрог: Изд-во ЮФУ, 2016. – 472 с.
3. *Булыга Ф.С., Курейчик В.М.* Сравнительный анализ методов векторизации текстовых данных большой размерности // Известия ЮФУ. Технические науки. – 2023. – № 2 (232). – С. 212-226.
4. *Титенко Е.А., Емельянов С.Г., Зерин И.С.* Однородные вычислительные структуры для параллельных символьных вычислений // Известия Юго-Западного государственного университета. – 2011. – № 6 (39). – Ч. 2. – С. 77-82.
5. *Lothaire M.* Algebraic Combinatorics on Words. – Cambridge: Cambridge University Press; 2002. (In Eng). – DOI: <https://doi.org/10.1017/CBO9781107326019>.
6. *Lothaire M.* Applied Combinatorics on Words. In: Encyclopedia of Mathematics and its Applications. – Cambridge: Cambridge University Press, 2005.
7. *Люгер Дж.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем. – М.: Издательский дом «Вильямс», 2003. – 864 с.
8. *Огнев И.В., Борисов В.В., Сутула Н.А.* Ассоциативные память, среды, системы, – М.: Горячая линия – Телеком, 2016. – 420 с.

9. *Гладков Л.А., Курейчик В.В., Курейчик В.М., Сороколетов П.В.* Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009. – 384 с.
10. *Адамов А.А., Эйсымонт Л.К.* Варианты архитектурных решений ЭКБ для систем искусственного интеллекта // Проектирование будущего. Проблемы цифровой реальности: Тр. 3-й Международной конференции. – М.: ИПМ им. М.В. Келдыша, 2020. – С. 112-131.
11. *Ва Б.У., Лоуррай М.Б., Гоцзе Ли.* ЭВМ для обработки символьной информации // ТИИЭР. – 1989. – Т. 77, № 4. – С. 5-40.
12. *Добрица В.П., Титенко Е.А., Халин Ю.А., Киселев А.В.* Системы искусственного интеллекта. – Курск: ЗАО «Университетская книга», 2023. – 143 с.
13. *Эйсымонт Л.К., Моляков А.С., Заборовский В.С., Федоров С.А.* Символьная обработка: эпизоды отечественной истории и перспективы // Матер. 2-й Всероссийской научно-технической конференции «Суперкомпьютерные технологии (СКТ-2012)». с. Дивноморское. 2012. – С. 202-206.
14. *Бурцев В.С.* Параллелизм вычислительных процессов и развитие архитектур суперЭВМ: Сб. статей. – М.: ТОРУС ПРЕСС, 2006. – 416 с.
15. *Wichert A.* Artificial intelligence and a universal quantum computer // AI Communications. – 2016. – Vol. 29, Issue 4. – P. 537-543.
16. *Бова В.В., Кравченко Ю.А., Родзин С.И.* Методы и алгоритмы кластеризации текстовых данных (обзор) // Известия ЮФУ. Технические науки. – 2022. – № 4 (228). – С. 122-143.
17. *Рыбина Г.В.* Основы построения интеллектуальных систем. – М.: Финансы и статистика. 2010. – 430 с.
18. *Попов Э.В.* Статические и динамические экспертные системы. – М.: Финансы и статистика. 1996. – 211 с.
19. *Довгаль В.М.* Методы модификации формальных систем обработки символьной информации. – Курск: Изд-во Курск. гос. техн. ун-т, 1996. – 115 с.
20. *Титенко Е.А.* Продукционная модель и абстрактная машина для реализации распределенных параллельных вычислений // Телекоммуникации. – 2012. – № 6. – С. 7-11.
21. *Тиликин А.П., Титенко Е.А.* Модификация цикла работы машины вывода для параллельных вычислительных устройств // Известия Юго-Западного государственного университета. – 2011. – № 6-2 (39). – С. 92-96.
22. *Тутов Е.Б., Титенко Е.А., Атакищев А.О., Цуканов К.С.* Применение контекстно-свободных грамматик для обработки текстов на естественном языке // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. – 2012. – № 2-1. – С. 97-101.
23. *Довгаль В.М., Титенко Е.А.* Параллельные стратегии выводов для систем поддержки принятия решения // Вестник новых медицинских технологий. – 2006. – Т. 13, № 1. – С. 143-144.
24. *Добрица В.П., Титенко Е.А., Халин Ю.А., Катыхин А.И.* Модели представления и обработки знаний в информационно-аналитических системах. – Курск: ЗАО «Университетская книга», 2023. – 172 с.
25. *Kapri X.B.* Основания математической логики: пер. с англ. – М.: Мир, 1969. – 568 с.

