

# Международный журнал информационных технологий и энергоэффективности |



Том 5 Номер 1(15)



2020



## СОДЕРЖАНИЕ / CONTENT

### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- 
1. **Балашов О.В., Букачев Д.С.** Подход к оценке качества управленческих решений на основе нечеткой логики **3**

**Balashov O.V., Bukachev D.S.** Approach to Assessing the Quality of Management Decisions on the Basis of Fuzzy Logic

---

2. **Пучков Ю.И.** Корректирующий код с повторением **8**

**Puchkov Yu.I.** The Correction Code with Repeat

---

3. **Кондратова Н.В., Мослякова А.Л.** Подход к оценке ситуативной важности задач объектов организационно-технических систем **14**

**Kondratova V. N, Moslyakova A.L.** Approach to Accessing the Situational Importance of Tasks of Objects of Organizational and Technical Systems

---

4. **Свириденкова М.А., Свириденков К.И.** Тенденции развития Big Data **23**

**Sviridenkova M.A., Sviridenkov K.I.** Big Data Development Trends

---

5. **Федулов Я.А., Федулова А.С.** Анализ использования профилировщиков программного продукта для параллельного кода в системе вычислительного кластера **30**

**Fedulov Ya.A., Fedulova A.S.** The Analyze of a Usage of Profilers for the Parallel Code in a Computer Cluster System

---

### ЭНЕРГОЭФФЕКТИВНОСТЬ

- 
6. **Свириденкова М.А., Свириденков К.И.** Модель оценки эффективности деятельности энергетических организаций **39**

**Sviridenkova M.A., Sviridenkov K.I.** Performance Assessment Model for Energy Organizations

---



Международный журнал информационных технологий и энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 681.3.06

## ПОДХОД К ОЦЕНКЕ КАЧЕСТВА УПРАВЛЕНЧЕСКИХ РЕШЕНИЙ НА ОСНОВЕ НЕЧЁТКОЙ ЛОГИКИ

<sup>1</sup> Балашов О.В., <sup>2</sup> Букачев Д.С.

<sup>1</sup> Смоленский филиал АО «Радиозавод», Россия, (214027, г. Смоленск, улица Котовского, 2), e-mail: smradio@mail.ru

<sup>2</sup> ФГБОУ ВО Смоленский государственный университет, Смоленск, Россия (21400, г. Смоленск, ул. Пржевальского, 4), e-mail: dsbuka@yandex.ru

Подход позволяет сформулировать требования к формированию перечня показателей качества управленческого решения. В качестве интегральной характеристики качества предложено использовать обобщенную степень соответствия данных показателей поставленной цели, определяемую на основе аппарата нечеткой логики и принципа Беллмана-Заде. Предложенный подход, в отличие от методов теории полезности, не требует формирования функций полезности.

Ключевые слова: решение, лингвистическое описание.

## APPROACH TO ASSESSING THE QUALITY OF MANAGEMENT DECISIONS ON THE BASIS OF FUZZY LOGIC

<sup>1</sup> Balashov O.V., <sup>2</sup> Bukachev D.S.

<sup>1</sup> Smolensk branch of joint-stock company "Radio factory", Russia, (214027, Smolensk, street Kotovskogo, 2), e-mail: smradio@mail.ru

<sup>2</sup> Federal State Educational Institution of Higher Education Smolensk State University, Smolensk, Russia (21400, Smolensk, street Przewalski, 4), e-mail: dsbuka@yandex.ru

The approach allows us to formulate requirements for the formation of a list of quality indicators of a managerial decision. It is proposed to use the generalized degree of compliance of these indicators with the set goal as an integral characteristic of quality, determined on the basis of the fuzzy logic apparatus and the Bellman-Zade principle. The proposed approach, in contrast to the methods of the theory of utility, does not require the formation of utility functions.

Keywords: decision, linguistic description.

Управление современными организационно-техническими системами (ОТС) становится все более сложным, к ним предъявляются противоречивые функциональные, технико-экономические, эргономические, экологические и социальные требования, которые характеризуются совокупностью показателей качества. Как правило, показатели качества зависимы между собой и являются антагонистическими. ОТС, как один из видов сложной информационной системы, можно представить упорядоченным набором элементов, свойств и их отношений. Их конкретное задание определяет структуру, характеристики и эффективность системы.

Актуальным является выбор оптимальных решений при долгосрочном планировании и проектировании системы, а также при краткосрочном планировании и ситуативном

управлении с учетом совокупности показателей качества [1, 2]. Здесь имеет место задача принятия решений, для которой характерны следующие основные компоненты: как сформировать множество альтернативных решений, каким ограничениям должны удовлетворять решения, по какому критерию и какими методами должны выбираться наилучшие (рациональные) решения.

Удобно считать, что выбор решений производит некоторое лицо, принимающее решение (ЛПР), которое преследует вполне определенные цели [3]. В зависимости от конкретной ситуации в роли ЛПР может выступать как отдельный человек (инженер, научный сотрудник, заказчик), так и целый коллектив (группа специалистов, занятая решением одной задачи). Каждое возможное решение характеризуется определенной степенью достижения цели. В соответствии с этим у ЛПР имеется свое представление о достоинствах и недостатках решений, на основании которого одно решение предпочитается другому.

Рациональное решение – это решение, которое с точки зрения ЛПР предпочтительнее других альтернативных решений. Это предпочтение на практике может выражаться в различной форме, и его математическая формализация может составить непростую задачу. Сложность заключается в том, что на начальных этапах ЛПР, как правило, не может сформулировать свои предпочтения (рациональность решений) ясно и четко с точки зрения математической формализации.

Предлагаемый подход в своей основе имеет следующие, как представляется, понятные на интуитивном уровне, принципы.

1. *Уникальность экономической операции.* Данный принцип, по сути, констатирует тот факт, что совпадающих экономических операций не бывает. Это означает, в частности, неприменимость для выработки и оценки качества управленческих решений вероятностного подхода (не выполняется условие повторяемости опытов, т. е. вероятностной устойчивости).

2. *Неопределённость условий задачи* описывается и представляется с помощью средств нечёткой логики [4, 5].

3. *Множественность показателей качества (критериев).* Этот принцип отражает необходимость комплексной, всесторонней оценки последствий любого решения, в противном случае задача становится практически тривиальной.

4. *Сочетание качественных (вербальных) и количественных характеристик.* Этот принцип допускает (и рекомендует) комбинировать различные виды моделей описания качества управленческого решения, поскольку далеко не всегда чисто количественный подход (из-за разномасштабности показателей, имеющейся неопределенности и т. п.) позволяет сравнивать различные альтернативы.

5. *Рациональность.* Принцип рациональности отражает введенное выше понятие критерия пригодности.

**Описание подхода.** С учетом приведенных принципов сам подход может быть описан последовательностью следующих шагов.

Шаг 1. ЛПР более высокого уровня иерархии, чем лицо, непосредственно принимающее решение, формирует перечень показателей  $Q_j, j = 1, 2, \dots, n$ , отражающих качество принимаемого решения. При формировании этого перечня необходимо принимать во внимание результативность, ресурсоемкость и оперативность операции, ради реализации которой принимается решение.

Шаг 2. По каждому количественному показателю определяются (оцениваются экспертным путем) минимальные  $Q_{i,\min}$  и максимальные  $Q_{i,\max}$  значения, после чего осуществляется их приведение к одному (единичному) масштабу по следующей формуле:

$$q_i = \frac{Q_i - Q_{i,\min}}{Q_{i,\max} - Q_{i,\min}}, \quad (1)$$

Шаг 3. Каждый нормированный показатель  $y_i$  рассматривается как лингвистическая (нечёткая) переменная [4, 5], имеющая 5–9 уровней (термов). Использование более 9 уровней нецелесообразно, поскольку из психологии известно, что в памяти человека удерживается одновременно не более  $7 \pm 2$  понятий [6]. Такие уровни могут иметь, например, следующую содержательную интерпретацию: 1 – незначительный; 3 – малый; 5 – средний; 7 – большой; 9 – значительный или очень большой; 2, 4, 6, 8 – промежуточные оценки. Например, показатель «убытки» имеет качественные значения «незначительные убытки», «малые убытки», «средние убытки» и т. д.

Нетрудно видеть, что таким образом можно учитывать не только количественные, но и качественные (номинальные) показатели, характеризующие решение.

Вообще говоря, данный шаг является необязательным, но зачастую он облегчает реализацию шага 6.

Шаг 4. Устанавливаются требования к рациональному решению, например: по первому критерию – уровень не меньше 5-го, по второму – не меньше 3-го и т. д.

Шаг 5. ЛПР формирует множество всех возможных вариантов принятия решения (альтернатив)  $A = \{a_1, a_2, \dots, a_m\}$ .

Шаг 6. Экспертным путем производится оценка каждой альтернативы по каждому из введенных показателей так, что для каждого  $j$ -го показателя  $q_j$   $i$ -й альтернативы  $a_i$  определяется степень достижения поставленной цели (степень принадлежности)  $\mu_{ij}$ , при этом должно выполняться условие  $0 \leq \mu_{ij} \leq 1$ .

Шаг 7. Составляется таблица (матрица) с элементами  $\mu_{ij}$  (таблица 1).

Таблица 1 – Матрица для определения наилучшего решения

$a_i$	$q_j$						$\min \mu_{ij}$
	$q_1$	$q_2$	...	$q_j$	...	$q_n$	
$a_1$	$\mu_{11}$	$\mu_{12}$	...	$\mu_{1j}$	...	$\mu_{1n}$	$\mu_1$
$a_2$	$\mu_{21}$	$\mu_{22}$	...	$\mu_{2j}$	...	$\mu_{2n}$	$\mu_2$
...	...	...	...	...	...	...	...
$a_i$	$\mu_{i1}$	$\mu_{i2}$	...	$\mu_{ij}$	...	$\mu_{in}$	$\mu_i$
...	...	...	...	...	...	...	...
$a_m$	$\mu_{m1}$	$\mu_{m2}$	...	$\mu_{mj}$	...	$\mu_{mn}$	$\mu_m$

Последний (правый) столбец данной матрицы содержит значения равные  $\min \mu_{ij}$ , т. е. значения, минимальные по строкам.

Шаг 8. Производится ранжирование вариантов решения (альтернатив) на основе пересечения нечетких множеств – критериев, которые отвечают известной в теории принятия решений схеме Беллмана-Заде [5].

Базируясь на принципе Беллмана-Заде, наилучшей системой будет считаться та, которая одновременно лучше по критериям  $q_1, q_2, \dots, q_n$ . Поэтому нечёткое множество, которое необходимо для рейтингового анализа, определяется в виде пересечения (интегральный критерий оценки качества решения). Учитывая, что в теории нечётких множеств операции пересечения соответствует  $\min$ , получается

$$\mu_i = \min \mu_{ij}. \quad (6)$$

В соответствии с логикой приведенных рассуждений наилучшей будет альтернатива  $a_q$ , для которой величина  $\mu_g$  является наибольшей, т. е.

$$\mu_q = \max \mu_i. \quad (7)$$

Шаг 9. Задается некоторый минимальный уровень  $\mu^*$  степени соответствия интегрального критерия оценки качества решения поставленной перед ЛПР цели ( $0 \leq \mu^* \leq 1$ ), и проверяется неравенство

$$\mu^* \leq \mu_q. \quad (8)$$

В случае его выполнения выявленная наилучшая альтернатива признается пригодной.

*Качество решений в приведенной процедуре оценивается степенями соответствия  $\mu_i$  альтернатив поставленной цели.*

*Замечания*

1. *Неравновесные критерии.* Пусть заданы  $v_1, v_2, \dots, v_L$  – коэффициенты относительной важности (или ранги) критериев  $q_1, q_2, \dots, q_n$  такие, что  $v_1 + v_2 + \dots + v_L = 1$ , при этом значения  $v_1 - v_n$  считаются заданными. При наличии коэффициентов важности соотношение (6) принимает вид

$$\mu_i = \min (\mu_{ij})^{v_j}. \quad (9)$$

Окончательный выбор альтернативы осуществляется, как и выше, с помощью соотношения (7).

2. *Применение принципа Беллмана-Заде, использующее в (6) операцию взятия минимума, во многих случаях, а именно, при большом числе критериев, может привести к тому, что итоговые степени принадлежности будут весьма близки к нулю, что затруднит их сравнение в соответствии с (7) и сделает общие выводы ненадежными.* В этой связи представляется целесообразным операцию пересечения реализовывать не как операцию взятия минимума, а как операцию нахождения медианного значения из ряда имеющихся после их упорядочивания, в частности, в порядке возрастания.

Иначе говоря, имеет смысл отказаться от идеи выбирать в качестве интегральной оценки общую часть (пересечение) всех частных оценок, заменив её более гибким и продуктивным принципом выбора в качестве интегральной той частной оценки, которую дает некоторый специально сконструированный «наиболее представительный эксперт» [7].

Необходимо отметить, что такой «эксперт» должен в каждой точке области всех возможных альтернатив выбирать в качестве меры принадлежности этой точки интегральной оценке ту из мер ее принадлежности частным оценкам, которая в общем случае удалена от крайних оценок и занимает некоторое «среднее» (медианное) положение. Такой выбор означает, что объединение частных оценок в интегральную производится не по правилу пересечения нечётких множеств (где берется минимальная из оценок меры принадлежности) или по правилу их объединения (берется максимальная из оценок), или в соответствии с любой другой известной операцией над нечёткими множествами, а представляет собой некоторую новую операцию над такими множествами, а именно их упорядоченный набор с выбором затем одного из элементов такого упорядоченного набора.

Пример использования такого подхода: пусть имеем ряд степеней принадлежности  $\{0.2, 0.4, 0.7, 0.3, 0.1, 0.9, 0.6\}$ .

После упорядочивания получаем {0.1, 0.2, 0.3, 0.4, 0.6, 0.7, 0.9}. Если использовать операцию взятия минимума, то её итогом будет значение 0.1; при использовании медианы получим значение 0.4.

Отметим, что близкий подход теоретически обоснован в статье [8] и, по-видимому, нуждается в дальнейшем исследовании.

### Список литературы

1. Балашов О.В., Букачев Д.С. Метод автоматизированного оперативного управления социально-экономическими системами // Международный журнал информационных технологий и энергоэффективности. – 2018. – Т.1, № 2, с. 8-13.
2. Анфилатов В.С., Емельянов А.А., Кукушкин А.А. Системный анализ в управлении: Учеб. пособие. М.: Финансы и статистика, 2002. – 368 с.
3. Ногин В.Д. Принятие решений в многокритериальной среде. – СПб.: Физматлит, 2002. – 176 с.
4. Борисов В. В., Круглов В. В., Федулов А. С. Нечёткие модели и сети. М., Горячая линия – Телеком, 2007. – 284 с.
5. Беллман Р., Заде Л. Принятие решений в расплывчатых условиях/ В кн.: Вопросы анализа и процедуры принятия решений. М.: Мир, 1976. С. 172-215.
6. Miller G. A. The Magic Number Seven plus or Minus Two: Some Limits on Our Capacity for Processing Information // Psychological Review. 1956. № 63. P. 81–97.
7. Литвак Б. Г. Экспертная информация: методы получения и анализа. М., Радио и связь, 1982. – 184 с.
8. Левин В.И. Новое обобщение операций над нечеткими множествами // Изв. РАН – Теория и системы управления. 2001. № 1. С. 143-146.

### References

1. Balashov O.V., Bukachev D.S. Metod avtomatizirovannogo operativnogo upravleniya social'no-ekonomicheskimi sistemami// Mezhdunarodnyj zhurnal informacionnyh tekhnologij i energoeffektivnosti. – 2018. – Т.1, № 2, s. 8-13.
  2. Anfilatov V.S., Emel'yanov A.A., Kukushkin A.A. Sistemnyj analiz v upravlenii: Ucheb. posobie. M.: Finansy i statistika, 2002. – 368 s.
  3. Nogin V.D. Prinyatie reshenij v mnogokriterial'noj srede. – SPb.: Fizmatlit, 2002. – 176 s.
  4. Borisov V. V., Kruglov V. V., Fedulov A. S. Nechyotkie modeli i seti. M., Goryachaya liniya – Telekom, 2007. – 284 s.
  5. Bellman R., Zade L. Prinyatie reshenij v rasplyvchatyh usloviyah/ V kn.: Voprosy analiza i procedury prinyatiya reshenij. M.: Mir, 1976. S. 172-215.
  6. Miller G. A. The Magic Number Seven plus or Minus Two: Some Limits on Our Capacity for Processing Information // Psychological Review. 1956. № 63. P. 81–97.
  7. Litvak B. G. Ekspertnaya informaciya: metody polucheniya i analiza. M., Radio i svyaz', 1982. – 184 s.
  8. Levin V.I. Novoe obobshchenie operacij nad nechetkimi mnozhestvami // Izv. RAN – Teoriya i sistemy upravleniya. 2001. № 1. S. 143-146.
-



Международный журнал информационных технологий и энергоэффективности

Сайт журнала:

<http://www.openaccessscience.ru/index.php/ijcse/>



УДК 681.3.042

## КОРРЕКТИРУЮЩИЙ КОД С ПОВТОРЕНИЕМ

**Пучков Ю.И.**

Филиал ФГБУ ВО «НИУ «МЭИ» в г. Смоленске, Смоленск, Россия  
(214013, г. Смоленск, Энергетический проезд, 1), e-mail: puchkov304@mail.ru

В статье рассматривается построению и декодированию кода, представляющего собой корректирующий код, исправляющий ошибки, и его повторение. Полученный код исправляет вдвое больше ошибок, чем исходный код. Полученный код, по сравнению с кодами аналогичными по корректирующей способности, незначительно проигрывает в скорости, но имеет существенно более простой алгоритм декодирования.

