

ПРИМЕНЕНИЕ ТЕОРИИ РЕШЕНИЯ ИЗОБРЕТАТЕЛЬСКИХ ЗАДАЧ ДЛЯ ПОДГОТОВКИ БУДУЩИХ ИТ-СПЕЦИАЛИСТОВ

С. А. ВЕЛЬЧЕНКО¹⁾, Д. Г. МЕДВЕДЕВ¹⁾

¹⁾Белорусский государственный университет, пр. Независимости, 4, 220030, г. Минск, Беларусь

Аннотация. Показано применение идей теории решения изобретательских задач в процессе обучения студентов и магистрантов механико-математического факультета Белорусского государственного университета параллельным вычислениям. Предложена многоуровневая модель развития алгоритмического и творческого мышления.

Ключевые слова: механико-математический факультет; параллельные вычисления; программирование; теория решения изобретательских задач; многоуровневая модель; алгоритмическое и творческое мышление.

APPLICATION OF THEORY OF INVENTIVE PROBLEM SOLVING FOR TRAINING FUTURE IT SPECIALISTS

S. A. VELCHENKO^a, D. G. MEDVEDEV^a

^aBelarusian State University, 4 Niezaliezhnasci Avenue, Minsk 220030, Belarus

Corresponding author: S. A. Velchenko (semmi.vall@gmail.com)

Abstract. The application of theory of inventive problem solving in the process of teaching parallel computing to students and master's degree student of the faculty of mechanics and mathematics of the Belarusian State University is shown. A multilevel model for the development of algorithmic and creative thinking is proposed.

Keywords: faculty of mechanics and mathematics; parallel computing; programming; theory of inventive problem solving; multilevel model; algorithmic and creative thinking.

Введение

Сегодня многоядерным ноутбуком и смартфоном никого не удивить. Принципы распараллеливания и конвейеризации получили широкое распространение благодаря тому, что становятся источником сверхвысокой производительности и надежности вычислительных средств, а также фактором ускорения вычислений и достижения их требуемой точности. В свою очередь, проблемы организации па-

раллельных вычислений обуславливают появление математических задач, одной из которых считается синхронизация большого числа взаимодействующих параллельных (многопоточных) процессов, вызванная тем, что при гигафлопсных, терафлопсных и петафлопсных скоростях протекания вычислительных операций привычная классическая оптимизация не работает. Возникают сложные в математическом

Образец цитирования:

Вельченко СА, Медведев ДГ. Применение теории решения изобретательских задач для подготовки будущих ИТ-специалистов. *Университетский педагогический журнал*. 2024;1:32–40.
EDN: PHZBUM

For citation:

Velchenko SA, Medvedev DG. Application of theory of inventive problem solving for training future IT specialists. *University Pedagogical Journal*. 2024;1:32–40. Russian.
EDN: PHZBUM

Авторы:

Сергей Александрович Вельченко – старший преподаватель кафедры веб-технологий и компьютерного моделирования механико-математического факультета.
Дмитрий Георгиевич Медведев – доктор педагогических наук, профессор; профессор кафедры теоретической и прикладной механики механико-математического факультета.

Authors:

Sergey A. Velchenko, senior lecturer at the department of web technologies and computer modelling, faculty of mechanics and mathematics.
semmi.vall@gmail.com
<https://orcid.org/0000-0001-6527-596X>
Dmitry G. Medvedev, doctor of science (pedagogy), full professor; professor at the department of theoretical and applied mechanics, faculty of mechanics and mathematics.
medvedev@bsu.by

отношении задачи по отображению алгоритмов и соответствующих программных реализаций (из разных предметных областей), которые «ложатся» на архитектуру многопроцессорных вычислительных комплексов в условиях массового параллелизма, разработки и обоснования новых приемов ускорения вычислений. Для решения таких задач необходимо создать математические модели, которые не только учитывают разнообразие видов параллелизма, но и позволяют получить количественные и качественные результаты. Ввиду дискретного и комбинаторного характера этих задач определенный прогресс на пути их решения может быть достигнут за счет применения математического аппарата дискретных динамических систем и дискретной оптимизации, теории расписаний и сетевых графов и гиперграфов, теории множеств и алгоритмов с учетом их сложности, а также алгебры матриц и гиперматриц, алгебры логики, дискретных преобразований Фурье и др. Особенно следует обращать внимание на размерности решаемых задач и точность вычислений.