REFERENCES

1. *Voevodin V.V.* Matematicheskie modeli i metody v parallel'nykh protsessakh [Mathematical models and methods in parallel processes]. Moscow: Nauka, 1986, 296 p.
2. *Guzik V.F., Kalyaeva I.A., Levin I.I.* Rekonfiguriruemye vychislitel'nye sistemy: ucheb. posobie [Reconfigurable computing systems: textbook; manual]. Taganrog: Izd-vo YuFU, 2016, 472 p.
3. *Bulyga F.S., Kureychik V.M.* Sravnitel'nyy analiz metodov vektorizatsii tekstovykh dannykh bol'shoy razmernosti [Comparative analysis of methods for vectorizing high-dimensional text data], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2023, No. 2 (232), pp. 212-226.
4. *Titenko E.A., Emel'yanov S.G., Zerin I.S.* Odnorodnye vychislitel'nye struktury dlya parallel'nykh simvol'nykh vychisleniy [Homogeneous computational structures for parallel symbolic computations], *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta* [News of the South-West State University], 2011, No. 6 (39), Part 2, pp. 77-82.
5. *Lothaire M.* Algebraic Combinatorics on Words. Cambridge: Cambridge University Press; 2002. (In Eng). – DOI: <https://doi.org/10.1017/CBO9781107326019>.
6. *Lothaire M.* Applied Combinatorics on Words. In: Encyclopedia of Mathematics and its Applications. Cambridge: Cambridge University Press, 2005.
7. *Lyuger Dzh.F.* Iskusstvennyy intellekt: strategii i metody resheniya slozhnykh problem [Artificial intelligence: strategies and techniques for solving complex problems]. Moscow: Izdatel'skiy dom «Vil'yams», 2003, 864 p.