Ключевые слова: код, кодирование, декодирование, корректирующая способность, конструктивное расстояние, скорость кода.

## THE CORRECTING CODE WITH REPEAT

**Puchkov Yu.I.**

Smolensk Branch of the National Research University "Moscow Power Engineering Institute",  
Smolensk, Russia (214013, Smolensk, Energeticheskyy proezd, 1), e-mail: puchkov304@mail.ru

The article looks at building and decoding code, which is a corrective code that corrects errors, and repeats it, corrects twice as many errors as the source code. The resulting code, compared to codes similar in corrective ability, slightly loses in speed, but has a significantly simpler decoding algorithm

Keywords: code, coding, decoding, corrective ability, constructive distance, code speed

Коды с повторением всегда рассматривались как один из вариантов построения корректирующих кодов, обнаруживающих одиночные ошибки и какую-то часть ошибок большей кратности. Такие коды применяются в системах, где эффективность использования канала связи не является актуальной. Например, в системах автоматики и телемеханики на железнодорожном транспорте [1]. В зависимости от типа ошибок в канале связи применяют повторение прямо или инверсно всей комбинации кода или каждого символа исходного кода. Декодирование кода заключается в сравнении обеих частей кода (инверсия второй части кода, если она применялась, предварительно восстанавливается). Если одноимённые разряды обеих частей кода не совпадают, то это воспринимается как обнаружение ошибки.

В корректирующем систематическом коде выделяют  $k$  информационных и  $m$  проверочных разрядов. Общая длина кодовой комбинации  $n = k + m$ . Скорость кода  $R = k/n$ . Число информационных разрядов выбирают из условия  $M \leq 2^k$ . Если выполняется равенство, то код будет оптимальным, а если неравенство, то код называют укороченным. Передаваемая информация заключена только в информационных разрядах кода. В кодах с повторением эти разряды передаются дважды (в каждой половине кода). Если знать в какой половине кода информационные разряды не содержат ошибки или ошибка в них может быть легко

исправлена, то эти разряды следует передать потребителю, предварительно исправив возможную ошибку. Это упрощает алгоритм декодирования кода с повторением.

Код с повторением является корректирующим. Он состоит из двух частей. Длина  $n = n_1 + n_2$ , где  $n_1$  и  $n_2$ , соответственно, длины первой и второй его частей. Обычно  $n_1 = n_2$ . За основу берётся некоторый исходный код, являющийся первой частью кода с повторением, и затем он повторяется, образуя вторую часть кода. Известно [2], что корректирующая способность кода определяется его минимальным кодовым расстоянием  $d_{\min}$ , называемое также конструктивным расстоянием или просто «кодовым расстоянием». Если исходный код будет иметь кодовое расстояние  $d_{\min}$  то, построенный на его основе код с повторением будет иметь уже расстояние  $2d_{\min}$ . Следовательно, его корректирующая способность будет практически вдвое выше, чем у исходного кода. Если код предназначен для исправления  $s$  ошибок, то его конструктивное расстояние должно быть равно  $d_{\min} = 2s + 1$ , то есть будет нечётным. Как следствие конструктивное расстояние кода с повторением окажется чётным. Что бы оно оказалось нечётным, необходимо увеличить конструктивное расстояние одной половины кода на 1, что всегда можно сделать, добавив в ней один разряд с проверкой на чётность.

Рассмотрим пример. Положим, исходная комбинация представляет собой 5-и разрядную комбинацию избыточного кода – 10110. Полученная на её основе комбинация кода с повторением будет иметь вид 1011010110. Если она принята без ошибки, то результат сравнения частей кода (поразрядное сложение по модулю 2 первой и второй частей кода) даст нулевую комбинацию  $10110 + 10110 = 00000$ . Если в одной из частей кода один разряд был принят не верно, то в результате сравнения частей кода получим в соответствующем разряде 1. Но в какой части кода произошла эта ошибка мы определить не сможем. Известно, что для кода, обнаруживающего  $r$  ошибок, должно выполняться условие  $d_{\min} = r + 1$ . У рассматриваемого кода с повторением  $d_{\min} = 2$  и, следовательно, получаем  $r = 1$ . Добавим, например, во вторую часть кода один разряд проверки на чётность. Получим комбинацию 10110101101. На приёмной стороне осуществляется проверка на чётность второй половины кода. Если в этой части кода ошибок не было, то результат проверки на чётность будет нулевым, а если произошла одиночная ошибка, то – 1. Теперь при наличии одиночной ошибки в результате сравнения частей кода получим комбинацию из нулей и одной единицы. Разряд проверки на чётность в операции сравнения не участвует. Положение единицы укажет номер искажённого разряда, а результат проверки на чётность позволит указать, в какой половине кода произошла ошибка. Для двоичного кода исправление ошибки сводится к инверсии указанного разряда кода. Если результат проверки на чётность показывает её нарушение, а результат сравнения частей кода оказался нулевым, то это свидетельствует об ошибке именно в разряде проверки на чётность.

Видим, что достаточно просто построили код, позволяющий исправлять одиночные ошибки, при простом алгоритме декодирования. Обратим внимание на то обстоятельство, что ошибку в принципе даже не надо исправлять, а передать получателю лишь неискажённую часть кода. Заметим, что при известной безошибочной части кода на приёмной стороне можно заново сформировать, если в этом есть необходимость, переданную комбинацию кода с повторением. Указанные особенности декодирования и исправления ошибок будут справедливы и для других типов исходных кодов.

В качестве исходного кода возьмём корректирующий код. В результате получим корректирующий код с повторением (ККП). Нецелесообразно выбирать исходный код с чётным конструктивным расстоянием, поскольку у построенного на его основе ККП конструктивное расстояние будет так же чётным. Это не позволит приведённое выше неравенство, связывающее кратность исправляемых ошибок с конструктивным расстоянием

кода, заменить на равенство. Добавим во вторую часть кода ещё один разряд проверки на чётность, тем самым увеличим не 1 её кодовое расстояние  $d_{\min}$  и оно станет чётным.

Рассмотрим процесс кодирования и декодирования ККП, взяв в качестве исходного кода код с кодовым расстоянием  $d_{\min} = 3$ , а именно, циклический код Хэмминга, исправляющий одиночные ошибки. Декодирование его осуществляется достаточно просто, поскольку к моменту окончания кодовой комбинации уже оказывается сформированной кодовая комбинация синдрома, указывающая на номер искажённого разряда. Для исправления ошибки в этом разряде достаточно подать один импульс исправления ошибки, то есть время исправления ошибки оказывается равным длине импульса считывания, которая может быть во много раз короче длительности одного импульса кодовой комбинации.

Как и в предыдущем случае ко второй части кода добавим разряд проверки на чётность. Кодовое расстояние этой части кода станет равным 4. Получим ККП с кодовым расстоянием равным 7. Код с таким кодовым расстоянием сможет исправлять тройные ошибки. То есть исходный код может исправлять только одиночные ошибки, а построенный на его основе ККП позволит исправлять уже трёхкратные ошибки. Исправление ошибок кратности большей 1 представляется достаточно сложной задачей и выполняется за ряд тактов после приёма всей кодовой комбинации.

Особенностью ККП является то обстоятельство, что исправляемые кодом ошибки распределяются между двумя его частями. И всегда найдётся часть кода, ошибки в которой не превышают корректирующей способности исходного кода. В данном случае всегда найдётся часть кода, в которой не больше одиночной ошибки, и она практически мгновенно исправляется. Тройная ошибка может поразить либо только одну часть кода, а другая часть кода безошибочной, либо в одной части кода будет одиночная ошибка, а в другой – двойная. Очевидно, одиночная ошибка и исправляется. Осталось лишь определить, какая часть кода безошибочна или имеет только одиночную ошибку. То есть ошибки кратностью больше единицы фактически не исправляются. Если возникнет необходимость безошибочного восстановления всей кодовой комбинации, то это всегда можно осуществить, имея неискажённую половину кода.

Алгоритм декодирования рассматриваемого ККП заключается в следующем. Для каждой части вычисляются их синдромы  $S1$  и  $S2$ . Если ошибки в рассматриваемой части нет, то синдром будет равен 0, а если ошибка (одиночная или двойная) присутствует, то синдром будет отличным от нуля (обозначим такие его значения символом -1). Осуществляется проверка на чётность  $P$  второй половины кода. В таблице1 рассмотрены возможные распределения ошибок по частям кода, значение синдромов  $S1$ ,  $S2$  при декодировании соответствующей части кода, результат проверки на чётность второй части кода.

Таблица1 – Пояснение к декодированию кода с  $d_{\min} = 3$ .

Кратность ошибок		0	1		2		3				
Распределение ошибок по частям кода	1ч.	0	1	0	2	0	1	3	0	2	1
	2ч.	0	0	1	0	2	1	0	3	1	2
Значение синдромов частей кода	$S1$	0	1	0	1	0	1	0	0	1	1
	$S2$	0	0	1	0	1	1	0	0	1	1
Результат проверки на чётность	$P$	0	0	1	0	0	1	0	1	1	0
Декодируемая часть кода		1,2	2	1	2	1	1,2	2	1	2	1

Учтено, что трёхкратная ошибка воспринимается декодирующим устройством как безошибочная, но другая (не переданная в данный момент) кодовая комбинация. В нижней строке таблицы указано, какую часть кода следует декодировать при соответствующем распределении ошибок по частям кода. Если указано 1,2, то это означает, что можно декодировать любую часть кода.

Из таблицы можно заметить, что в зависимости от ситуации с распределением ошибок по частям кода следует декодировать либо первую, либо вторую часть кода. Декодируемая половина кода или не содержит ошибок, или содержит только одиночную ошибку. Из анализа таблицы следует, что в случае совпадения чётностей у  $P$  и  $S_2$ , следует декодировать только вторую часть кода, а в остальных случаях – первую.

Рассмотренный пример позволяет сделать следующие выводы. Корректирующая способность рассмотренного ККП вдвое больше, чем у исходного кода. Ошибки исправляются не во всей кодовой комбинации, а лишь в одной её части, и кратность их не превышает корректирующей способности исходного кода. При необходимости на приёмной стороне можно заново сформировать всю переданную комбинацию ККП, то есть, фактически исправить все ошибки в принятой кодовой комбинации. Такие коды с относительно невысокой корректирующей способностью и простым, практически не требуемым времени для декодирования можно применять в оперативных системах управления [3], в информационно-управляющих комплексах [4], в системах контроля [5].

Технически просто решается задача декодирования БЧХ – кодов, исправляющих двукратные ошибки, если число информационных разрядов кода хотя бы на единицу было меньше числа проверочных разрядов. Этому условию удовлетворяет код (15,7). Для его декодирования используют декодер Мегитта [6]. Конструктивное расстояние этого кода равно 5. На его основе можно построить ККП, с кодовым расстоянием равным 11. Код позволит исправлять 5-и кратные ошибки. Правило декодирования: исправлять ошибки следует во второй части кода, если результат проверки на чётность совпадает с чётностью ошибки этой части кода, если не совпадает, то исправление ошибок осуществляется в первой части кода.

Декодирование кодов, исправляющих ошибки кратности больше 1 (кроме кода (15,7)), осуществляется более сложно. Вначале вычисляются синдромы, число которых соответствует максимальной кратности исправляемых кодом ошибок, на их основе составляется система нелинейных уравнений. Определяется ранг синдромной матрицы. Оставляют число уравнений равное рангу матрицы и находят их решение. По результату решения производят исправление ошибок.

Для исходных кодов, исправляющих более одной ошибки, при формировании правила декодирования ККП следует учитывать ранг матрицы синдромов. Кроме того, необходимо принять во внимание, что если число ошибок в декодируемой половине кода будет превышать её корректирующие возможности, то она будет декодирована в другую кодовую комбинацию. Например, код позволяет исправлять 5-и кратные ошибки, его  $d_{\min} = 11$ . При 7-и кратной ошибке принятая кодовая комбинация будет декодирована в другую комбинацию, которая, как бы передавалась, и в ней произошло 4-е ошибки ( $11 - 7 = 4$ ), которые кодом и будут исправлены.

Построим на основе этого кода ККП. Такой код позволит исправлять 11-и кратные ошибки. Для определения правила выбора части кода, которую при приёме надлежит декодировать, необходимо проанализировать возможное распределение ошибок по частям кода. Фрагмент распределения при 11-и кратной ошибки приведён в таблице 2. Аналогичные таблицы были составлены и для других кратностей ошибок.

Таблица 2 – Пояснение к декодированию кода с  $d_{\min} = 23$  при 11-и кратной ошибке.

Кратность ошибок		11											
Кратность ошибок в частях кода	1ч	11	0	10	1	9	2	8	3	7	4	6	5
	2ч	0	11	1	10	2	9	3	8	4	7	5	6
Ранг синдромов частей кода	S1	0*	0	1*	1	2*	2	3*	3	4*	4	5*	5
	S2	0	0*	1	1*	2	2*	3	3*	4	4*	5	5*
Результат проверки на чётность	P	0	1	1	0	0	1	1	0	0	1	1	0
Декодируемая часть кода		2	1	2	1	2	1	2	1	2	1	2	1

Проведённый анализ позволяет сделать вывод, что в случае, если чётности ранга матрицы синдромов во второй части кода и результата проверки на чётность совпадают, то декодировать следует вторую часть кода. Если это условие не выполняется, то надлежит декодировать первую часть кода.

Сравним по скорости аналогичные по корректирующей способности коды ККП и БЧХ-код. Сравнение проведём для частного случая, выбрав в качестве исходного корректирующего кода БЧХ – код (31,11), позволяющий исправлять 5-и кратные ошибки. На его основе построим ККП код (63,11). В этом коде имеется 11 информационных разрядов, а все остальные разряды, включая полностью вторую часть кода, следует отнести к проверочным. Для сравнения возьмём также укороченный БЧХ – код (58,11), полученный из полного БЧХ - кода (63,16), имеющий такое же число информационных символов и позволяющий исправлять 11-ти кратные ошибки. Скорость ККП - кода (63,11)  $R = 0.175$ , а укороченного БЧХ – кода (58,11)  $R = 0.19$ , что практически соизмеримо.

Поскольку рассматриваемые коды имеют одинаковую корректирующую способность и примерно одинаковую скорость, то можно утверждать, что они имеют одинаковую вероятность возникновения ошибок. При малых по абсолютному значению вероятностях возникновения ошибки отличия меньше порядка обычно не принимаются во внимание.

Время для декодирования обычно ограничено длительностью одного символа кода. Очевидно, что с увеличением порядка уравнений существенно вырастает число арифметических операций. Число операций при декодировании БЧХ - кода оценивается как  $n^2 \log_2 n$  [7]. В коде ККП с такой же корректирующей способностью ошибки распределены по двум практически равным по длине  $n/2$  частям кода. Число операций для одной части кода будет равно  $(n/2)^2 \log_2 (n/2) = (1/4) n \log_2 (n-1)$ , то есть в 4 раза меньше. Вычисление ранга матрицы опознавателей первой части ККП кода выполняется в процессе приёма второй части кода и на быстрдействие процесса декодирования не оказывает влияния.

Коды ККП могут найти применение в автономных устройствах, где остро не стоит вопрос эффективного использования канала связи. Например, в системах передачи телемеханической информации [8], в системах обеспечения надёжности устройств [9]. Коды могут найти применение и в устройствах реального времени, где важным являются скорость декодирования и простота технической реализации декодирующих устройств при числе исправляемых ошибок не более пяти.