В целях достижения прогресса в вышеупомянутых прорывных направлениях развития информационных технологий (ИТ) в первую очередь нужно уделить внимание качеству подготовки специалистов в области параллельных алгоритмов и параллельного программирования. Важную роль при формировании соответствующих навыков и компетенций у студентов играет правильно выстроенный процесс обучения от первой лекции до защиты диплома: по-

следовательность изучения академических предметов, предшествующих конкретной дисциплине и следующих за ней, проработка межпредметных связей между математикой и программированием, обеспечение студентов адаптированными учебными пособиями и передовыми средствами обучения, предоставление им выбора индивидуальных обучающих технологий в созданной информационно-образовательной среде кафедры и факультета.

При прохождении курса «Параллельное программирование» студентам предлагается разработанный авторами данной статьи учебно-методический комплекс, включающий программную, теоретическую и практическую составляющие. Решающее значение приобретает правильный подбор примеров и задач для лабораторных работ в компьютерных классах и тем докладов для обсуждения на семинарских занятиях. В данном комплексе представлены задания для изучения различных видов параллелизма, а также методы по развитию творческих способностей и варианты их применения для решения конкретных задач и получения нужного программного продукта.

На данный момент в связи с освоением современной наукой сложных саморазвивающихся объектов и появлением в математике соответствующего инструментария для их описания, к которому относятся параллельные вычисления и параллельное программирование, формирование творческих способностей у обучающихся становится особенно важной задачей высшего образования.

Основные положения

Актуальность и сложность распараллеливания и конвейеризации обуславливают выстраивание особого подхода к обучению студентов. Опыт авторов настоящей статьи позволяет утверждать, что использование хорошо зарекомендовавшей себя во многих сферах теории решения изобретательских задач (ТРИЗ)¹ [1], различных командных и групповых методов обучения, основанных на проблемных и неразрешимых ситуациях, ломает стереотипное мышление у студентов, поскольку развитие творческого мышления – особое условие самой предметной области.

Разработка и написание параллельных алгоритмов требуют от студентов задействования творческих способностей и их применения для решения задач. Соответственно, данные способности нужно развивать, и этому могут способствовать адаптированные материалы системы ТРИЗ, которые позволяют вовлекать студентов в процесс написания параллельных алгоритмов. Чтобы обучающиеся могли создавать параллельные алгоритмические цепочки, необходимо вовлечь их в деятельность,

связанную с анализом параллельного процесса, включая программное взаимодействие. Для написания параллельных алгоритмов и программ нужно использовать специальное алгоритмическое мышление, которое при осознанной и организованной последовательности действий позволяет построить параллельный алгоритм. В итоге студенты смогут раскрыть потенциал параллельной обработки данных и достичь новых высот в программировании.

Цель авторов настоящей статьи – формирование у студентов алгоритмической и творческой мыслительной деятельности, связанной с осознанием, анализом и оценкой способов протекания параллельных процессов. ТРИЗ также помогает продуктивно обучать параллельному программированию [1]. Студенты раскрывают свои личностные особенности, создается возможность для построения индивидуальной образовательной траектории. Они могут ставить цели учебного познания, выбирать подходящие формы и методы обучения, а также осуществлять рефлексию своей образовательной деятельности.

¹Кларин М. В. Корпоративный тренинг, наставничество, коучинг : учеб. пособие для вузов. М. : Юрайт, 2023. 288 с. ; Орлов М. А. Первичные инструменты ТРИЗ : справ. практика для создания инновационных идей и решений. М. : Солон-пресс, 2010. 128 с.

ТРИЗ появилась в середине XX в., когда ИТ еще не было. По этой причине информационная база ТРИЗ была связана только с классическими техническими системами: машиностроением, электротехникой, ракетостроением и др. Однако обнаруженные закономерности «...оказались универсальными и эффективными в том числе в ИТ. Во-первых, информационные системы – это разновидность технических систем. Во-вторых, в продумывании новых решений разработчики сталкиваются с проблемами»².

Методология рассматриваемой теории «основана на том, что системы развиваются по объективным законам»³. Знание последних «позволяет совершенствовать системы целенаправленно, без перебора вариантов и ошибок»⁴. Г. С. Альтшуллер [1] проанализировал огромный патентный фонд и выявил, «...что для создания изобретений нужно знать закономерности развития техники. Здесь не имеются в виду законы физики или мироздания»⁵, в ТРИЗ выявлены свои специфические законы, в частности закон повышения идеальности технических систем (полезная функция, которую выполняет система, сохраняется или улучшается, а сама система становится меньше, проще и дешевле (пример – тачскрин вместо кнопок)).