8. *Ognev I.V., Borisov V.V., Sutula N.A.* Assotsiativnye pamyat', sredy, sistemy [Associative memory, environments, systems], Moscow: Goryachaya liniya – Telekom, 2016, 420 p.
9. *Gladkov L.A., Kureychik V.V., Kureychik V.M., Sorokoletov P.V.* Bioinspirirovannye metody v optimizatsii [Bioinspired optimization methods]. Moscow: Fizmatlit, 2009, 384 p.
10. *Adamov A.A., Eysymont L.K.* Varianty arkhitekturnykh resheniy EKB dlya sistem iskusstvennogo intellekta [Options for ECB architectural solutions for artificial intelligence systems], *Proektirovanie budushchego. Problemy tsifrovoy real'nosti: Tr. 3-y Mezhdunarodnoy konferentsii* [Designing the future. Problems of digital reality: proceedings of the 3rd International Conference]. Moscow: IPM im. M.V. Keldysha, 2020, pp. 112-131.
11. *Va B.U., Louray M.B., Gotsze Li.* EVM dlya obrabotki simvol'noy informatsii [Computer for processing symbolic information], *TIIER* [TIIER], 1989, Vol. 77, No. 4, pp. 5-40.
12. *Dobritsa V.P., Titenko E.A., KHalin Yu.A., Kiselev A.V.* Sistemy iskusstvennogo intellekta [Artificial intelligence systems]. Kursk: ZAO «Universitetskaya kniga», 2023, 143 p.
13. *Eysymont L.K., Molyakov A.S., Zaborovskiy V.S., Fedorov S.A.* Simvol'naya obrabotka: epizody otechestvennoy istorii i perspektivy [Symbolic processing: episodes of Russian history and prospects], *Mater. 2-y Vserossiyskoy nauchno-tekhnicheskoy konferentsii «Superkomp'yuternye tekhnologii (CKT-2012)»* [Materials of the 2nd All-Russian Scientific and Technical Conference “Supercomputer Technologies (SKT-2012)”]. Divnomorskoe, 2012, pp. 202-206.
14. *Burtsev V.S.* Parallelizm vychislitel'nykh protsessov i razvitie arkhitektur superEVM: Sb. statey [Parallelism of computing processes and development of supercomputer architectures: collection of articles: Articles]. Moscow: TORUS PRESS, 2006, 416 p.
15. *Wichert A.* Artificial intelligence and a universal quantum computer, *AI Communications*, 2016, Vol. 29. Issue 4, pp. 537-543.
16. *Bova V.V., Kravchenko Yu.A., Rodzin S.I.* Metody i algoritmy klasterizatsii tekstovykh dannykh (obzor) [Methods and algorithms for clustering text data (review)], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2022, No. 4 (228), pp. 122-143.
17. *Rybina G.V.* Osnovy postroeniya intellektual'nykh sistem [Fundamentals of building intelligent systems]. Moscow: Finansy i statistika. 2010, 430 p.
18. *Popov E.V.* Statische i dinamicheskie ekspertnye sistemy [Static and dynamic expert systems]. Moscow: Finansy i statistika. 1996, 211 p.
19. *Dovgal' V.M.* Metody modifikatsii formal'nykh sistem obrabotki simvol'noy informatsii [Methods for modifying formal systems for processing symbolic information]. Kursk: Izd-vo Kursk. gos. tekhn. unt, 1996, 115 p.
20. *Titenko E.A.* Produktsionnaya model' i abstraktnaya mashina dlya realizatsii raspredelennykh parallel'nykh vychisleniy [Production model and abstract machine for implementing distributed parallel computing], *Telekommunikatsii* [Telecommunications], 2012, No. 6, pp. 7-11.
21. *Tipikin A.P., Titenko E.A.* Modifikatsiya tsikla raboty mashiny vyvoda dlya parallel'nykh vychislitel'nykh ustroystv [Modification of the output machine operation cycle for parallel computing devices], *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta* [News of the South-West State University], 2011, No. 6-2 (39), pp. 92-96.
22. *Tutov E.B., Titenko E.A., Atakishchev A.O., TSukanov K.S.* Primenenie kontekstno-svobodnykh grammatik dlya obrabotki tekstov na estestvennom yazyke [Application of context-free grammars for processing texts in natural language], *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta. Seriya: Upravlenie, vychislitel'naya tekhnika, informatika. Meditsinskoe priborostroenie* [News of the South-Western State University. Series: Management, computer technology, computer science. Medical instrumentation], 2012, No. 2-1, pp. 97-101.
23. *Dovgal' V.M., Titenko E.A.* Parallelnye strategii vyvodov dlya sistem podderzhki prinyatiya resheniya [Parallel inference strategies for decision support systems], *Vestnik novykh meditsinskikh tekhnologiy* [Bulletin of new medical technologies], 2006, Vol. 13, No. 1, pp. 143-144.
24. *Dobritsa V.P., Titenko E.A., Khalin Yu.A., Katykhin A.I.* Modeli predstavleniya i obrabotki znaniy v informatsionno-analiticheskikh sistemakh [Models of representation and processing of knowledge in information and analytical systems]. Kursk: ZAO «Universitetskaya kniga», 2023, 172 p.
25. *Karri Kh.B.* Osnovaniya matematicheskoy logiki [Foundations of mathematical logic]: transl. from english. Moscow: Mir, 1969, 568 p.