### Список литературы

1. Теория передачи сигналов на железнодорожном транспорте: учебник / Г.В. Горелов и др.; под ред. Г.В. Горелова. – М.: ФГБОУ «Учебно-методический центр по образованию на железнодорожном транспорте», 2013. — 532с.
2. Основы теории кодирования. Б. Д. Кудряшов Издательство: БХВ-Петербург, Учебники и пособия для вузов, 2016.
3. К вопросу о целесообразности применения корректирующих кодов с исправлением ошибок в командных радиоперелиниях управления. Ашимов Н.М., Шустик Н.А. Спецтехника и связь. 2009. № 3. С. 42-45.
4. Алгоритмы декодирования избыточных кодов в системе информационно-управляющих комплексов. Гладких А.А., Наместников С.М., Чилихин Н.Ю. REDS: Телекоммуникационные устройства и системы. 2015. Т. 5. № 1. С. 105-108.
5. Особенности применения кодов Хэмминга при организации самопроверяемых схем встроенного контроля. Сапожников В.В., Сапожников Вл.В., Ефанов Д.В., Известия высших учебных заведений. Приборостроение. 2018. Т. 61. № 1. С. 47-59.
6. Кларк Дж., мл., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи: Пер. с англ. – М.: Радио и связь, 1987. – 392 с.
7. Теория кодирования / Т. Касами, Н. Токура, Е. Ивадари, Я. Ирагаки. – М.: Мир, 1978.
8. Хлынов А.А. Помехоустойчивое кодирование в каналах передачи телеметрической информации // Труды Научно-исследовательского института радио. 2015. № 2. С. 27-31.
9. Применение корректирующих способностей кодов для обеспечения отказоустойчивости. Петлеванный С.В., Сагдеев А.К. Современные наукоемкие технологии. 2007. № 4. С. 41-42

### References

1. The theory of railway transmission: textbook/ G.V. Gorelov et al.; Ed. G.V. Gorelova. - Yuri Miner, 2013. 532с.
  2. The basics of coding theory. B.D. Kudryashov Publishing House: BKH-Petersburg Genre: Mathematics, Programs, Textbooks and University Manuals, 2016, -400s
  3. On the issue of the appropriateness of using corrective codes with error correction in command radio control lines. Ashimov N.M., Shustik N.A. Special equipment and communications. 2009. No. 3. P. 42-45.
  4. Algorithms for decoding redundant codes in the system of information management systems. Gladkikh A.A., Namestnikov S.M., Chilikhin N.Yu. REDS: Telecommunication devices and systems. 2015. Vol. 5. No. 1. P. 105-108.
  5. Features of the application of Hamming codes in the organization of self-checking schemes of built-in control. Sapozhnikov VV, Sapozhnikov VI.V., Efanov D.V. Izvestiya Vysshikh Uchebnykh Zavedenii. Instrument making. 2018. V. 61. No. 1. S. 47-59.
  6. Clark J., Jr., Kane J. Error correction coding in digital communication systems: Per. from English - M.: Radio and communications, 1987. - 392s.
  7. The coding theory / T. Kasami, N. Tokura, E. Iwadari, Y. Iragaki. — M.: World, 1978.
  8. Khlynov A.A. Noise-resistant coding in telemetry information transmission channels // Transactions of Radio Research Institut, 2015. No. 2. P. 27-31
  9. The use of corrective capabilities of codes to ensure fault tolerance. Petlevanny S.V., Sagdeev A.K. Modern high technology. 2007. No.4. P. 41-42
-



ОТКРЫТАЯ НАУКА  
издательство

Международный журнал информационных технологий и энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 519

## ПОДХОД К ОЦЕНКЕ СИТУАТИВНОЙ ВАЖНОСТИ ЗАДАЧ ОБЪЕКТОВ ОРГАНИЗАЦИОННО-ТЕХНИЧЕСКИХ СИСТЕМ

**Кондратова Н.В., Мослякова А.Л.**

*Смоленский филиал АО «Радиозавод», Россия, (214027, г. Смоленск, улица Котовского, 2),  
e-mail: smradio@mail.ru*

При оперативном управлении организационно-технической системой возникает задача оценки важности её объектов для приоритетного снабжения их различного рода ресурсами. В современной теории управления отсутствуют подходы к оценке ситуативной важности объектов. В основе предлагаемого подхода к оценке ситуативной важности объектов лежит оценка важности через анализ важности задач выполняемых объектами в рассматриваемый момент времени по показателям соответствия функциональному предназначению, вклада в возможность выполнения общей задачи и причинно-следственной важности задачи.

Ключевые слова: важность, задача, лингвистическое описание.

## APPROACH TO ACCESSING THE SITUATIONAL IMPORTANCE OF TASKS OF OBJECTS OF ORGANIZATIONAL AND TECHNICAL SYSTEMS

**Kondratova V. N, Moslyakova A.L.**

*Smolensk branch of joint-stock company "Radio factory", Russia, (214027, Smolensk, street Kotovskogo, 2), e-mail: smradio@mail.ru*

In the operational management of the organizational and technical system, the task arises of assessing the importance of its facilities for the priority supply of their various kinds of resources. In modern control theory, there are no approaches to assessing the situational importance of objects. The proposed approach to assessing the situational importance of objects is based on an assessment of importance through an analysis of the importance of tasks performed by objects at a given point in time according to indicators of compliance with their functional purpose, contribution to the ability to complete the general task and the cause and effect importance of the task.

Keywords: importance, task, linguistic description.

При оперативном управлении объектами организационно-технической системы (ОТС) возникает задача оценки важности данных объектов для приоритетного снабжения их различного рода ресурсами. Результаты исследований показали, что оценка важности объектов должна быть ситуативной, зависящей от важности выполняемых объектами задач (действий) [1]. В основе предлагаемого подхода к оценке ситуативной важности объектов лежит оценка, получаемая в результате анализа важности задач, выполняемых объектами ОТС в рассматриваемый момент времени. Данная оценка включает (рисунок 1):

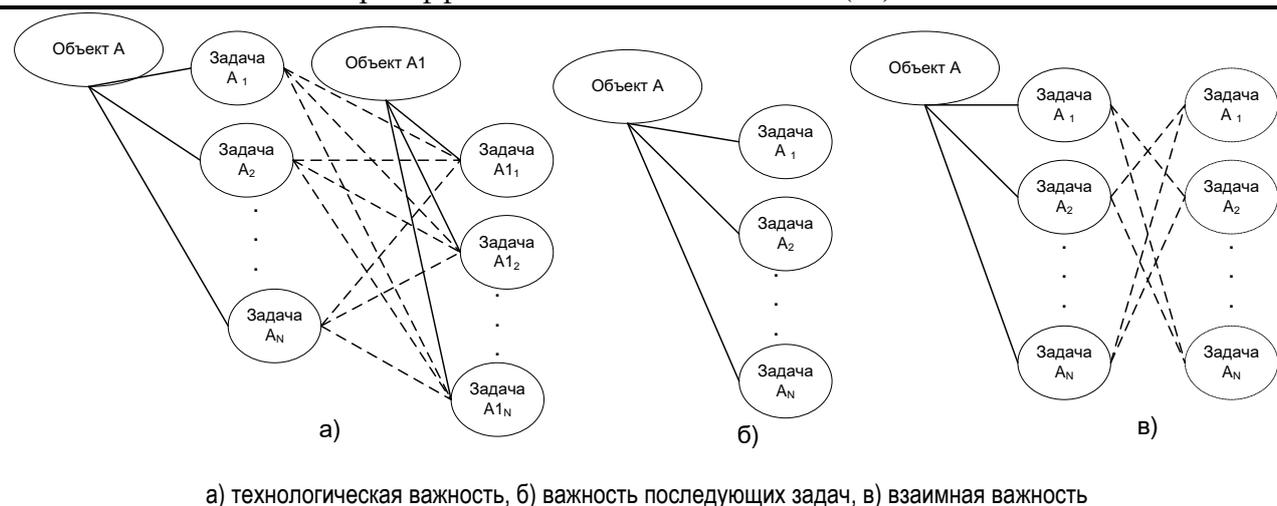


Рисунок 1 – Графическое представление способа определения различных классов важности задач объектов

- оценку важности задач, исходя из их соответствия функциональному предназначению объекта;
- оценку важности частных задач, исходя из анализа их вклада в возможность выполнения общей задачи, в технологию выполнения которой входят рассматриваемые задачи;
- оценку причинно-следственной важности задачи, как степени необходимости выполнения этой задачи после выполнения предыдущей относительно других задач, которые может выполнить объект;

В основе упорядочения типовых задач по важности лежит использование универсальной логико-лингвистической шкалы оценки важности типовых задач, приведенной в таблице 1.

Таблица 1 – Шкала для упорядочения задач объекта по важности

Качественная оценка	Числовая оценка	Классификационное определение важности задачи
Очень высокая	(0,8...1,0)	Задача определяет функциональное предназначение объекта
Высокая	(0,6...0,8)	Невыполнение задачи делает невозможным выполнение основных задач объекта, определяемых функциональным назначением
Средняя	(0,4...0,6)	Задача оказывает существенное, но не определяющее влияние на выполнение основных задач объекта
Низкая	(0,2...0,4)	Задача оказывает несущественное влияние на выполнение основных задач объекта
Очень низкая	(0...0,2)	Задача не связана с выполнением основных задач объекта (вспомогательная)

Использование шкалы обеспечивает упорядочение типовых задач различных объектов по одним и тем же классификационным определениям, качественным и количественным оценкам, что в свою очередь делает возможным сравнение этих задач.

Содержание методики упорядочения задач объекта по важности включает следующие элементы: проведение экспертного опроса для определения оценки важности задач; сравнение мнений экспертов по каждой задаче; формирование ЛЛШ оценки важности задач.

**Оценка технологической важности частных задач.** Для объектов, имеющих подобъекты, при рассмотрении модели задач ОТС выделяются отношения между задачами подобъектов ОТС. Одним из таких отношений является отношение включения, позволяющее

представить в формализованном виде технологию выполнения задачи любого объекта на уровне задач его подобъектов. Каждой задаче  $Z_{Q_j}$  объекта  $Q_j$  может быть поставлен в соответствие кортеж

$$Z_{Q_j} = \langle Z_{Q_i}, N^T(Z_{Q_i}) \rangle, \quad (1)$$

где  $N^T(Z_{Q_i})$  – технологическая важность задачи  $Z_{Q_i}$ , входящей в технологию выполнения задачи  $Z_{Q_j}$  объекта  $Q_j$  (объект  $Q_i$  является управляемым объектом для объекта  $Q_j$ ).

Значение показателя технологической важности задачи  $N^T(Z_{Q_i})$  определяется экспертным путем. Результатом является построение логико-лингвистической шкалы оценки технологической важности задачи  $Z_{Q_i}$  объекта  $Q_i$  для задачи  $Z_{Q_j}$ , объекта  $Q_j$  (рисунок 2).

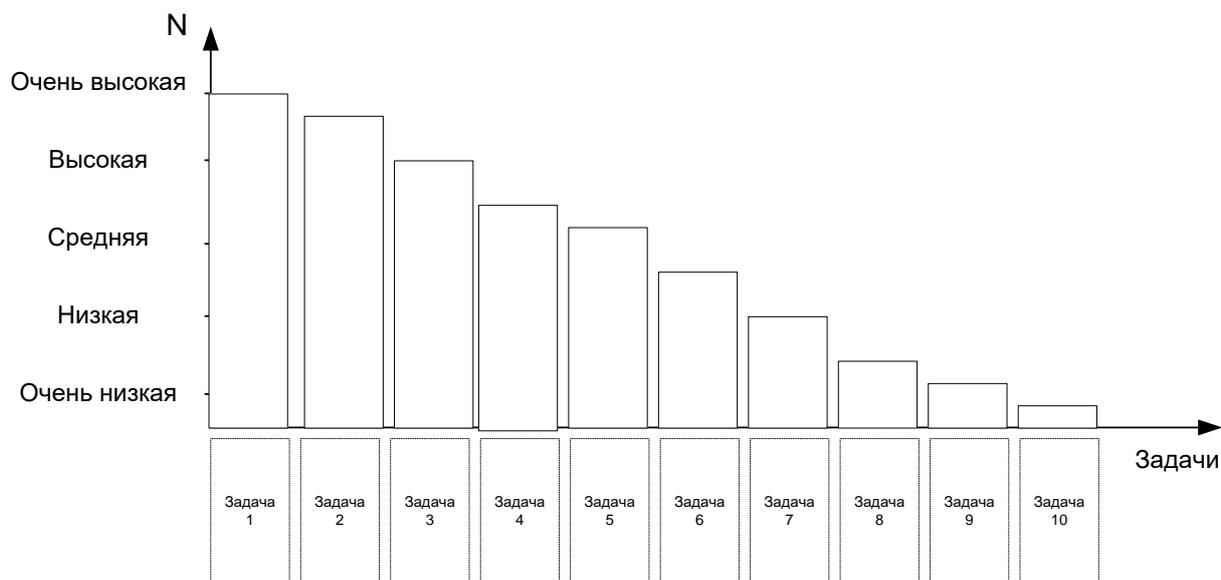


Рисунок 2 – Логико-лингвистическая шкала оценки технологической важности

ЛЛШ оценки технологической важности обеспечивает количественную и качественную оценку показателей  $N^T(Z_{Q_i})$  для каждой из множества задач, которые способен выполнять объект  $Q_i$ , исходя из своего функционального предназначения. Технологическая важность  $N^T(Z_{Q_i})$  задачи  $Z_{Q_i}$  характеризует степень соответствия её наилучшего результата требуемому результату выполнения задачи  $Z_{Q_{ij}}$  объекта  $Q_j$ , в технологию выполнения которой входит задача  $Z_{Q_i}$ .

Показатель  $N^T(Z_{Q_i})$  необходим при разработке алгоритмов определения общей задачи объекта на основе известных задач его подобъектов (процесс прогнозирования и оценки обстановки).

В основе оценки важности частных задач лежит анализ их включения в технологию выполнения некоторой общей задачи выполняемой объектом (надобъектом), в состав которого входят объекты, выполняющие частные задачи [1]. Если частная задача объекта не входит в технологию выполнения общей задачи, то величина её вклада в возможность выполнения общей задачи равна нулю. Таким образом, формируется множество задач объекта  $\{Z_i^j\}$ , а каждой задаче из этого множества соответствует оценка её важности  $N(Z_i^j)$ .

Оценка вкладов частных задач объектов в возможности надобъекта по выполнению общей задачи производится с помощью соответствующей модели оценки возможностей надобъекта по выполнению общей задачи. При этом оценивается возможность выполнения требуемого результата общей задачи при условии, что в состав надобъекта входит один единственный объект, выполняющий частную задачу. Получение требуемого результата

общей задачи рассматривается как единица, относительно которой измеряется величина вклада, который вносит в выполнение общей задачи надобъекта некоторый объект при выполнении своей частной задачи. Иными словами, производится соотношение результата выполнения частной задачи к требуемому результату выполнения общей задачи. В результате оценки вкладов, вносимых всеми объектами в возможности надобъекта по выполнению общей задачи, формируется множество  $\{N^T(Z_{Qi})\}$ .

Важность каждого объекта, входящего в состав надобъекта в рассматриваемый момент времени, определяется как

$$N_i = N(Z_i^j) N_D(z_i^j) \quad (2)$$

**Оценка причинно-следственной важности задач.** Для каждой задачи  $Z_{Qi}$  объекта  $Q_i$  существует множество задач  $\{Z_{Qi}^n\}$ , которые могут выполняться объектом  $Q_i$  после выполнения им задачи  $Z_{Qi}$ . Рассматриваемые задачи связаны отношением «последующие». Каждой задаче  $Z_{Qi}$  может быть поставлен в соответствие кортеж

$$Z_{Qi}^* = \langle Z_{Qi}^n, N^n(Z_{Qi}^n) \rangle, \quad (3)$$

где  $N^n(Z_{Qi}^n)$  – причинно-следственная важность последующей задачи  $Z_{Qi}^n$ , которую может выполнить объект  $Q_i$  после выполнения задачи  $Z_{Qi}$ .

Причинно-следственная важность задачи  $Z_{Qi}^n$  характеризует степень необходимости выполнения этой задачи после выполнения задачи  $Z_{Qi}$  относительно других задач, которые может выполнить объект  $Q_i$ . Следовательно, для расчёта значения показателя  $N^n(Z_{Qi}^n)$  требуется выяснить степень взаимовлияния системных факторов и выявить причинно-следственные отношения между ними с учётом воздействия на эти факторы. Для решения подобных задач в [2] была предложена модификация нечётких когнитивных карт Коско. Отношения между концептами нечёткой когнитивной карты данного типа представлены, как и в модели Коско, в виде весов  $[-1, 1]$ , но они рассматриваются как элементы нечёткой матрицы смежности для графа нечёткой когнитивной карты (НКК). На основе этой матрицы могут быть рассчитаны основные системные показатели нечёткой когнитивной карты, по которым определяется взаимный консонанс, диссонанс, положительное и отрицательное влияние концептов друг на друга и на систему в целом и другие системные и интегральные показатели, используемые для исследования свойств анализируемой системы [2]. Анализ данных НКК позволяет сделать следующие выводы.

Во-первых, выбор в качестве результата максимального из отдельных влияний концептов полностью исключает учет меньших по значению влияний, что снижает чувствительность результата к входным воздействиям и плохо согласуется с физическим смыслом используемых отношений между концептами.