Еще одно положение ТРИЗ можно назвать принципом противоречия. В процессе развития систем возникают, обостряются и разрешаются противоречия. ТРИЗ включает методы и инструменты для их устранения. У любой технической задачи или бизнес-задачи существует единый метод обработки: необходимо от ситуации, где есть проблема, перейти к ситуации, где ее нет. В ТРИЗ этот процесс происходит через поиск противоречий. Противоречие – это способ фокусировки нашего мышления на сути проблемы⁶. В этом смысле ТРИЗ является противоположностью популярному в ИТ методу мозгового штурма, или брейнстормингу, где идеи генерируются без фокусировки на моментах, важных для решения проблемы. По этой причине такой способ имеет низкую эффективность.

Создавая новый ИТ-продукт, инженеры также постоянно сталкиваются с противоречиями. Например,

чтобы было удобнее читать с экрана смартфона, надо увеличить размер гаджета, но это приведет к тому, что людям неудобно будет носить его в кармане. ТРИЗ помогает понять причины, которые лежат в основе проблемы, и предлагает способы совершенствования, опираясь на объективные законы развития техники. Используя накопленные многими предшественниками знания, можно изобретать продукты гораздо быстрее. Поэтому ТРИЗ способствует подготовке ИТ-специалиста параллельному программированию [2].

Многолетний опыт преподавания позволяет предложить многоуровневую модель, которая отображает единство и взаимосвязь учебных предметов и методов, а также определенных навыков и знаний, которые приобретают студенты специальности 1-31 03 08 «Математика и информационные технологии (по направлениям)» до изучения дисциплины «Параллельное программирование» и в процессе ее освоения (рис. 1).

Названная дисциплина преподается студентам третьего курса очной формы обучения. Учебный материал состоит из следующих тем:

- темы 1 «Метод структурирования программных ресурсов»;
- темы 2 «Математическая модель сосредоточенной обработки конкурирующих процессов»;
- темы 3 «Задачи оптимальной организации конкурирующих процессов при сосредоточенной обработке»;
- темы 4 «Конкурирующие процессы при ограниченном числе копий программного ресурса»;
- темы 5 «Время реализации конкурирующих процессов при распределенной обработке»;
- темы 6 «Оптимизационные задачи конкурирующих процессов при распределенной обработке»;
- темы 7 «Векторно-конвейерная обработка»;
- темы 8 «Построение оптимальной системы одинаково распределенных процессов в многопроцессорных вычислительных комплексах».

С опорой на образовательный стандарт высшего образования⁷ и учебный план⁸ можно выделить компетенции ИТ-специалистов⁹.

²К коду применить сложно. Зачем EPAM и Wargaming изобретательская теория советского новатора и фантаста // devby.by : сайт. URL: <https://devby.io/news/triz> (дата обращения: 12.03.2024).

³Там же.

⁴Там же.

⁵Там же.

⁶Там же.

⁷Образовательный стандарт высшего образования (ОСВО 1-31 03 08-2021). Специальность 1-31 03 08 «Математика и информационные технологии (по направлениям)» [Электронный ресурс] // Электронная библиотека БГУ. URL: <https://elib.bsu.by/bitstream/123456789/291473/1/Математика%20и%20информационные%20технологии%20%28по%20направлениям%29%20ОСВО%201-31%2003%2008-2021.pdf> (дата обращения: 24.03.2024).

⁸Учебный план. Специальность 1-31 03 08 «Математика и информационные технологии (по направлениям)» [Электронный ресурс] // Электронная библиотека БГУ. URL: <https://elib.bsu.by/bitstream/123456789/278498/1/1-31%2003%2008%20Математика%20и%20информационные%20технологии%20уч.план.pdf> (дата обращения: 24.03.2024).

⁹Коваленко Н. С., Вельченко С. А. Параллельное программирование : учеб. программа учреждения высш. образования по учеб. дисциплине для спец. 1-31 03 08 «Математика и информационные технологии (по направлениям)», направления спец. 1-31 03 08-01 «Веб-программирование и интернет-технологии» [Электронный ресурс] // Электронная библиотека БГУ. URL: <https://elib.bsu.by/bitstream/123456789/242130/1/Параллельное%20программирование.PDF> (дата обращения: 24.03.2024).