Статью рекомендовал к опубликованию д.т.н. Ю.А. Кравченко.

Титенко Евгений Анатольевич – Юго-Западный государственный университет; e-mail: johntit@mail.ru; г. Курск, Россия; тел.: +79051588904; к.т.н.; доцент; в.н.с.

Чернецкая Ирина Евгеньевна – e-mail: white@mail.ru; д.т.н.; доцент; зав. кафедрой вычислительной техники.

Титенко Михаил Андреевич – e-mail: mikhail-titenko@mail.ru; аспирант.

Мельник Эдуард Всеволодович – Федеральный исследовательский центр Южный научный центр Российской академии наук; e-mail: evml17@mail.ru; г. Таганрог, Россия; тел.: 88634615459; д.т.н., г.н.с.

Трокоз Дмитрий Анатольевич – Пензенский государственный технологический университет, e-mail: trokoz@penzgtu.ru; г. Пенза, Россия; проректор по научной работе.

Titenko Evgeny Anatolievich – South-West State University; e-mail: johntit@mail.ru; Kursk, Russia; phone: +79051588904; candidate of technical sciences, associate professor, leading researcher.

Chernetskaya Irina Evgenievna – e-mail: white@mail.ru; dr. of eng. sc.; associate professor; head of the department of computer science.

Titenko Mikhail Andreevich – e-mail: mikhail-titenko@mail.ru; postgraduate of the department of computer science.

Melnik Eduard Vsevolodovich – Federal Research Center, The Southern Scientific Center of the Russian Academy of Sciences; e-mail: evml17@mail.ru; Taganrog, Russia; phone: +78634615459; dr. of eng. sc.; chief researcher.

Trokoz Dmitry Anatolievich – Penza State Technological University; e-mail: trokoz@penzgtu.ru; Penza, Russia; dr. of eng. sc.; associate professor; vice-rector for scientific work.

УДК 004.9

DOI 10.18522/2311-3103-2024-2-268-278

А.А. Белевцев, А.М. Белевцев, В.А. Балыбердин

МЕТОДИКА ПОСТРОЕНИЯ ДОРОЖНЫХ КАРТ РАЗВИТИЯ ТЕХНОЛОГИЧЕСКИХ ТРЕНДОВ И ТЕХНОЛОГИЙ В УСЛОВИЯХ ОГРАНИЧЕНИЙ

В современных условиях перехода к шестому технологическому укладу и обострения глобальной конкуренции первостепенное значение приобретает проблема обеспечения технологического суверенитета и технологического превосходства на национальном и корпоративном уровнях. В основе решения этой проблемы лежит организация эффективного развития научных исследований и, в первую очередь, исследований, направленных на создание новых высокоэффективных технологий в различных сферах деятельности. В работе предлагается методика анализа и развития технологических трендов и технологий в условиях неполноты и нечеткости информации. Методика обеспечивает прогнозирование и оценку развития общемировых технологически трендов для заданных предметных областей, а также определение возможных альтернатив технологического развития и разработки стратегии достижения технологического паритета и технологического превосходства в условиях ограничений. Реализация предлагаемых в методике процедур позволяет определить организационную стратегию достижения технологического паритета на различных уровнях рассмотрения, а также сформировать перечень технологий развития технологических трендов, обеспечивающих технологическое превосходство. Определено, что методика построения дорожных карт развития технологических трендов и технологий в условиях ограничений должна обеспечивать: 1. Анализ развития общемировых технологических трендов для заданных предметных отраслей на национальном (на пример в РФ), отраслевом и корпоративном уровнях. 2. Анализ и разработку стратегии обеспечения технологического паритета и технологического превосходства в условиях ограничений. В отличие от имеющихся исследований представленная методика позволяет: – получать количественные прогнозные оценки развития технологи-