Во-вторых, в рассмотренной модели для описания влияния не в полной мере реализованы постулаты нечёткого подхода, рассматривающего нечёткие множества, по двум «координатам»: базовое множество значений и принадлежность к базовому множеству. Как и в модели Коско, в модифицированных нечётких когнитивных картах одна «координата» не учитывается.

Стремление максимально использовать постулаты теории нечётких множеств привело к появлению нечётких когнитивных карт, для описания влияний между концептами в которых используются нечёткие продукционные правила [3]. В этих картах концепты представлены в виде нечётких множеств, определяемых функциями принадлежности к базовому множеству.

Причинно-следственные отношения между двумя концептами выражены в виде нечёткого продукционного правила со структурой «один вход – один выход» относительно нечётких приращений концептов. Правила имеют такой вид:

$$\begin{aligned} & \text{ЕСЛИ «приращение концепта } K_i \text{ малое»,} \\ & \text{ТО «приращение концепта } K_j \text{ среднее».} \end{aligned} \quad (4)$$

Передача влияния между концептами осуществляется на основе способа нечёткого логического вывода Мамдани [3]. К недостаткам рассмотренных НКК можно отнести следующее. Не учтены совместные воздействия состояний (абсолютного уровня значений) концептов и их приращений. Кроме того, эти модели являются линейными и не учитывают в полной мере динамические свойства реальных систем, которые могут быть существенно нелинейными и нестационарными.

В работе [3] предложены обобщённые нечёткие производственные когнитивные карты. Обобщенной нечёткой производственной когнитивной картой называется нечёткая причинно-следственная сеть (сеть взаимовлияния) вида

$$G = (K, L), \quad (5)$$

где  $K = \{K_1, K_2, \dots, K_p\}$  – множество концептов;

$L = \{l_{ij}\}$  – множество связей между концептами.

Степень влияния концептов определялась посредством парных сравнений оценки влияния одного концепта на другой. При этом использовались лингвистические оценки, приведенные в таблице 3.

Таблица 3 – Шкала относительной силы влияния концептов

Оценка	Определение	Объяснение
0	Нет влияния	Концепт $K_i$ не влияет на концепт $K_j$ ;
0,1	Слабая степень	Опыт и суждения дают незначительную степень влияния концепта $K_i$ на концепт $K_j$ ;
0,3	Умеренная	Опыт и суждения дают легкую степень влияния концепта $K_i$ на концепт $K_j$ ;
0,5	Существенное или сильное влияние	Опыт и суждения дают среднюю силу влияния концепта $K_i$ на концепт $K_j$ ;
0,7	Значительное влияние	Очевидное сильное влияние одного концепта на другой
0,9	Очень сильное влияние	Влияние одного концепта на другой подтверждается очень сильно
0,2; 0,4; 0,6; 0,8	Промежуточное решение	Применяются в компромиссном случае

Результатом применения рассматриваемой методики является построение логико-лингвистической шкалы оценки причинно-следственной важности задачи  $Z_{Q_i}$  объекта  $Q_i$  для других задач этого же объекта.

ЛЛШ оценки причинно-следственной важности обеспечивает количественную и качественную оценку показателей  $N^n(Z_{Q_i})$  для каждой из множества задач, которые может выполнять объект  $Q_i$  исходя из сложившейся обстановки (ситуации). Вариант когнитивной карты, построенной для бригады строителей, показан на рисунке 3.

Показатель  $N^n(Z_{Q_i})$  необходим при разработке алгоритмов формирования рекомендаций на выполнение объектом тех или иных функциональных задач, а также при разработке сценариев и прогнозов предстоящих действий (процессы оценки обстановки, планирования и оперативного управления).

Использование обобщенных нечётких когнитивных карт для построения проблемно-целевых моделей позволяет учесть ситуационные аспекты формируемых когнитивных структур относительно целей, характеризующих пространство возможных состояний систем и оценку мотивационных действий в анализируемых системах.

В рамках предлагаемого способа, причинно-следственная важность, определяющая необходимость выполнения задачи должна обуславливаться складывающимися условиями обстановки и их влиянием на степень достижения цели, стоящей перед системой.

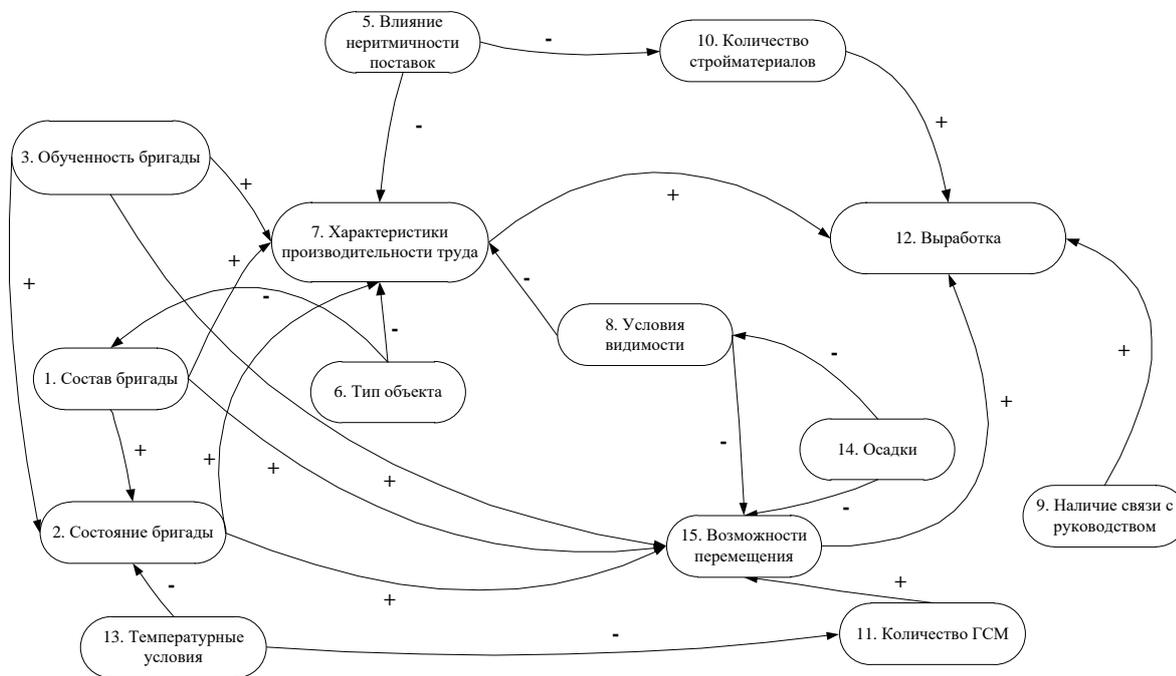


Рисунок 3 – Когнитивная карта, построенная для бригады строителей

В результате оценки текущих значений факторов обстановки формируется множество задач  $\{Z_i\}$ , каждому из которых соответствует показатель, характеризующий необходимость её выполнения  $N(Z_i)$ .

**Оценка важности задач, исходя из их соответствия функциональному предназначению объекта.** Если объекты являются управляемыми объектами, то возможно сравнение их задач по важности. Причем в один и тот же момент времени реальные объекты могут выполнять как одинаковые, так и разные типовые задачи, как из одной, так и из нескольких предметных областей. Анализ логики процессов функционирования различных ОТС показывает, что одновременное выполнение всеми управляемыми объектами своих основных задач невозможно [4, 5]. На практике может иметь место одновременное выполнение основных задач лишь частью управляемых объектов. Причем, логика функционирования объектов ОТС показывает, что выполнение основных задач одними управляемыми объектами в одни моменты времени, определяет возможность выполнения основных задач этих и/или других управляемых объектов в последующие моменты времени.

Каждому объекту ОТС  $Q_i$  ставится в соответствие множество задач  $\{Z_{Qi}\}$ , которые он может выполнять, исходя из своего функционального предназначения [6]. Рассматриваемое соответствие может быть представлено в виде кортежа

$$Q_i = \langle Z_{Qi}, N^b(Z_{Qi}) \rangle, \quad (6)$$

где  $N^b(Z_{Qi})$  – взаимная важность задач  $Z_{Qi}$ .

Взаимная важность задач  $N^b(Z_{Qi})$ , характеризует место рассматриваемой задачи в перечне всех задач  $Z\{Q_i\}$ , которые способен выполнять объект  $Q_i$ , исходя из своего функционального предназначения или другими словами, важность задачи характеризует

степень соответствия наилучшего результата её выполнения функциональному предназначению объекта (цели его функционирования).

Упорядочение задач объектов по важности производится экспертами в области управления этими объектами, значение показателя  $N^b(Z_{Qi})$  определяется экспертным путем. В результате осуществляется построение логико-лингвистической шкалы оценки взаимной важности задач объекта  $Q_i$  (рисунок 4).

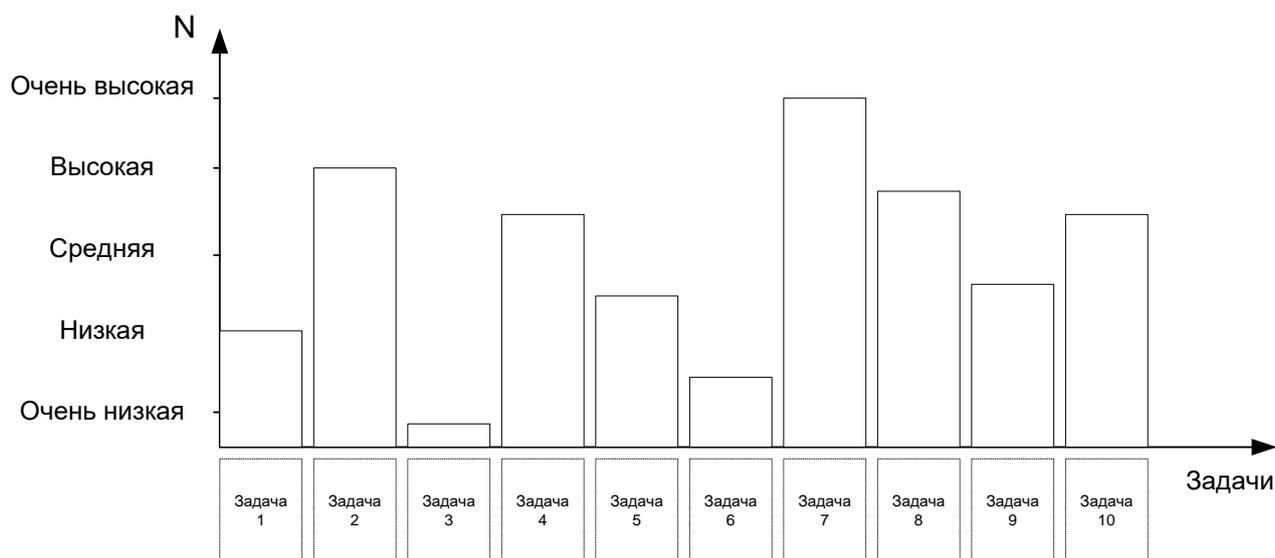


Рисунок 4 – Логико-лингвистическая шкала оценки взаимной важности задач

Среди множества задач, которые может выполнять каждый объект ОТС, выделяются главные задачи, определяющие его функциональное предназначение, а также задачи, влияющие или обеспечивающие (в предыдущие моменты времени) в той или иной мере выполнение основных задач.

В рамках предлагаемого способа, взаимная важность, определяющая необходимость выполнения задачи должна обуславливаться складывающимися условиями обстановки и их влиянием на свойства  $\{w\}$  системы, определяющие способности её по выполнению своего функционального предназначения.

Исходя из этого, процесс оценки необходимости выполнения задачи, должен осуществляться идентификацией текущих значений ситуационных признаков (СПр) [7], а количественные или качественные значения, характеризующие необходимость выполнения этих задач, должны определяться их влиянием на сохранение или улучшение тех или иных свойств системы. Каждому значению  $T_y^l$  некоторого СПр ставится в соответствие множество свойств  $\{w^j\}$  системы и множество задач –  $\{Z^j\}$ . Содержание данных множеств определяется физическим смыслом значения  $T_y^l$ . Каждому свойству  $w$  соответствует его важность  $N(w)$ , характеризующая влияние этого свойства на способности системы по выполнению своего функционального предназначения. Например, в строительной системе можно выделить следующие свойства: производительность; устойчивость и оперативность. Для строительной бригады выделяются следующие функциональные подсистемы: управления; выполнения строительных работ; обеспечения; транспортная (снабжения). Значения СПр отражают состояние различных функциональных подсистем. Таким образом, при наличии в строительной системе 4-х подсистем, количество рассматриваемых свойств системы равно 12. Ввиду того, что одновременное сохранение или улучшение всех свойств практически невозможно, требуется определение взаимной важности 12-ти свойств, характеризующих

готовность рассматриваемой комплексной строительной бригады к достижению стоящей перед ней цели.

Логико-лингвистическая шкала оценки важности свойств системы должна создаваться на этапе проектирования системы поддержки принятия решений (СППР) и корректироваться в процессе её эксплуатации. В основе построения данной ЛЛШ лежит обработка мнений экспертов в области эксплуатации рассматриваемой ОТС. Количественное значение показателя важности  $N(w_c/C)$  свойства  $w_c$ , проявляемые подсистемой  $C$  оценивается по результатам обработки мнений экспертов о важности тех или иных свойств по соответствующей шкале (таблица 5).

Таблица 5 – Шкала для определения важности свойств системы

Качественная оценка важности	Числовая оценка	Классификационное определение важности свойства
Очень высокая	(0,8...1,0)	Свойство оказывает очень сильное влияние на возможности системы по достижению цели
Высокая	(0,6...0,8)	Свойство оказывает сильное влияние на возможности системы по достижению цели
Средняя	(0,4...0,6)	Свойство оказывает умеренное влияние на возможности системы по достижению цели
Низкая	(0,2...0,4)	Свойство оказывает слабое влияние на возможности системы по достижению цели
Очень низкая	(0...0,2)	Свойство не оказывает влияния на возможности системы по достижению цели

ЛЛШ оценки взаимной важности обеспечивает количественную и качественную оценку показателей  $N^b(Z_{Qi})$  для каждой из множества задач, которые может выполнять объект  $Q_i$  исходя из своего функционального предназначения.

В качестве условий оценки важности свойств служит предположение об одновременном проявлении всех свойств системы (или необходимости в одновременном решении всех задач). Задача экспертов состоит в ранжировании важности свойств, исходя из их физического смысла.

Любому значению  $T_{ly}$  каждого фактора обстановки  $u \in Y$  ставится в соответствие множество свойств функциональных подсистем  $\{w_{yc}\}$ , проявляемых ОТС при идентификации значения  $T_{ly}$ . Так как значению каждого фактора обстановки может соответствовать несколько свойств различных подсистем ОТС, то оценка важности идентифицированного значения рассматриваемого фактора производится в виде

$$N(T_{ly}) = \max(N(w_{yc}/C)). \quad (7)$$

В результате идентификации текущих значений  $T_{ly}$  всех факторов обстановки формируется множество оценок  $\{N(T_{ly})\}$ . Кроме того, при построении базы знаний СППР [8], каждому значению  $T_{ly}$  ставится в соответствие множество задач  $\{Z_i\}$ , тогда оценка необходимости выполнения задачи из данного множества определяется в виде

$$N(Z_i) = N(w_{yc}/(C/Z_i)). \quad (8)$$

Таким образом, показатель  $N^b(Z_{Qi})$  необходим при разработке алгоритмов выбора задачи объекта ОТС, которую необходимо спланировать для её выполнения этим объектом (процессы планирования, оперативного управления). Следовательно, оценка важности объектов в текущей ситуации определяется через оценку важности выполняемых ими задач

$$N_i = N(z_i^j). \quad (9)$$

Оценка важности всей ситуации определяется с помощью выражения

$$N(C_i^j) = \max \{N_i\}. \quad (10)$$

Таким образом, оценка ситуативной важности объектов ОТС позволит определить важности её объектов для приоритетного снабжения их различного рода ресурсами.

### Список литературы:

1. Кондратова Н.В., Мослякова А.Л. Анализ существующих подходов к оценке управляющих решений и предложения по их совершенствованию// Международный журнал информационных технологий и энергоэффективности. – 2018. – Т. 3 № 2(8) с. 26-33.
2. Силов В. Б. Принятие стратегических решений в нечеткой обстановке. – М.: ИНПРО–РЕС, 1995. – 228 с.
3. Борисов В. В., Круглов В. В., Федулов А. С. Нечёткие модели и сети. М., Горячая линия – Телеком, 2007. – 284 с.
4. Клир Дж. Системология. Автоматизация решения системных задач. М., Радио и связь, 1990. – 544 с.
5. Романов А. Н., Одинцов Б. Е. Советующие информационные системы в экономике. М., ЮНИТИ-ДАНА, 2000. – 487 с.
6. Балашов О.В. Сравнительная оценка параметров нечётких моделей в задаче аппроксимации функции//Нейрокомпьютеры: разработка, применение. № 8, 2009. С. 49-51.
7. Поспелов Д. А. Ситуационное управление: Теория и практика. М., Наука, 1986. – 288 с.
8. Балашов О. В., Круглов В. В. Подход к извлечению продукционных правил для систем поддержки принятия решений. Математическая морфология. Электронный математический и медико-биологический журнал. Смоленск. <http://www.smolensk.ru/user/sgma/MMORPH/N-12-html/cont.htm>. 2006. Т. 5. – Вып. 4.