Рис. 1. Многоуровневая модель учебных предметов, методов и навыков развития алгоритмического и творческого мышления
Fig. 1. Multilevel model of educational subjects, methods and skills for the development of algorithmic and creative thinking

Использование эвристических задач и системы ТРИЗ способствует освоению учебной дисциплины «Параллельное программирование» и должно обеспечить формирование таких специализированных компетенций [2], как проектирование, разработка, тестирование и маркетинг информационных решений в интернете с учетом их последующего масштабирования и обработки возникающих больших

объемов данных; разработка информационных решений для поиска и извлечения данных, а также специализированных моделей машинного обучения и алгоритмов анализа данных; становление и тестирование прикладных гипотез на основе данных; развертывание моделей в облачной среде; осуществление визуализации и бизнес-анализа полученных решений¹⁰.

¹⁰Все специализированные компетенции прописаны в учебном плане (см.: Учебный план. Специальность 1-31 03 08 «Математика и информационные технологии (по направлениям)» [Электронный ресурс] // Электронная библиотека БГУ. URL: https://elib.bsu.by/bitstream/123456789/278498/1/1-31%2003%2008%20Математика%20и%20информ_технологии_31-1-220%20уч.pdf (дата обращения: 24.03.2024)).

Методическое применение

По изучаемому курсу было издано учебно-методическое пособие¹¹, полностью соответствующее упомянутым учебным программам и стандарту. Оно содержит значительное количество обучающих задач, при решении которых студенты приобретают алгоритмическое мышление и постигают процесс распараллеливания.

Набор обучающих задач включает задачи пяти уровней сложности, которая зависит от многих факторов, будь то использование определенных технологий и применение их для решения данных задач; применение математических методов или сведение решения к готовым алгоритмам и программным библиотекам; поиск количества алгоритмов, необходимых для их решения. Задача может быть и относительно простой, но если студент подошел к ее решению креативно, грамотно обращаясь к технологиям и методам решения и эффектно презентуя и тестируя результаты, то он заслуживает дополнительных баллов. Формально задача считается тем сложнее, чем

больше действий и технологий в ней надо применять, чтобы получить адекватное решение. Значительное количество приведенных в пособии задач не могут быть решены однозначно, для них требуется найти оптимальное решение или решение, близкое к нему¹².

Ознакомление будущих ИТ-специалистов с идеями ТРИЗ и грамотный подбор задач показали их высокую продуктивность в овладении профессиональными компетенциями. ТРИЗ – это едва ли не единственная системная теория обучения творчеству, поэтому в решении задач по ТРИЗ нет оценок и единственного правильного ответа.

По мнению авторов настоящей статьи, процесс использования подходов, разработанных с помощью ТРИЗ, при решении задач параллельного программирования наиболее наглядно представлен на классической схеме простейшего алгоритма изобретения «Самый типичный алгоритм решения задач»¹³ (рис. 2). Детальное пояснение данной схемы можно найти в источнике [3].