### References

1. Kondratova N. V., Moslyakova A. L. Analysis of existing approaches to the assessment of management decisions and suggestions for their improvement // International Journal of Information Technologies and Energy Efficiency. - 2018. - V. 3 No. 2 (8) p. 26-33.
  2. Silov V. B. Making strategic decisions in a fuzzy environment. - M. : INPRO-RES, 1995. – 228 p.
  3. Borisov V. V., Kruglov V. V., Fedulov A. S. Fuzzy models and networks. M., Hotline - Telecom, 2007 . – 284 p.
  4. Clear J. Systemology. Automation of solving system problems. M., Radio and Communications, 1990 . – 544 p.
  5. Romanov A. N., Odintsov B. E. Advisory information systems in the economy. M., UNITY-DANA, 2000 . – 487 p.
  6. Balashov O. V. A comparative evaluation of the parameters of fuzzy models in the problem of approximating a function // Neurocomputers: development, application. No. 8, 2009. S. 49-51.
  7. Pospelov D. A. Situational management: Theory and practice. M., Science, 1986. – 288 p.
  8. Balashov O. V., Kruglov V. V. An approach to extracting production rules for decision support systems. Mathematical morphology. Electronic mathematical and biomedical journal. Smolensk. <http://www.smolensk.ru/user/sgma/MMORPH/N-12-html/cont.htm>. 2006.V. 5. - Vol. 4.
-



Международный журнал информационных технологий и энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 004.6

## ТЕНДЕНЦИИ РАЗВИТИЯ BIG DATA

**Свириденкова М.А., Свириденков К.И.**

*Филиал ФГБУ ВО «НИУ «МЭИ» в г. Смоленске, Смоленск, Россия (214013, г. Смоленск, Энергетический проезд, 1), e-mail: sviridenkova-m-a@mail.ru*

**Ключевой идеей в данном исследовании выступает определение тенденций развития технологий Big Data. Представлена история возникновения Big Data и формирования понятия «большие данные». Выделено пять основных тенденций развития больших данных. Обоснованы положительные стороны использования в организациях технологий Big Data в процессах принятия решений и выработке стратегических направлений развития организации.**

Ключевые слова: большие данные, технологии, информация, принятие решений.

## BIG DATA DEVELOPMENT TRENDS

**Sviridenkova M.A., Sviridenkov K.I.**

*Smolensk Branch of the National Research University "Moscow Power Engineering Institute", Smolensk, Russia (214013, Smolensk, Energeticheskyy proezd, 1), e-mail: sviridenkova-m-a@mail.ru*

**The key idea in this study is the identification of trends in the development of Big Data technologies. The history of the emergence of Big Data and the formation of the concept of "big data" is presented. Five main trends in the development of big data are highlighted. The positive aspects of the use of Big Data technologies in organizations in decision-making processes and the development of strategic directions for the development of organizations are substantiated.**

Keywords: big data, technology, information, decision making.

Сегодня BIG DATA являются ключевым фактором развития информационных технологий. В эпоху информационных технологий и особенно цифровизации всех сфер человеческой деятельности по каждому пользователю Интернета накапливается большое количество информации. Это во многом определяет развитие направления BIG DATA. Термин Big Data нельзя считать синонимом информации или анализа информации. Информация чаще всего может быть представлена в неструктурированном виде, например, это могут быть видеозаписи, машинный код, изображения, текстовые документы и не только. К тому же эти данные могут быть разбросаны по различным хранилищам, которые могут находиться даже на разных континентах. Термин «Большие данные» означает большой объем накопленной информации, а также технологии хранения, вычисления, сервисные услуги.

Значительная часть информации создается не столько человеческими усилиями, сколько с использованием автоматизированных устройств. Причем такие устройства взаимодействуют и друг с другом, и с другими сетями данных, например, сенсоры и интеллектуальные устройства. По прогнозам исследователей, объем данных в мире будет

ежегодно удваиваться. Во много раз вырастет количество виртуальных и физических серверов, в частности, за счет расширения и создания новых data-центров. Поэтому растет потребность в эффективной обработке, анализе больших данных в целях их использования в процессах принятия решений. В связи с этим интересным представляется определение тенденций развития технологий Big Data.

Впервые термин «большие данные» упоминается в 2005 году. Однако до сих пор не существует его устоявшегося определения, которое было бы принято в научном и профессиональном сообществе [1]. Возможно, это связано с гибкостью понятия, за которым находится постоянно изменяющаяся сфера разработки новых средств работы с данными. А может быть причиной этой неопределенности является огромное количество различных областей, в которых «большие данные» могут быть применены: экономика, социология, медицина, ритейл, Интернет вещей, промышленность и многие другие [2].

Феномен «Большие данные» сопровождается тремя основными характеристиками:

- volume – большой объем;
- velocity – большая скорость поступления новых данных;
- variety – разнообразие и неструктурированность.

Область понимания BIG DATA постоянно изменяется, а точнее расширяется. Например, к трем указанным выше характеристикам добавляются другие, тоже начинающиеся на “V” (например, veracity, value). В результате термин обрастает новыми признаками и эволюционирует. Отсюда можно сделать вывод, что спрос на технологию «больших данных» растет [2].

В результате использования технологии Big Data предполагается решение трех типов задач: 1) хранение и управление большими объемами данных в сотни терабайт ; 2) организация неструктурированной информации; 3) анализ Big Data. Под большими объемами данных понимаются данные в сотни терабайт, которые неэффективно обрабатывать с использованием реляционных баз данных. Неструктурированную информацию тоже сложно обрабатывать с использованием таких баз данных и, соответственно, сложно формировать аналитические отчеты и разрабатывать прогностические модели развития исследуемых явлений.

В сфере Big Data существует много направлений. Среди них можно выделить два основных: Big Data engineering, Big Data Analytics (Scientist). Big Data engineering занимается сбором, хранением, преобразованием данных к виду, которого требуют приложения для корректной работы. Big Data Analytics — это следующая часть, в которой возможно использование объемных данных из уже созданных систем. Анализ состоит из расширенных вычислений по данным, где происходит прогнозирование, выявление закономерностей, тенденций и построение классификаций.

Основной движущей силой технологии является постоянно растущий объем производимых и потребляемых данных. Аналитики предсказывают, что к 2025 году этот объем достигнет 163 Збайт (1 Збайт = 1 триллион Гбайт) [3]. Технология «больших данных» стремительно развивается, расширяются и направления их использования. Сегодня спрос на «большие данные» формируется как в государственном, так и в частном секторе. Привлекательным для предприятий является возможность использовать технологию в принятии решений. Поэтому перспективы развития технологии BIG DATA огромные. Они обусловлены увеличением количества данных, поступающих на обработку, а также ужесточением требований к скорости их обработки. Пока невозможно понять, какая страна или корпорация займет в будущем лидирующее положение в этой области. Но уже очевидно, что тот, кто овладеет методами хранения и анализа огромных массивов данных, получит мощнейший инструмент для оптимизации принятия решений и выбора стратегии развития.

Технология все еще находится на раннем этапе своего внедрения, поэтому наблюдать итоги ее реализации в конкретных проектах затруднительно. Сегодня разрыв между «большими данными» и их внедрением в реальных сферах жизни продолжает убывать.

Таким образом, уже можно наблюдать их использование и выделить некоторые тенденции развития.

Первая тенденция развития «больших данных» – разработка аппаратных и программных средств, осуществляющих работу с информацией, распределенной на многих компьютерах. Понятие «большие данные» начинает применяться тогда, когда одного современного компьютера становится недостаточно для обработки и анализа массива информации. То есть, когда речь идет о распределенных системах. Существующие классические СУБД плохо масштабируются в таких случаях, поэтому необходимо создавать новые удобные инструменты. К таким средствам относятся, например, фреймворк Hadoop и СУБД Apache Hive. Но развиваются и новые системы, улучшающие скорость работы с данными – например, отечественное модульное хранилище данных Exarch.

Сегодня основополагающая технология – Hadoop, проект Apache Software Foundation, используемый для реализации поисковых и контекстных механизмов высоконагруженных веб-сайтов [4]. Hadoop свободно распространяет набор утилит, библиотек и программный каркас для разработки и выполнения распределенных программ, работающих на кластерах из множества узлов. Hadoop является основной платформой развития российского рынка, способствующей повышению отечественной конкуренции. В таблице 1 представлены преимущества платформы Hadoop.

Таблица 1 – Преимущества платформы Hadoop

Преимущество	Краткое описание
Снижение времени на обработку данных	Существенное уменьшение времени обработки данных на кластере
Снижение стоимости оборудования	Сокращение затрат на оборудование, требуемое для хранения и обработки данных, в десятки раз.
Повышение отказоустойчивости. Технология позволяет построить отказоустойчивое решение	Выход узла, влияет лишь на производительность системы, но не на ее работу.
Линейная масштабируемость	Наращивание производительности.
Работа с неструктурированными данными	Обработка любых файлов, в том числе неструктурированных. Повышение полезности информации.

Вторая тенденция – разработка инструментов, оперирующих большими массивами данных. Несмотря на наличие некоторого парка инструментов, технология Big Data не так давно перешла с этапа академических исследований к применению в реальности. Уже разработаны и успешно используются инструменты, позволяющие оперировать данными огромных объемов. Однако их всеобщее внедрение еще не наступило. Применение Big Data носит избирательный, хоть и масштабный характер. Например, в Китае работает система «социального кредита», которая определяет рейтинг граждан по их действиям. Эти действия отслеживаются системами массового видеонаблюдения, а поступающая информация обрабатывается с помощью технологий «больших данных». В результате мы наблюдаем еще одну тенденцию: основным ресурсом технологии становится объем обрабатываемых данных в отличие от имевшегося ранее, на этапе академических исследований, - трудовых ресурсов профессионалов. В качестве примера можно привести аналогию с добычей нефти: богаче будет та страна, в которой находятся огромные запасы нефти, а не та, в которой будут сконцентрированы лучшие профессионалы нефтедобычи. При этом предполагается и прогнозируется, что эффективные методы работы с колоссальными массивами информации естественным образом возникнут там, где этой информации наибольшее количество, так как ее необходимо уметь эффективно обрабатывать.

Третья тенденция – централизация хранения данных. На практике могут быть использованы централизованные, децентрализованные и смешанные хранилища данных. В рамках централизованного хранилища справочная информация извлекается из различных источников, систематизируется, дополняется, записывается в эталонное хранилище. На текущий момент времени такая организация хранения данных получила наибольшее распространения, в основном, из-за простоты и высокой скорости доступа к анализируемой информации. Принцип централизованного хранения реализуют системы, указанные в таблице 2.

Таблица 2 – Продукты, использующие принцип централизованного хранения данных

Продукты	Применение
1. IBM Client Information Integration Solution (IBM CIIS)	Хранилище данных, использующее пакетный режим и обработку данных в режиме реального времени. Управление осуществляется через графическую оболочку, что сопровождается снижением требований к квалификации сотрудников, обеспечивающих поддержку нормативно-справочной информации. На сегодняшний день IBM CIIS используют преимущественно банки и страховые организации.
2. Oracle Customer Data Hub (Oracle CDH)	Продукт может использоваться для управления реестрами клиентов, сотрудников, населения как отдельных регионов, так и страны в целом. На сегодняшний момент времени продукт используют в телекоммуникационных и высокотехнологичных компаний.
3. SAP Master Data Management (SAP MDM)	Данный продукт служит вспомогательным инструментом для управления коллаборативным и гибким бизнесом. Используемая технология – MDM (Управление основными мастер-данными). В его основе лежит обеспечение тесных связей с деловыми партнерами. При этом предоставляется возможность централизованного ведения справочников и классификаторов, поддерживаются процессы изменения информации в справочниках, обеспечивается распределенный доступ пользователей к этим данным, а также распространение согласованных данных между всеми приложениями предприятия. Разработчик продукта немецкий ИТ-гигант SAP AG

При использовании децентрализованного хранилища справочной информации создается виртуальная база данных, в случае обращения к которой идет запрос данных к тем системам, где они хранятся непосредственно. В результате информация по одному клиенту будет собрана из фрагментов, содержащихся в разнородных базах данных, но в виртуальной базе данных она будет представлять собой одну запись. Продукты, реализующие децентрализованное хранение данных, MetaBase и MetaMatrix Server, разработчик - компания MetaMatrix.

Находят применение и смешанные решения, использующие одновременно принципы централизованного и децентрализованного хранения данных. Они представлены ниже, в таблице 3.

Таблица 3 – Продукты, использующие принцип децентрализованного хранения данных

Продукты	Применение
1. Initiate Systems Identity Hub	Разработка компании Initiate Systems. Ранее было использовано в централизованной архитектуре.. Особенно выделяется система сверки данных. На сегодняшний день продукт широко используется в фармацевтических компаниях и организациях здравоохранения, а также благодаря динамичному развитию стремится занять и другие ниши рынка управления нормативно-справочной информацией.
2. DWL Customer	Разработка компании DWL. Представляет собой гибридное хранилище данных, которое обладает лучшей среди представленных в статье систем поддержки сервисно-ориентированной архитектуры (Service-Oriented Architecture, SOA). Продукт внедрен в финансовых организациях, но предполагает внедрение решения для сферы телекоммуникаций и здравоохранения;
3. Siebel Universal Customer View (Siebel UCM)	Аналогично DWL Customer, обладает развитой поддержкой архитектуры SOA; в составе продукта насчитывается около 140 Web-сервисов.
4. Siperian Master Reference Manager (Siperian MRM)	Продукт представляет собой систему с гибкой моделью хранения, включающую: 1) эталонную базу данных, по которой можно идентифицировать сущности и ключи соответствия между используемыми источниками информации; 2) другие источники, включающие дополнительные (например, операционные) характеристики сущности. Характеристики из других источников загружаются в эталонную базу данных.

Таким образом, среди централизованных, децентрализованных и смешанных хранилищ данных наибольшее распространение получили первые из перечисленных. К ним можно отнести облачные хранилища и дата-центры. С целью практического использования больших объемов собранной информации предварительно ее необходимо систематизировать. Тенденция к централизации и дисциплинированному управлению данными, несомненно, очень важна при принятии решений.

Качество обработки информации целесообразно оценивать путем каких-либо показателей. Как известно, показатели могут носить индивидуальный, групповой, комплексный характер (или интегральный). На сегодняшний день компания IDC в рамках исследований «Глобальная цифровизация от периферии к центру» («The Digitization of the World – from Edge to Core») разработала один из таких показателей. Это индекс DATCON (DATA readliness CONdition) - «уровень готовности к работе с данными», который позволяет руководителям определить, на каком уровне в организации находятся управление данными, их использование и монетизация [5]. Это интегральный показатель. Его значение находится в диапазоне 1 – 5 (1 – критичный, 5 – оптимизированный). При расчете индекса учитываются следующие факторы: темпы изменения объемов данных, ценность данных, уровень информационной безопасности организации, размеры инвестиций, зрелость и качество управления, наличие у персонала организации необходимых навыков по обработке больших массивов данных, степень вовлеченности топ-менеджеров организации в проекты, связанные с данными. В рамках исследований условий работы с постоянно увеличивающимся объемом

данных, проведенных компанией IDC в четырех областях (производство, финансовый сектор, здравоохранение, СМИ и индустрия развлечений), лидирует производство. В целом индекс DATCON может быть использован в процессах принятия решений в различных областях и особенно в производственной деятельности организаций. Предполагается, что это позволит выработать наиболее эффективное стратегическое решение.

Четвертая тенденция, связанная с получением данных – это развитие IoT (интернет вещей) технологий, которые обеспечивают автоматизированный сбор данных на самых различных типах устройств. На примере автомобилестроения можно видеть, что данная технология уже успешно используется. Данные в реальном времени, поступающие от автомобилей, позволяют оказывать потребителям более грамотный сервис: начиная от исправления поломки до поиска угнанного транспортного средства. Все больше компаний внедряют IoT-технологии в свои продукты, что приводит к необходимости овладения методами сбора больших массивов информации. В целом это выгодно для производителя, так как позволяет быстрее и полнее получать информацию о работе своих продуктов для их дальнейшего развития. Рынок интернета вещей продолжает расти, и сейчас сложно найти сферу человеческой жизнедеятельности, где он бы не использовался (рисунок 1).