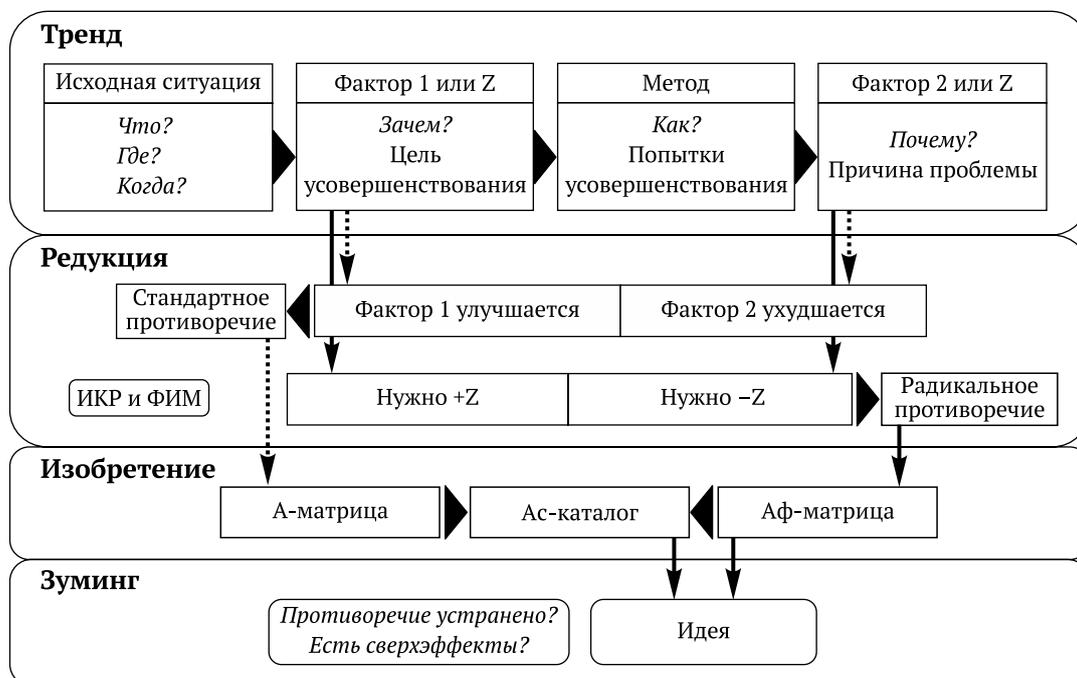


Рис. 2. Самый типичный алгоритм решения задач по ТРИЗ:
 ИКР – идеальный конечный результат; ФИМ – функциональная идеальная модель
 (разработано авторами по [3, с. 14])

Fig. 2. The most typical algorithm for solving TRIZ problems:
 ИКР – perfect end result; ФИМ – functional ideal model
 (developed by authors on [3, p. 14])

¹¹Коваленко Н. С., Вельченко С. А., Овсеев М. И. Параллельное программирование. Математические модели, методы и алгоритмы : учеб.-метод. пособие. Минск : БГУ, 2022. 255 с.

¹²Там же.

¹³Орлов М. А. Основы классической ТРИЗ : практ. рук. для изобретат. мышления. 2-е изд., испр. и доп. М. : Солон-пресс, 2006. 432 с. ; Он же. Азбука ТРИЗ. Основы изобретательного мышления. М. : Солон-пресс, 2014. 208 с.

Ниже показан пример (из учебно-методического пособия¹⁴) решения задачи при изучении синхронных режимов взаимодействия процессов, процессоров (ядер) и блоков программного ресурса. При разборе задачи продемонстрировано, как основные идеи и термины ТРИЗ применяются авторами настоящей статьи для решения задач параллельного программирования (жирным шрифтом выделена терминология ТРИЗ).

Тренд. Рассмотрим следующую задачу. Пусть p – число процессоров многопроцессорной системы ($p = 3$), n – число конкурирующих неоднородных сосредоточенных процессов ($n = 9$), s – число блоков линейно-структурированного программного ресурса ($s = 4$). Задана матрица времени выполнения блоков программного ресурса (T^c) с параметрами $[t_{ij}]$, $i = \overline{1, 9}$, $j = \overline{1, 4}$:

$$T^c = \begin{bmatrix} 4 & 2 & 5 & 1 \\ 5 & 1 & 2 & 4 \\ 1 & 5 & 3 & 4 \\ 3 & 1 & 5 & 1 \\ 2 & 4 & 1 & 3 \\ 3 & 2 & 1 & 1 \\ 1 & 5 & 1 & 4 \\ 5 & 1 & 4 & 1 \\ 1 & 1 & 3 & 1 \end{bmatrix}.$$

Необходимо построить сетевой дуговзвешенный граф, отображающий время выполнения блоков программного ресурса в условиях первого синхронного режима, и найти минимальное общее время выполнения девяти процессов на трех процессорах с четырьмя блоками при заданной матрице. При этом для построения нужно использовать аналитические формулы из упомянутого пособия.

Кроме того, необходимо составить совмещенную линейную диаграмму для первого синхронного режима.

Ниже показано, как рассматриваемый алгоритм работает на этапе решения.

Редукция. На данном этапе происходят экстрагирование, первое выявление навигаторов трансформации (см. таблицу).

Навигаторы трансформации Transformation navigators

Навигатор	Применение
Копирование	Повторяем ход решения задачи на основе математических формул, закономерностей и теорем, выведенных для данного случая
Наоборот	Можно решать задачу методом «от общего к частному» или методом «от частного к общему»
Дробление	Добавление дуговзвешенного графа, построение диаграмм Ганта
Объединение	Объединение математических моделей с графическим представлением и с написанием программ
Обратная связь	Проверка алгоритма путем тестирования на выборке, проверка программного продукта путем тестирования приложения

Изобретение. Решаем аналитически данную задачу с использованием теорем и примеров из названного учебно-методического пособия. Найдем минимальное общее время выполнения множества Тогда

конкурирующих процессов. Обозначим следующие показатели: T_j^i – время выполнения i -го блока j -го процесса; Δ_m^k – вид наклона дуг в сетевом графе. В данном случае $k = 3$.

$$T_i^m = \sum [t_{ij}^m], i = \overline{1, 3}, m = \overline{1, 3}.$$

Таким образом,

$$T_1^1 = \sum_{j=1}^4 t_{1j}^1 = 12, T_2^1 = \sum_{j=1}^4 t_{2j}^1 = 12, T_3^1 = \sum_{j=1}^4 t_{3j}^1 = 12;$$

¹⁴Коваленко Н. С., Вельченко С. А., Овсеев М. И. Параллельное программирование... 255 с.

$$T_1^2 = \sum_{j=1}^4 t_{1j}^2 = 10, T_2^2 = \sum_{j=1}^4 t_{2j}^2 = 10, T_3^2 = \sum_{j=1}^4 t_{3j}^2 = 7;$$

$$T_1^3 = \sum_{j=1}^4 t_{1j}^3 = 11, T_2^3 = \sum_{j=1}^4 t_{2j}^3 = 11, T_3^3 = \sum_{j=1}^4 t_{3j}^3 = 6.$$

Находим

$$\Delta_i^m = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{ij}^m - \sum_{j=1}^{u-1} t_{i+1,j}^m \right], i = \overline{1, 2}, m = \overline{1, 3}.$$

Так,

$$\Delta_1^1 = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{1j}^1 - \sum_{j=1}^{u-1} t_{2,j}^1 \right] =$$

$$= \max \left[t_{11}^1, t_{11}^1 + t_{12}^1 - t_{21}^1, t_{11}^1 + t_{12}^1 + t_{13}^1 - t_{21}^1 + t_{22}^1, t_{11}^1 + t_{12}^1 + t_{13}^1 + t_{14}^1 - t_{21}^1 - t_{22}^1 - t_{23}^1 \right] =$$

$$= \max \left[4, 4 + 2 - 5, 4 + 2 + 5 - 5 - 1, 4 + 2 + 5 + 1 - 5 - 1 - 2 \right] = 5,$$

$$\Delta_2^1 = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{2j}^1 - \sum_{j=1}^{u-1} t_{3,j}^1 \right] =$$

$$= \max \left[t_{21}^1, t_{21}^1 + t_{22}^1 - t_{31}^1, t_{21}^1 + t_{22}^1 + t_{23}^1 - t_{31}^1 - t_{32}^1, t_{21}^1 + t_{22}^1 + t_{23}^1 + t_{24}^1 - t_{31}^1 - t_{32}^1 - t_{33}^1 \right] =$$

$$= \max \left[5, 4 + 1 - 1, 5 + 1 + 2 - 1 - 5, 5 + 1 + 2 + 4 - 1 - 5 - 3 \right] = 5;$$

$$\Delta_1^2 = \max \left[3, 2, 3, 3 \right] = 3, \Delta_2^2 = \max \left[3, 2, 3, 3 \right] = 4;$$

$$\Delta_1^3 = \max \left[1, 1, 1, 1 \right] = 1, \Delta_2^3 = \max \left[5, 5, 8, 6 \right] = 8.$$

Находим

$$\Delta_3^m = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{3j}^m - \sum_{j=1}^{u-1} t_{1,j}^m \right], m = \overline{1, 2}.$$

Таким образом,

$$\Delta_3^1 = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{3j}^1 - \sum_{j=1}^{u-1} t_{1,j}^1 \right] =$$

$$= \max \left[t_{31}^1, t_{31}^1 + t_{32}^1 - t_{11}^1, t_{31}^1 + t_{32}^1 + t_{33}^1 - t_{11}^1 - t_{12}^1, t_{31}^1 + t_{32}^1 + t_{33}^1 + t_{34}^1 - t_{11}^1 - t_{12}^1 - t_{13}^1 \right] =$$

$$= \max \left[1, 1 + 5 - 3, 1 + 5 + 3 - 3 - 1, 1 + 5 + 3 + 4 - 3 - 1 - 5 \right] = 5;$$

$$\Delta_3^2 = \max_{1 \leq u \leq 4} \left[\sum_{j=1}^u t_{3j}^2 - \sum_{j=1}^{u-1} t_{1,j}^2 \right] =$$

$$= \max \left[t_{31}^2, t_{31}^2 + t_{32}^2 - t_{11}^2, t_{31}^2 + t_{32}^2 + t_{33}^2 - t_{11}^2 - t_{12}^2, t_{31}^2 + t_{32}^2 + t_{33}^2 + t_{34}^2 - t_{11}^2 - t_{12}^2 - t_{13}^2 \right] =$$

$$= \max \left[3, 3 + 2 - 1, 3 + 2 + 1 - 1 - 5, 3 + 2 + 1 - 1 - 1 - 5 - 1 \right] = 4.$$