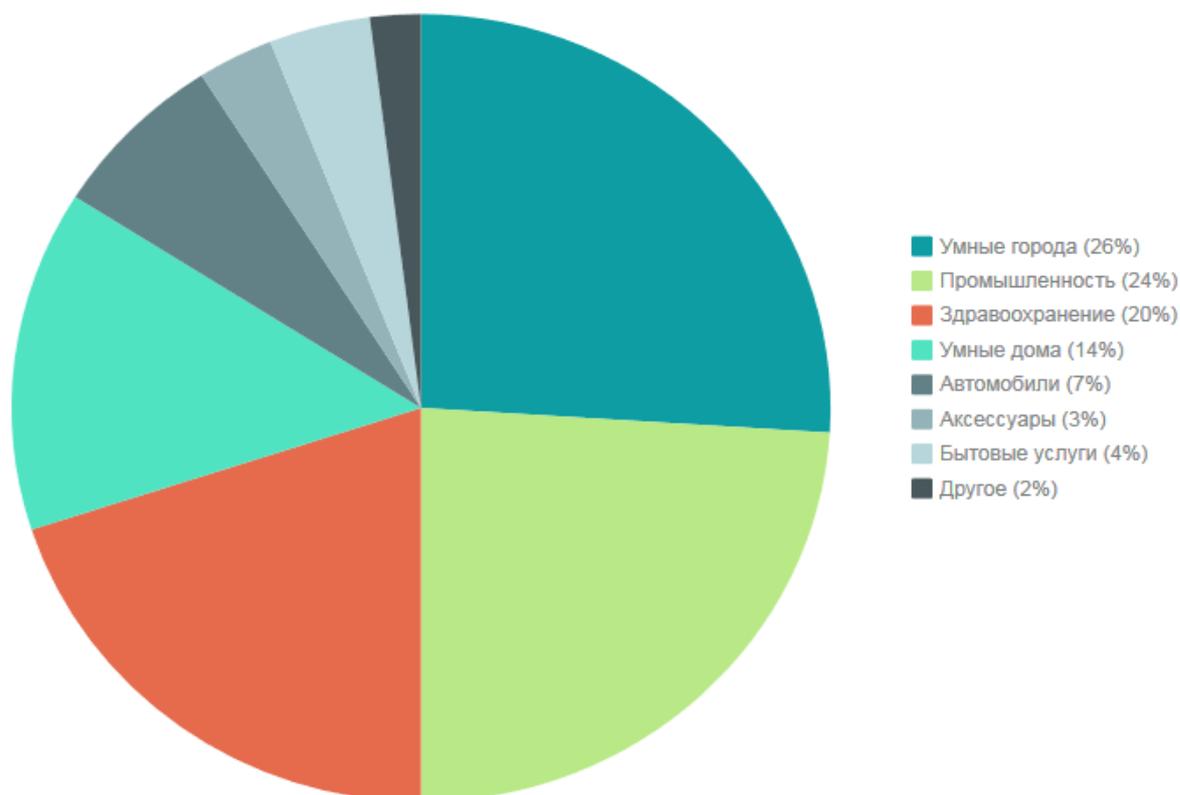


Рисунок 1 – Использование IoT в различных сферах в мире

Пятая тенденция связана с анализом полученных данных. В этой области все большую роль играет искусственный интеллект [6]. Современные алгоритмы машинного обучения на данный момент способны анализировать данные лучше человека во многих областях. Например, сейчас искусственный интеллект по образцам крови может лучше предсказать заболевание пациента, чем самый опытный врач-гематолог. Тот факт, что машина допускает меньше ошибок и имеет лучшую способность к предсказанию, позволяет предположить, что постепенно человеческий ресурс во многих областях будет заменен компьютерным. И это на самом деле является очередной тенденцией развития работы с данными. Искусственный интеллект активно используется многими технологическими компаниями нашего времени именно потому, что может обработать огромный массив информации и выдать адекватный результат. Например, виртуальный голосовой помощник «Алиса», созданный компанией

«Яндекс», хранит огромную базу информации, помогающую отвечать на запросы пользователей. Компьютер можно обучить работать над специализированной задачей быстрее и лучше, чем человека. Возможно, в будущем человек будет принимать только самые важные решения, доверяя какие-то рутинные задачи роботам и алгоритмам.

В России развитие и использование в практической деятельности технологий Big Data определяются требованиями цифровизации экономики, реализуемыми программой «Цифровая экономика Российской Федерации». Указом Президента РФ от 9 мая 2017 г. № 203 утверждена государственная программа «О Стратегии развития информационного общества в Российской Федерации на 2017 - 2030 годы», в рамках которой планируется развивать технологии Big Data [7].

Таким образом, в данной работе рассмотрены некоторые аспекты рынка больших данных, выявлены положительные стороны и тенденции развития данного направления, а также обоснована целесообразность использования Big Data на российском рынке.

### Список литературы:

1. Большие данные // Википедия, 2017. [Электронный ресурс]. Режим доступа: <http://ru.wikipedia.org/?oldid=87934960/>
2. Васильев В.И. Обзор технологий для работы с Big Data // Молодой ученый. – 2020. - №9. – с. 13 – 14. -Режим доступа : <https://moluch.ru/archive/299/67818/>
3. Резванов А. К 2025 году общий объем данных в мире достигнет 163 зеттабайт. [Электронный ресурс]. Режим доступа: <http://www.macster.ru/news/170412-k-2025-godu-obshchiy-obem-dannykh-v-mi>.
4. Аналитический обзор рынка Big Data. [Электронный ресурс] – Режим доступа: <https://www.sostav.ru/publication/big-data-razmer-ne-imeet-znachenie-31028.html>
5. The Digitization of the World From Edge to Core. David Reinsel – John Gantz – John Rydning
6. Митрович С. Рынок «больших данных» и их инструментов: тенденции и перспективы в России // МИР (Модернизация. Инновации. Развитие). 2018. Т. 9. № 1. С. 74–85. DOI: 10.18184/2079-4665.2018.9.1.74–85
7. Указ Президента Российской Федерации от 09.05.2017 № 203 "О Стратегии развития информационного общества в Российской Федерации на 2017 - 2030 годы" [Электронный ресурс] – Режим доступа: <http://publication.pravo.gov.ru/Document/View/0001201705100002>

### References

1. Big data // Wikipedia, 2017. [Electronic resource]. Access Mode: <http://ru.wikipedia.org/?oldid=87934960/>
  2. Vasiliev V.I. Overview of technologies for working with Big Data // Young scientist. - 2020. - No. 9. - with. 13 - 14. -Access mode: <https://moluch.ru/archive/299/67818/>
  3. Rezvanov A. By 2025, the total amount of data in the world will reach 163 zettabytes. [Electronic resource]. Access mode: <http://www.macster.ru/news/170412-k-2025-godu-obshchiy-obem-dannykh-v-mi>.
  4. Analytical review of the Big Data market. [Electronic resource] - Access mode: <https://www.sostav.ru/publication/big-data-razmer-ne-imeet-znachenie-31028.html>
  5. The Digitization of the World From Edge to Core. David Reinsel - John Gantz - John Rydning
  6. Mitrovich S. Market of "big data" and their tools: trends and prospects in Russia // MIR (Modernization. Innovations. Development). 2018. Vol. 9. No. 1. P. 74–85. DOI: 10.18184 / 2079-4665.2018.9.1.74–85
  7. Decree of the President of the Russian Federation of 05.09.2017 No. 203 "On the Strategy for the Development of the Information Society in the Russian Federation for 2017 - 2030" [Electronic resource] - Access mode: <http://publication.pravo.gov.ru/Document/View / 0001201705100002>.
-



ОТКРЫТАЯ НАУКА  
издательство

Международный журнал информационных технологий и энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 004.77

## АНАЛИЗ ИСПОЛЬЗОВАНИЯ ПРОФИЛИРОВЩИКОВ ПРОГРАММНОГО ПРОДУКТА ДЛЯ ПАРАЛЛЕЛЬНОГО КОДА В СИСТЕМЕ ВЫЧИСЛИТЕЛЬНОГО КЛАСТЕРА

<sup>1</sup>Федулов Я.А., <sup>2</sup>Федулова А.С.

Филиал ФГБУ ВО «НИУ «МЭИ» в г. Смоленске, Смоленск, Россия (214013, г. Смоленск, Энергетический проезд, 1), e-mail: <sup>1</sup>fedulov\_yar@mail.ru, <sup>2</sup>voitsitskay@mail.ru

В работе представлен общий анализ существующих профилировщиков программ, позволяющих отследить время выполнения программного кода. Отражены простые программы с функциями для проверки выполнения профилирования. Использование каждого из способов вывода времени затраченного на выполнение отдельной функции, позволяет сделать вывод о необходимости использования стандартных или специализированных продуктов профилирования программ для оптимизации участков кода, что существенно позволит облегчить работу программисту.

Ключевые слова: профилирование, профилировщик программного продукта, статистические профайлеры, компилятор, кластер, параллельное программирование, динамический анализ.

## THE ANALYZE OF A USAGE OF PROFILERS FOR THE PARALLEL CODE IN A COMPUTER CLUSTER SYSTEM

<sup>1</sup>Fedulov Ya.A., <sup>2</sup>Fedulova A.S.

Smolensk Branch of the National Research University "Moscow Power Engineering Institute", Smolensk, Russia (214013, Smolensk, Energetichesky proezd, 1), e-mail: <sup>1</sup>fedulov\_yar@mail.ru, <sup>2</sup>voitsitskay@mail.ru

The paper presents a general analysis of existing program profilers, which track the execution time of program code. Simple programs show with functions for checking profiling. The use is demanded for each method of outputting the time spent on performing a separate function, it allows us to conclude that it is necessary to use standard or specialized products for profiling programs to optimize sections of code, which will greatly facilitate the work of the programmer.

Keywords: profiling, software product profiler, statistical profilers, compiler, cluster, parallel programming, dynamic analysis.

В настоящее время используется большое количество многопроцессорных систем для расчетов больших данных и для повышения производительности вычислительных систем [1]. Разработчикам программы важно понять особенность выполнения кода, чтобы выполнять его эффективно и задействовать большинство доступных вычислительных ресурсов. Это особенно актуально для параллельных программ, где требуется решение основных вопросов, таких как: загрузка, балансировка, синхронизация и связь узловых потоков. Многопроцессорные вычислительные системы, использующие в своем составе несколько вычислительных узлов с 2-мя или более процессорами, далее называемые вычислительными кластерами.

При использовании вычислительных кластеров требуется более высокий уровень программирования для использования всех возможностей используемой системы. Одним из важных критериев программного кода является эффективность программного кода, необходимо отслеживать результат распараллеливания программы, чтобы выполнение осуществлялось быстрее и использовалось больше ресурсов предоставляемого оборудования. Производительность параллельной программы зависит от модели, для которой код разработан. Ряд инструментов различной сложности и мощности были разработаны для параллельных языков программирования и систем [2]. Инструменты для сбора данных о производительности сохраняют полученную информацию в виде трасс событий или файле. Трассировка дает более подробный анализ временных характеристик, но является наиболее трудоемкой. Профилирование обладает важным преимуществом по сравнению с трассировкой, так как выдает время, затраченное при вызове каждой из функций программного кода.

Целью исследования является изучение известных методов профилирования параллельных программ и экспериментальное подтверждение преимущества стандартных средств профилирования для удаленного исследования параллельного кода.

Материал и методы исследования: проведение экспериментов на гибридном вычислительном кластере ФГБОУ ВО НИУ МЭИ в г. Смоленске (ГВКС) с использованием технологии распараллеливания передачи информации, которая позволяет обмениваться сообщениями между процессами в параллельных программах для получения данных от установленных профилировщиков.

Результатами исследования будут поставленные эксперименты и отчеты о профилировании параллельных программ.

#### Описание экспериментов

Для получения оптимального времени выполнения программы, необходимо учитывать работу программного кода, который обладает рядом ограничений, которые необходимо учесть при дальнейшей работе.

Профилирование – это процесс осуществления сбора данных о работе программы, об эффективности выполнения кода, выделение мест, в которых программа выполняется дольше всего, а также выводы о дальнейшем усовершенствовании программного кода на определенных его участках для дальнейшей оптимизации. За анализ работы программы отвечает профилировщик (профайлер), а для получения данных реального времени используется динамический анализ выполнения программы. Динамический анализ программного кода представляет собой анализ программы при ее выполнении. К основным этапам динамического анализа относят следующие: подготовка данных, тестовый запуск программного кода и сбор определенных параметров, анализ полученных данных.

Характеристики параллельной программы – это выделение функций программы, для определения времени выполнения вычислений и обменных операция. Профилировщики сохраняют последовательность событий и временные метки в специальных файлах gmon.out.

К самым известным современным профилировщикам относятся: стандартный профилировщик Linux – Gprof [3], Intel VTune Profiler [4], DTrace [5], Intel Threading Analysis Tools [6], Intel Trace Analyzer and Collector 7.1 [7], Pallas GmbH [8], EDPEPPS [9].

Выбор профилировщиков, из представленных выше (используемых для определения участков программного кода, где тратится много времени, и которые должны быть преобразованы), зависит в первую очередь от установленного программного обеспечения, а также от установленных программ. Профилирование может проводиться для любых сфер деятельности, где необходимо ускорение работы программного кода за счет распараллеливания и точное определение мест для распараллеливания.

Выделяются наборы базовых показателей, которые учитываются с помощью профилировщиков [10].

1. Общее время выполнения каждого участка программы.

2. Удельное время выполнения каждого участка программы.
3. Выявление причин конфликтов и штрафа.
4. Получение количества вызовов определенного участка программы.
5. Расчет степени покрытия программы.

Общее время рассчитывается для выполнения каждой точки программы, и также отслеживаются времязатратные участки. При этой операции получения времени профилировщик отразит, что 99 % общего времени выполнения программа работает в функции main(). Для упрощения работы и понимания программистов профайлеры выводят время, за которое осуществлялось выполнение дочерних функций.

Удельное время выполнения команд (растактовка). Профилировщик VTune позволяет рассчитать данный показатель.

В статье представлен анализ установленных профилировщиков на гибридном вычислительном кластере ФГБОУ ВО НИУ МЭИ в г. Смоленске (ГВКС), работа профайлеров проверена с помощью написанных программ для параллельных технологий MPI и OpenMP.

Как правило, большинство профилировщиков используются только для вычисления общего времени, но большинство из них могут выполнять и другие функции для удобства понимания времязатратных мест кода, но когда нет необходимости использовать сложные в понимании и установке, поэтому достаточны стандартные ресурсы. Для этого необходимо знать основные особенности выбранных профайлеров, представленные далее. На ГВКС можно использовать большое количество профайлеров, но из-за сложности установки на кластере был выбран стандартный gprof, strace и Intel VTune Profiler для технологии MPI, так как является более сложной и интересной для рассмотрения.

Рассмотрим основные характеристики технологии MPI (Message Passing Interface). MPI – это библиотека функций при работе с параллельными процессами с помощью передачи сообщений. Данная технология на сегодняшний день является самым востребованным стандартом обмена сообщениями для параллельных программ. Множество реализаций MPI разработано для компьютерных платформ. Применяется при написании программного кода для кластеров. Стандарт MPI передачи сообщений важен для корректной работы приложения на платформе, а также на системе пользователя. Данная технология используется в различных языках Fortran, Java, C, C++. MPI ориентирован на работу с распределенной памятью, а OpenMP – на работу с общей памятью.

Для оптимизации времени выполнения программы необходимо выделить участки кода, которые чаще всего используются, выделение времени выполнения функций. Из этого можно сделать вывод, что необходимо точно знать места кода, где больше всего затрачивается время в программе при реальных входных данных, для эффективной ее оптимизации. Для этого будут рассмотрены различные инструменты работы с программным кодом и выделены достоинства и недостатки использования каждого из них. В статье представлены наиболее известные способы получения данных о затраченных ресурсах, а также задействование стандартных ключей компиляторов для других компоновщиков с несвойственными применениями, позволяющих получить требуемые результаты.

## 1. Gprof

Gprof (GNU-профайлер) – стандартная программа профилирования для компилятора набора GCC(GNU Compiler Collection) GNU(GNU is Not Unix) проекта по разработке свободного программного обеспечения (СПО) для различных аппаратных и программных платформ, а также для высокопроизводительных систем (кластеров), а также осуществляется поддержка большого количества языков C, C++, Objective-C, Fortran, Ada, Go, D, и библиотек данных языков (libstdc ++).

Gprof-профилировщик разработан для компилятора gcc и имеет различные ключи для компилятора такие как -pg, -g, -b, каждый из которых является необходимым для получения

требуемой информации [3]. Главная особенность рассмотренного способа заключается в использовании стандартной конструкции компилятора GCC для компилятора другой технологии MPI (компилятор mpicc), информации о работе данного компилятора большое множество, но об использовании стандартных профилировщиков немного информации [11, 12]. Был сделан вывод, что если mpicc является частью набора GCC, то и ключи компилятора gcc будут выполняться и на mpicc.