Строим сетевой дуговзвешенный граф (рис. 3), вершины которого размещены в узлах прямоугольной решетки размерности 3×4 . Значения величин T_i^m , Δ_i^m и Δ_p^m указаны на соответствующих дугах этого графа. Длина критического пути в графе равна 42 ед. (дуги, через которые проходит критический путь, выделены полужирными стрелками). Следовательно,

минимальное общее время выполнения девяти процессов на трех процессорах с четырьмя блоками при заданной матрице составляет 42 ед.

Проверим вышеприведенное решение построением совмещенной диаграммы Ганта для первого синхронного режима (рис. 4). Римскими цифрами обозначены номера блоков.

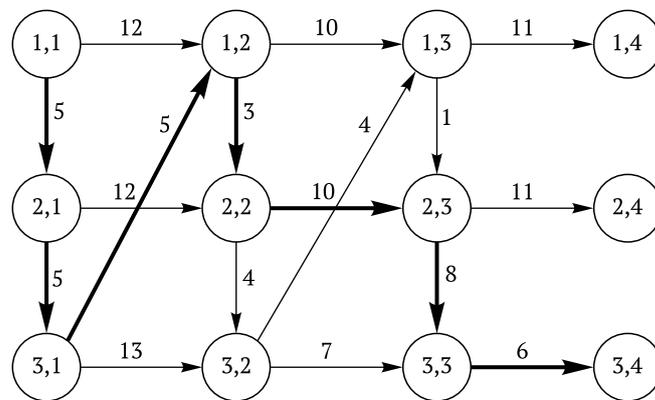


Рис. 3. Дуговзвешенный граф
Fig. 3. Arc-weighted graph

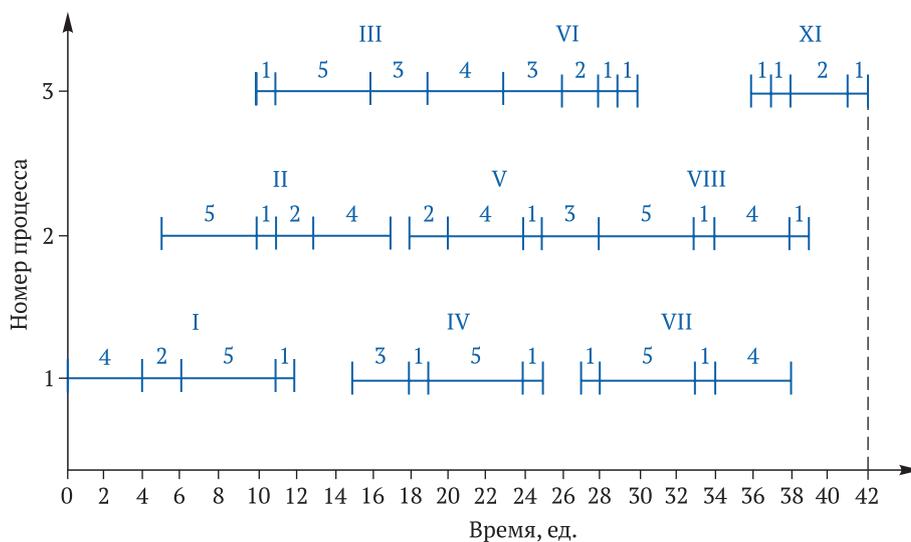


Рис. 4. Диаграмма Ганта для первого синхронного режима
Fig. 4. Gantt chart for the first synchronous mode

Из полученной диаграммы видно, что длина критического пути также составляет 42 ед.

Зуминг. Таким образом, два способа решения задачи – аналитический (формульный и алгоритмический) с графовым (построение и анализ сетевого дуговзвешенного графа с помощью матричного представления) и диаграммный (использование аппарата линейных диаграмм Ганта) – подтверждают правильность результата.

Приведенные выкладки используются при написании программных модулей для решения задач данного типа на различных языках программиро-

вания высокого уровня, содержащих расширения и библиотеки и поддерживающих описание параллельных потоков (например, языки C++, Java, Python).

При решении задач параллельного программирования на разных этапах используются как индуктивные (от простого к сложному), так и дедуктивные (от общей математической модели к решению конкретной задачи) методы.

Опытным путем установлено, что применение ТРИЗ студентами ИТ-специальностей положительно сказывается на результатах решения задач на распараллеливание.

Заключение

Выявлена эффективность использования ТРИЗ для развития алгоритмического и творческого мышления студентов при написании параллельных алгоритмов и программ. При этом были обнаружены различные варианты ее применения: в исследовательской деятельности, развитии творческих

способностей ИТ-специалистов при анализе задач, не имеющих однозначного решения.

Результаты, которые получились при апробации ТРИЗ в учебных группах, были следующими:

- значительно возросла мотивация студентов к обучению;

- укрепились навыки командной работы;
- повысился уровень творческого подхода к решению сложных задач.

Предложена многоуровневая дидактическая модель для формирования и развития способов алгоритмической мыслительной деятельности с помощью эвристических задач и системы ТРИЗ. Модель практически реализована в учебном процессе. Разработаны учебные программы по дисциплине «Параллельное программирование», издано учебно-методическое пособие¹⁵, которое включает целостную систему задач разного уровня сложности. Были определены психолого-педагогические условия, способствующие наиболее успешному формированию алгоритмической мыслительной деятельности студентов. Важными факторами для благополучного развития этой деятельности являются командная работа, азарт игры, сотрудничество и взаимопомощь, а также стойкое

формирование навыка генерирования новых идей. Разработаны методики проведения лабораторно-компьютерных практикумов с использованием системы ТРИЗ, которые успешно проверены в ходе учебного процесса.

Особое внимание следует уделить проблеме организации самостоятельной работы студентов в информационно-образовательной среде в целях более интенсивного изучения учебной информации и ее творческого осмысления с акцентом на активизацию алгоритмической мыслительной деятельности и творческого развития.

Предлагаемый подход, основанный на педагогическом дизайне, эвристических заданиях и системе ТРИЗ, позволяет создать учебно-методический комплекс и подготовить ИТ-специалиста как в смешанной, так и в дистанционной образовательной среде [4].

Библиографические ссылки

1. Альтшуллер ГС. *Найти идею. Введение в теорию решения изобретательских задач*. Новосибирск: Наука; 1986. 208 с.
2. Вельченко СА. Формирование технической ИТ-компетенции при обучении студентов параллельному программированию. *Университетский педагогический журнал*. 2022;2:66–72. EDN: NAUUTP.
3. Орлов МА. *Первичные инструменты ТРИЗ*. Москва: Солон-пресс; 2010. 128 с.
4. Вельченко СА. Использование полипарадигмального подхода при обучении параллельному программированию студентов университета. У: Академія паслядыпломнай адукацыі. *Зборнік навуковых прац Акадэміі паслядыпломнай адукацыі*. Выпуск 19. Мінск: Акадэмія паслядыпломнай адукацыі; 2021. с. 96–107.

References

1. Altshuller GS. *Naiti ideyu. Vvedenie v teoriyu resheniya izobretatel'skikh zadach* [Find an idea. Introduction to the theory of solving inventive problems]. Novosibirsk: Nauka; 1986. 208 p. Russian.
2. Velchenko SA. Formation of technical IT competence when teaching students in parallel programming. *University Pedagogical Journal*. 2022;2:66–72. Russian. EDN: NAUUTP.
3. Orlov MA. *Pervichnye instrumenty TRIZ* [Primary TIPS tools]. Moscow: Solon-press; 2010. 128 p. Russian.
4. Velchenko SA. The use of a multi-paradigm approach in teaching parallel programming to university students. In: Academy of Postgraduate Education. *Zbornik navukovykh prats Akademii paslyadyplomnai adukatsyi. Vypusk 19* [Collection of scientific works of the Academy of Postgraduate Education. Issue 19]. Minsk: Academy of Postgraduate Education; 2021. p. 96–107. Russian.

Статья поступила в редакцию 08.04.2024.
Received by editorial board 08.04.2024.

¹⁵Коваленко Н. С., Вельченко С. А., Овсеев М. И. Параллельное программирование... 255 с.