Тестирование MPI-профилировщика было проведено на программном коде параллельной программы, с помощью которой осуществляется вычисление времени выполнения обменных операций и вывода адресов переменных, используемых в программе, для этого были применены функций MPI. Программа написана на языке высокого уровня C.

Для проверки работы были использованы опции для профилирования программы `-pg, -g` при компилировании (получении исполняемого файла):

```
mpicc -pg main.c -o main.ex -g,
```

где опция `-pg` для создания исполняемого файла с настройками для генерации профилирующих данных. В результате будет создан исполняемый файл, который необходимо запустить с помощью функции `mpirun`

```
mpirun -n 32 main.ex 33 4,
```

где `-n` – число потоков, на которых выполняется программный код и будет получен файл отчета `gmon.out`, `33` и `4` – числа, которые меняются друг с другом большое количество раз. Профайлер подсчитывает время участков кода, в которых задерживалась программа, совмещает данные с графом вызовов и результат заносит в отчет (файл `gmon.out`). После выполнения

```
gprof main.ex gmon.out>main_L.txt
```

`main_L.txt` – файл, в котором записано время выполнения всех функций программы на рисунке 1.

```
Flat profile:
Each sample counts as 0.01 seconds.
% cumulative self          self   total
time  seconds seconds    calls ms/call ms/call name
100.29    0.04    0.04         1   40.12   40.12  obmen
...
Call graph (explanation follows)
granularity: each sample hit covers 2 byte(s) for 24.93% of
0.04 seconds

index % time    self children  called    name
[1]   100.0    0.04    0.00     1/1      main [2]
-----
[1]   100.0    0.04    0.00     1       obmen [1]
-----
<spontaneous>
[2]   100.0    0.00    0.04          main [2]
[2]   100.0    0.04    0.00     1/1      obmen [1]
-----
...
```

Рисунок 1 – Результирующий файл с временными замерами

Из результирующего файла видно, что функция `obmen()` вызвана из функции `main()` 1 раз и занимает «не все время» выполнения программы.

При добавлении `-l` будет получен файл с более подробной информацией компановки, представленный на рисунке 2.

```
gprof main.ex gmon.out>main_L.txt -l

Flat profile:
Each sample counts as 0.01 seconds.
% cumulative self      self      total
time  seconds seconds  calls  Ts/call  Ts/call  name
75.22   0.03   0.03
25.07   0.04   0.01
 0.00   0.04   0.00      1    0.00    0.00  obmen (main.c:181 @ 401555)
...
Call graph (explanation follows)
granularity: each sample hit covers 2 byte(s) for 24.93% of 0.04 seconds

index % time    self children  called    name
[3]    0.0    0.00  0.00    1/1      main (main.c:130 @ 4012ec) [58]
[3]    0.0    0.00  0.00    1        obmen (main.c:181 @ 401555) [3]
Index by function name

[3] obmen (main.c:181 @ 401555) [2] obmen (main.c:190 @ 4015cf) [1] obmen
(main.c:184 @ 40163b)
```

Рисунок 2 – Результирующий файл с подробными временными замерами

После выполнения был получен результат работы функций с использованием стандартных опций для иного вида технологий компилятора, что удобно и не требует дополнительных временных и денежных затрат.

Недостатком является неточность выдаваемых данных, что может быть важным для определенных задач.

Далее рассмотрим еще один стандартный профилировщик, он также будет использоваться для параллельной программы описанной ранее.

## 2. Strace

В вычислительной системе и написанных параллельных программах могут возникать ошибки или работа программы некорректна, то можно использовать стандартные средства системы Linux, позволяющие следить за работой каждой программы, системными вызовами, которые используются программой [14].

Команда `strace` позволят получать все системные вызовы программы, отправленные во время выполнения, а также время каждого системного вызова. Команда является более универсальной для Linux систем и позволяет получить общее время выполнения программного кода, но при этом получаются более громоздкие результаты, что не всегда удобно для восприятия.

Далее представлен краткий отчет, полученный при выполнении команды `strace`.

Для получения результирующего файла были выполнены следующие шаги компилирование программы с помощью стандартного компилятора MPI:

```
mpicc main.c -o main.exe.
```

Запуск программы с помощью команды `strace` со специальными опциями `-ttt` (время выполнения системных вызовов в формате UNIX) и `-T` (время выполнения вызова):

```
strace -ttt -T -o strace.txt mpirun -n 32 main.exe 12 15.
```

Все подробно описано, так как мало информации о совместном использовании команды `strace` и компилятора `mpicc` (14) на рисунке 3 представлен отчет с временными выводами.

```

1585743740.490193  execve("/usr/lib64/openmpi/bin/mpirun",  ["mpirun",  "-n",  "32",
"main.exe", "12", "15"], [/* 52 vars */) = 0 <0.000464>
1585743740.491041  brk(NULL) = 0x22cc000 <0.000017>
1585743740.491141  mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7effa3c5b000 <0.000018>
.....
1585743757.531581  close(11) = 0 <0.000007>
1585743757.531602  rmdir("/tmp/openmpi-sessions-voitsitskay_as@mng1_0") = 0 <0.000030>
1585743757.531647  openat(AT_FDCWD, "/tmp/openmpi-sessions-voitsitskay_as@mng1_0/14331",
O_RDONLY|O_NONBLOCK|O_DIRECTORY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
<0.000027>
1585743757.531714  munmap(0x7effa15a5000, 2127344) = 0 <0.000018>
1585743757.531760  munmap(0x7effa1bb6000, 2101648) = 0 <0.000013>
1585743757.531811  munmap(0x7effa17ad000, 2110072) = 0 <0.000015>
1585743757.532211  exit_group(0) = ?

```

Рисунок 1 – Результирующий файл с подробными временными выводами

Команда `strace` с опцией `-c` позволяет получить отчет о системных вызовах и определить самые вызываемые функции на рисунке 4.

% time	seconds	usecs/call	calls	errors	syscall		
-----	-----	-----	-----	-----	-----	30.73	0.039963
17	2287	304	openat				
12.77	0.016609	7	2353	32	read		
11.04	0.014360	15	974	959	newfstatat		
9.02	0.011723	5	2462		close		
5.81	0.007553	229	33		clone		
5.62	0.007305	22	326	139	open		
3.24	0.004214	13	329		write		
3.22	0.004187	6	663		getdents		
2.19	0.002845	38	75	1	wait4		
2.08	0.002699	4	746		fcntl		
2.00	0.002604	7	381	63	stat		
2.00	0.002599	31	84		poll		
1.05	0.001369	4	351		fstat		
0.78	0.001016	9	114		munmap		
0.77	0.001004	8	130		pipe		
0.77	0.001002	5	209		mmap		
0.75	0.000981	30	33		sched_setaffinity		
.....							
0.00	0.000000	0	2		lseek		
0.00	0.000000	0	1		dup		
0.00	0.000000	0	1		execve		
0.00	0.000000	0	1		getpgrp		
-----	-----	-----	-----	-----	-----		
100.00	0.130035		12540	1551	total		

Рисунок 4 – Отчет о системных вызовах

Команда `strace` позволяет получить большое количество системных данных и является стандартной для Linux, но описание использования ее только для последовательных программ затрудняет работу и понимание, поэтому проведенные эксперименты на реальном вычислительном кластере позволили сделать выводы о возможности ее использования также для параллельных программ.

Основным недостатком является сложность получения информации для работы с параллельными программами. Дальнейшая работа с данной командой и технологиями распараллеливания позволит получить необходимые данные о совместной работе технологий.

Существуют специализированные пакеты для профилирования параллельных программ к ним относятся Intel VTune Profiler, Intel Parallel Studio XE, Intel System Studio [15], они являются платными и сложны в установке.

### 3. Intel VTune Profiler

Intel VTune Profiler – специализированный профилировщик, используемый для повышения производительности программы на высокотехнологичном оборудовании. Для получения эффективных улучшений необходимо обладать корректными данными для повышения производительности. Intel VTune Profiler отражает основные данные профилирования в понятном для пользователя интерфейсе, который позволяет легче проводить анализ и интерпретацию.

Профилировщик может быть использован для различных направлений и позволит оптимизировать программное обеспечение. К основным можно отнести высокопроизводительные вычисления прогнозирования погоды и биоинформатике, приложения для транспортировки и производства, медиа-программы для транскодирования видео и обработки изображений, облачные приложения, драйверы устройств, игровые движки.

Профайлер используется для Linux. Используется Intel VTune Profiler для профилирования последовательных и многопоточных приложений, которые выполняются на различных аппаратных платформах (CPU, GPU, FPGA), графический интерфейс позволяет отследить работу программы и проанализировать полученные результаты, рисунок 5 [4].

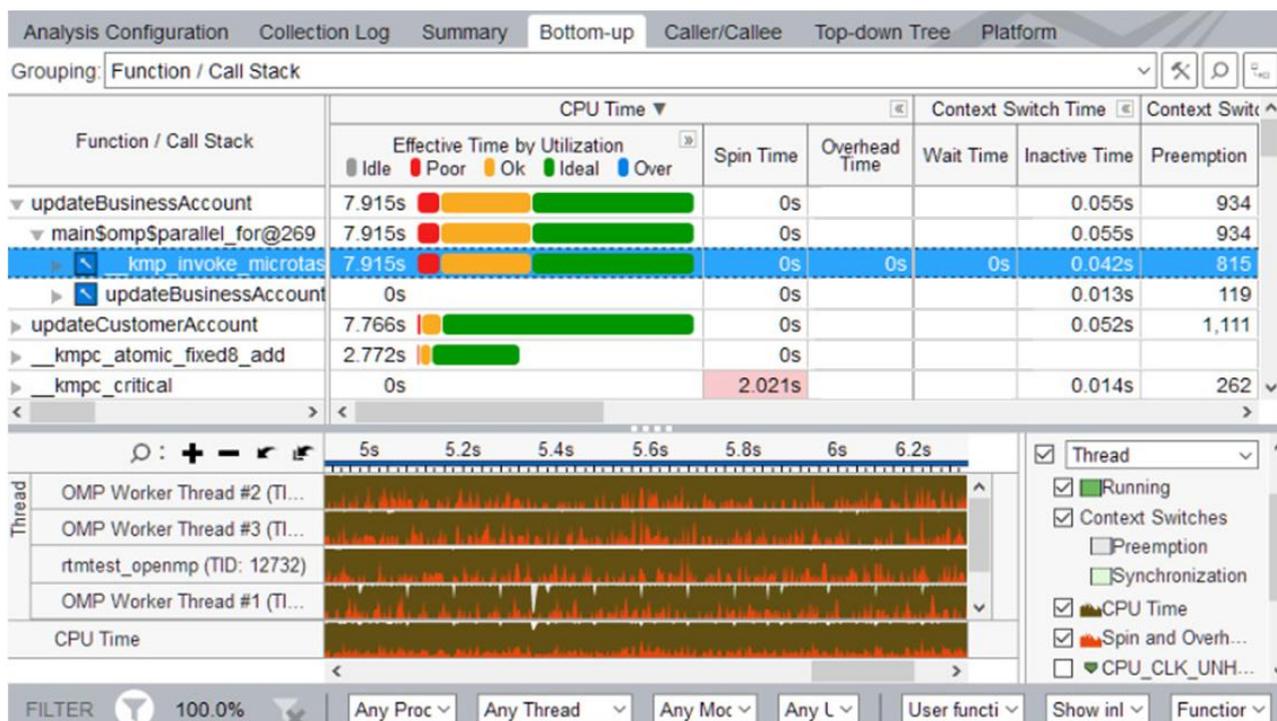


Рисунок 5 – Вывод результирующего файла при наличии графического интерфейса на вычислительном кластере

Данная программа удобна, если она закуплена и есть специалист, умеющий работать с данным обеспечением. Недостатком можно считать, что профайлер имеет графический интерфейс исключительно на машине вычислительного кластера, при удаленном подключении через PuTTY (клиент для различных протоколов удалённого доступа, таких как SSH, Telnet, rlogin) нет возможности видеть все графики, работать сложно и неудобно.

Поэтому для профилирования программ удобен и прост в использовании стандартный профилировщик компилятора Linux, который выдает результаты в файл, который можно

прочитать, разобрать и понять, что очень важно для удаленной работы с многопроцессорной вычислительной системой.

#### Заключение

В статье представлены результаты экспериментального исследования стандартных средств профилирования, а также их соединение с компиляторами, которые только теоретически могли использоваться совместно, но на практике были проведены необходимые тесты и получены требуемые результаты, были проверены на практике и работа стандартных средств профилирования для Linux, были отражены в отчетах о выполнении профилирования для анализа производительности параллельных программ.

Описаны сильные и слабые стороны исследованных стандартных и специализированных профилировщиков, сделаны выводы по наиболее приемлемому использованию вышеописанных инструментов анализа производительности высокотехнологичных систем в зависимости от технических и экономических возможностей, а также возможности удаленного контроля для исследования работы программ.

#### Список литературы:

1. Чайковский, Д.С.. Анализ современных средств профилирования параллельных программ / Д.С. Чайковский, Н.А. Гулевич // Информационные технологии (Вестник СГТУ) – Саратов, 2014. – 109-113.
2. Matthias S. Müller. OpenMP Shared Memory Parallel Programming: International Workshops / Matthias S. Müller, Barbara M. Chapman // IWOMP 2005 and IWOMP – USA. 2006.
3. GPROF Tutorial [Сайт]. URL: <https://www.thegeekstuff.com/2012/08/gprof-tutorial/>, 27.03.2020
4. Intel VTune Profiler [Сайт]. URL: <https://software.intel.com/en-us/vtune>, 28.03.2020
5. Кляус С.М. Инструменты динамической трассировки DTrace и SystemTap, 2017, 220 с. URL: <http://www.unilib.neva.ru/dl/local/407/oe/oe.ppt> (дата обращения: 01.04.2020)
6. Intel® Threading Building Blocks [Сайт]. URL: <https://software.intel.com/en-us/articles/intel-threading-building-blocks-threading-performance-and-correctness-analysis>, 28.03.2020
7. Getting Started with Intel® Trace Analyzer and Collector on Linux\* OS [Сайт]. URL: <https://software.intel.com/en-us/get-started-with-itac-for-linux>, 28.03.2020
8. Компании-разработчики программных средств [Сайт]. URL: [https://parallel.ru/tech/software\\_vendors.html](https://parallel.ru/tech/software_vendors.html), 28.03.2020
9. Средства создания и проектирования параллельных программ [Сайт]. URL: [https://parallel.ru/tech/tech\\_dev/build\\_par.html#edpepps](https://parallel.ru/tech/tech_dev/build_par.html#edpepps), 29.03.2020
10. Касперски К. Техника оптимизации программ. Эффективное использование памяти. - СПб.: БХВ-Петербург, 2003. - 464 с.
11. Значимые результаты от gprof для кода MPI [Сайт]. URL: <https://stackoverflow.com/questions/53794093/how-do-i-get-meaningful-results-from-gprof-on-an-mpi-code>, 29.03.2020
12. Евсеев, И.. Использование MPI на компьютерах ЦСТ [Сайт]. URL: [https://www.opennet.ru/docs/RUS/MPI\\_intro/using.html](https://www.opennet.ru/docs/RUS/MPI_intro/using.html), 29.03.2020
13. Команда Strace в Linux [Сайт]. URL: <https://losst.ru/komanda-strace-v-linux>, 29.03.2020
14. Using Strace to Examine Your IO [Сайт]. URL: <https://www.clustermonkey.net/FileSystems/using-strace-to-examine-your-io/Page-2.html>, 29.03.2020
15. Choose the Best Option [Сайт]. URL: <https://software.intel.com/en-us/vtune/choose-download>, 01.04.2020

#### References

1. Tchaikovsky, D.S. Analysis of profiling parallel programs modern means / D.S. Tchaikovsky, N.A. Gulevich // Information Technologies (Bulletin of SSTU) - Saratov, 2014. - 109-113.

2. Matthias S. Müller. OpenMP Shared Memory Parallel Programming: International Workshops / Matthias S. Müller, Barbara M. Chapman // IWOMP 2005 and IWOMP - USA. 2006.
  3. GPROF Tutorial [Site]. URL: <https://www.thegeekstuff.com/2012/08/gprof-tutorial/>, 03/27/2020
  4. Intel VTune Profiler [Site]. URL: <https://software.intel.com/en-us/vtune>, 03/28/2020
  5. Klyaus S.M. Dynamic Tracing Tools DTrace and SystemTap, 2017, 220 p. URL: [https://www.tune-it.ru/documents/10136/828092/dtrace\\_stap\\_book\\_b10a.pdf/66cae909-5677-40d2-a306-ca265392d142](https://www.tune-it.ru/documents/10136/828092/dtrace_stap_book_b10a.pdf/66cae909-5677-40d2-a306-ca265392d142), 04.01.2020
  6. Intel® Threading Building Blocks [Site]. URL: <https://software.intel.com/en-us/articles/intel-threading-building-blocks-threading-performance-and-correctness-analysis>, 03/28/2020
  7. Getting Started with Intel® Trace Analyzer and Collector on Linux \* OS [Site]. URL: <https://software.intel.com/en-us/get-started-with-itac-for-linux>, 03/28/2020
  8. Software companies [Site]. URL: [https://parallel.ru/tech/software\\_vendors.html](https://parallel.ru/tech/software_vendors.html), 03/28/2020
  9. Tools for creating and designing parallel programs [Electronic resource]. URL: [https://parallel.ru/tech/tech\\_dev/build\\_par.html#edpepps](https://parallel.ru/tech/tech_dev/build_par.html#edpepps), 03/29/2020
  10. Kaspersky K. Technique for optimizing programs. Efficient use of memory. - SPb .: BHV-Petersburg, 2003 .- 464 p.
  11. Significant gprof results for MPI code [Site]. URL: <https://stackoverflow.com/questions/53794093/how-do-i-get-meaningful-results-from-gprof-on-an-mpi-code>, 03/29/2020
  12. Evseev И. Using MPI on computers TsST [Site]. URL: [https://www.opennet.ru/docs/RUS/MPI\\_intro/using.html](https://www.opennet.ru/docs/RUS/MPI_intro/using.html), 03/29/2020
  13. Strace command in Linux [Site]. URL: <https://losst.ru/komanda-strace-v-linux>, 03/29/2020
  14. Using Strace to Examine Your IO [Site]. URL: <https://www.clustermonkey.net/FileSystems/using-strace-to-examine-your-io/Page-2.html>, 03/29/2020
  15. Choose the Best Option [Site]. URL: <https://software.intel.com/en-us/vtune/choose-download>, 04/01/2020.
-



Международный журнал информационных технологий и энергоэффективности

Сайт журнала: <http://www.openaccessscience.ru/index.php/ijcse/>



УДК 620.9

## МОДЕЛЬ ОЦЕНКИ ЭФФЕКТИВНОСТИ ДЕЯТЕЛЬНОСТИ ЭНЕРГЕТИЧЕСКИХ ОРГАНИЗАЦИЙ

**Свириденкова М.А., Свириденков К.И.**

Филиал ФГБУ ВО «НИУ «МЭИ» в г. Смоленске, Смоленск, Россия (214013, г. Смоленск, Энергетический проезд, 1), e-mail: [sviridenkova-m-a@mail.ru](mailto:sviridenkova-m-a@mail.ru)

**В настоящей статье приведены основные результаты исследования возможности оценки эффективности деятельности энергетических организаций. Разработана модель оценки эффективности деятельности энергетических субъектов.**

Ключевые слова: энергетические организации, эффективность деятельности, оценка эффективности деятельности.

## PERFORMANCE ASSESSMENT MODEL FOR ENERGY ORGANIZATIONS

**Sviridenkova M.A., Sviridenkov K.I.**

Smolensk Branch of the National Research University "Moscow Power Engineering Institute", Smolensk, Russia (214013, Smolensk, Energeticheskyy proezd, 1), e-mail: [sviridenkova-m-a@mail.ru](mailto:sviridenkova-m-a@mail.ru)

**This paper presents the main results of the study of the possibility of evaluating the effectiveness of energy organizations. A model for evaluating the performance of energy entities has been developed.**

Keywords: energy organizations, performance, performance evaluation.

Электроэнергия является важнейшим ресурсом для промышленности, транспорта, сельского хозяйства. Невозможно представить ни одну сферу деятельности человека, без ее использования. Электроэнергетика охватывает все сферы жизни человека. Она достаточно сильно влияет на конкурентоспособность и темпы роста экономики любой страны, в том числе и РФ, значительную долю которой составляют энергоёмкие отрасли. Генерацией, распределением электроэнергии необходимо управлять. Базой для эффективного управления является анализ и оценка результатов деятельности энергетических организаций. Поэтому интересным представляется формирование модели оценки эффективности деятельности энергетических организаций.

Россия в 2019 году по итогам производства электроэнергии находится на четвертом месте в мире (рис. 1). Первые три места занимают Китай, США и Индия [1,2]. В 2019 году Россия произвела электроэнергии приблизительно в 5,5 раз меньше, чем Китай, и в 4 раза меньше, чем США, достигнув уровня производства электроэнергии 1990 года. При сравнении объемов производства электроэнергии разных стран, конечно, следовало бы учесть количество населения государств, а также объемы производимой продукции.

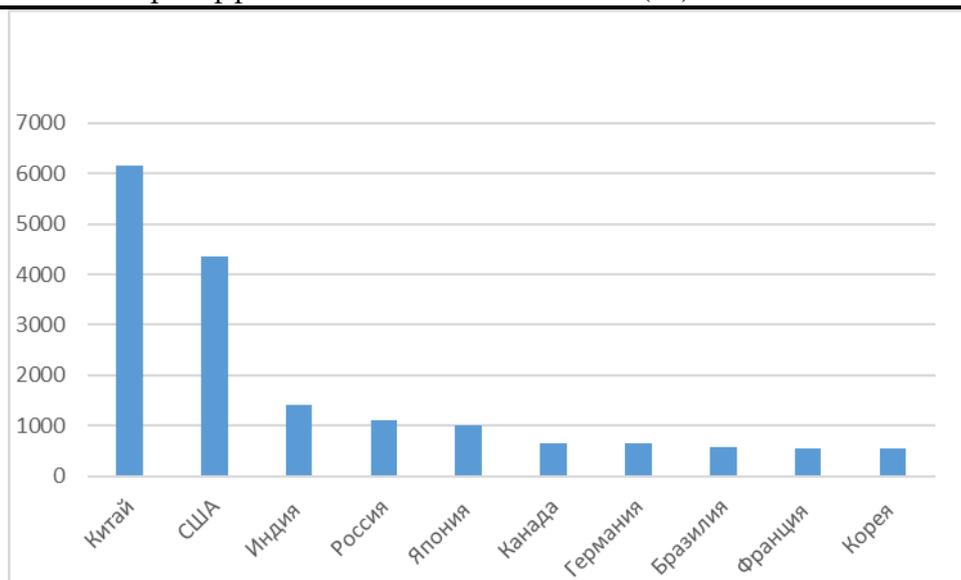


Рисунок 1 – Производство электроэнергии по странам мира в 2019 г., млрд. кВт\*ч

Динамика изменения производства электроэнергии в России представлена на рис. 2 [3]. С 1970 по 1990 гг. наблюдался устойчивый рост производства электроэнергии. В последующие годы, а именно с 1991 по 1994 гг., наблюдалось резкое падение производства электроэнергии, которое продолжалось до 1998 г. С 1998 г. опять начался устойчивый рост производства электроэнергии со среднегодовым темпом прироста 2,6% в год. В 2009 году намечился незначительный спад производства. С 2016 года на протяжении трех лет наблюдается устойчивый среднегодовой темп прироста производства в размере 10%, который изменился лишь в 2019 г. и составил 5%.

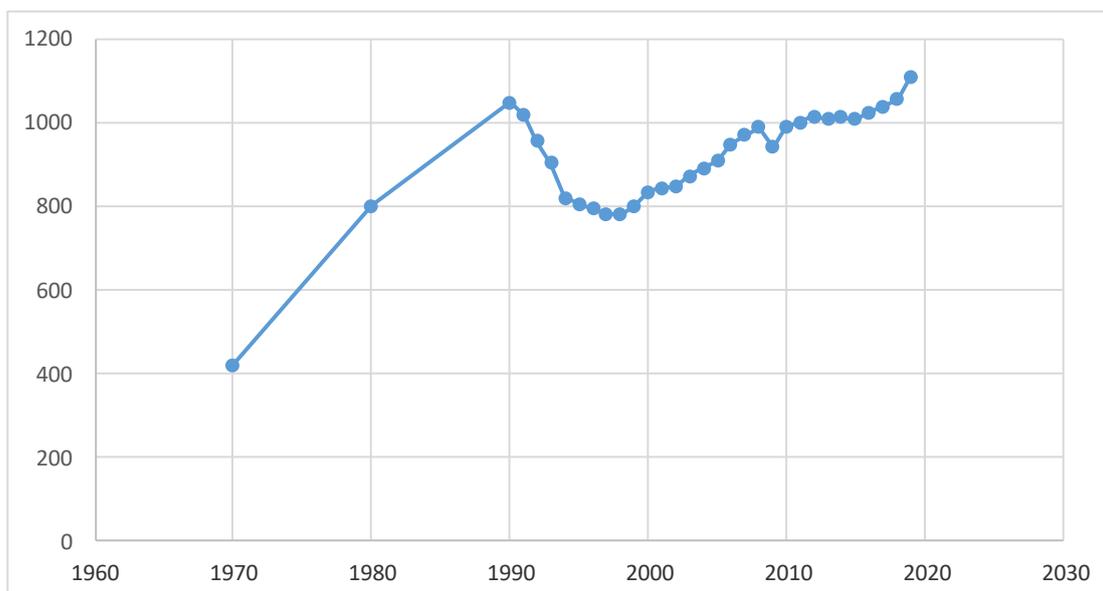


Рисунок 2 – Динамика производства электроэнергии в России, млрд. кВт\*ч

Так как энергетический сектор - самый главный сектор хозяйства страны, то развитие других секторов обусловлено эффективной деятельностью энергетического сектора, а именно энергетических предприятий. Актуальным является поиск резервов повышения эффективности деятельности организаций энергетики.

Текущее состояние энергетики характеризуется следующим [4]:

1. устаревшее оборудование и технологии, требующие срочной модернизации или замены из-за снижающихся характеристик эффективности и надежности;
2. высокий уровень цен для конечного потребителя, который обуславливает высокие цены на продукцию и делает ее мало конкурентоспособной на мировом рынке.

Для снижения (или устранения) вышеуказанных факторов Приказом ФСТ России от 30.03.2012 № 228-э были утверждены Методические указания по регулированию тарифов с применением метода доходности инвестированного капитала для использования в электроэнергетике [5]. Начался плановый переход к регулированию тарифов на услуги по передаче электрической энергии с использованием метода доходности инвестированного капитала (RAB или RAB-регулирование). По словам В.В. Путина, тарифная политика должна обеспечивать инвестиционную привлекательность инфраструктуры отрасли, стимулировать повышение надежности и качества услуг, энергосбережение и применение наиболее эффективных технологических решений (В.В. Путин. Выступление на заседании Правительства Российской Федерации 17 декабря 2009 г.). В целом использование RAB-регулирования способствует повышению экономической эффективности деятельности субъектов электроэнергетики, привлечению частных инвестиций на модернизацию и развитие инженерной инфраструктуры.

Перспективными направлениями развития энергетической отрасли являются:

1. Использование интеллектуальных технологий, в частности, развитие «умной энергетики» – Smart Grid. Умная сеть – это программный комплекс, который позволяет эффективно распределить энергию между потребителями, обеспечив стабильность работы энергосети [6].
2. Замена старого оборудования на новые образцы (в большей степени за счет привлечения инвестиций).
3. Замена старого оборудования на инновационные образцы, имеющие современные характеристики и позволяющие снизить вредные воздействия на окружающую среду, сократить затраты в части использования энергетических ресурсов, повысить эффективность производства и передачи электроэнергии.
4. Использование возобновляемых источников энергии (солнечной энергии, энергии ветра, биомассы, геотермальной энергии, водной энергии и др.).

Каждое из этих направлений при умелой организации и качественной реализации приведет к повышению эффективности электроэнергетической отрасли. Однако основой отрасли являются сами энергетические организации. Именно они являются производителями, и на них лежит ответственность за производство и доставку электроэнергии до потребителей. Поэтому следует уделять особое внимание электроэнергетическим субъектам. Эффективность внедрения и использования вышеуказанных направлений необходимо оценивать и анализировать. Использование для этой цели только данных бухгалтерской и финансовой отчетности недостаточно. Эти данные отражают только сторону самой энергетической организации. Однако эти организации предлагают и поставляют свои услуги потребителям, а поэтому в некоторой степени зависят от них.

Модель оценки эффективности деятельности энергетических организаций целесообразно представить в виде комплексного показателя, включающего интегральные показатели по различным направлениям, отражающим внешнюю и внутреннюю сторону функционирования исследуемых объектов. Энергетические организации – это социально значимые объекты, поэтому помимо чисто отраслевой характеристики – энергетической, а

также свойственной для любого предприятия – экономической, необходимо исследовать социальные и экологические характеристики. В результате выделяем направления интегральных показателей:

1. Оценка энергетической эффективности.
2. Оценка экономической эффективности.
3. Оценка социальной эффективности
4. Оценка экологической эффективности.

Каждое направление должно включать показатели, по которым наиболее эффективно можно дать оценку деятельности конкретного экономического субъекта. В рамках исследований, проведенных в работе [7], было выделено более 80 показателей деятельности энергетического предприятия. Далее для выделения наиболее значимых показателей по направлениям используем экспертный метод. Результаты отбора показателей по направлениям энергетической, экономической, экологической, социальной эффективности и надежности энергоснабжения потребителей представлены в таблицах 1,2,3,4.

Таблица 1 - Показатели оценки энергетической эффективности

Наименование показателя	Расчет показателя	Комментарий
<b>1. Оценка энергетической эффективности</b>		
Энергоемкость, кВт.ч	$\frac{\text{Энергия, потребляемая системой}}{\text{Результат функционирования данной системы}}$	Характеризует расход энергии на единицу продукции или национального дохода
Удельные расходы топлива на производство электрической энергии, Уд.э	$\frac{\text{Полный расход топлива на ед. отпущенной электроэнергии}}{\text{Выработанная энергия за отчетный период}}$	Характеризует эффективность комбинированного потребления (производства) тепловой и электрической энергии
Коэффициент полезного действия (КПД) брутто, %	$\frac{\text{Полезно выработанная энергия}}{\text{Вся затраченная энергия}}$	Характеризует степень технического совершенства
Коэффициент полезного действия (КПД) нетто, %	$\frac{\text{Полезно отпущенная энергия}}{\text{Вся затраченная энергия}}$	Характеризует коммерческую экономичность

Энергетическая эффективность представляет собой результативность производственной деятельности организации, которая определяется путем сравнения стоимости выпущенной продукции (то есть результатов) и затрат (то есть энергетических ресурсов) на достижение этих результатов [8]. Оценка энергетической эффективности и целесообразности, как правило, основывается на показателях энергоиспользования - коэффициентах полезного действия (КПД), энергоемкости, а также на удельных расходах энергии, которые позволяют выявить энергетические потери и, самое главное, возможность определить конкретные пути энергосбережения. Повышение энергетической эффективности означает следующее: 1) получить больше продукции при одинаковых энергетических

затратах; 2) получить тот же объем продукции при меньших энергозатратах; 3) получить больше продукции при меньших энергетических затратах.

Таблица 2 - Показатели оценки экономической эффективности

Наименование показателя	Расчет показателя	Комментарий
<b>2. Оценка экономической эффективности</b>		
Рентабельность от продаж, дол. или %	$\frac{\text{Чистая прибыль}}{\text{Выручка}}$	Показывает насколько эффективно в целом ведется деятельность организации. В данном расчете рентабельность означает долю прибыли в выручке.
Трудоемкость, руб./руб.	$\frac{\text{Затраты труда}}{\text{Общий объем произведенной продукции (работ, услуг)}}$	Величина, обратная выработке, характеризует затраты труда на производство единицы продукции (работ, услуг)
Фондоемкость, руб./руб.	$\frac{\text{Среднегодовая стоимость основных производственных фондов}}{\text{Годовой объем реализованной продукции (работ услуг)}}$	Отражает стоимость основных производственных фондов в расчете на 1 руб. реализованной продукции (работ, услуг)
Капиталоемкость, руб./руб.	$\frac{\text{Стоимость основных средств}}{\text{Объем продукции (работ услуг), выпускаемой за год}}$	Отражает основной капитал, который необходим для выпуска продукции (работ, услуг) стоимость в 1 руб.
Материалоемкость, руб./руб.	$\frac{\text{Сумма материальных затрат, включаемых в себестоимость продукции (работ, услуг)}}{\text{Объем материальных затрат}}$	Характеризует суммарный расход всех материальных ресурсов на производство единицы продукции (работ, услуг)

Экономическая эффективность - это результативность деятельности организации, определяемая сравнением полезных конечных результатов ее функционирования (например, выручки, прибыли) к затраченным ресурсам (затратам). Оценка экономической эффективности характеризуется способностью реализовать максимальный объем продукции (работ, услуг) допустимого качества с наименьшими затратами. При измерении экономической эффективности деятельности электроэнергетических организаций используют такие показатели как рентабельность (рентабельность от продаж, рентабельность активов, рентабельность затрат), трудоемкость, фондоемкость, материалоемкость, капиталоемкость и другие. С их помощью сопоставляются разнообразные вариации развития

производства и решаются структурные проблемы. В отличие от показателей других направлений деятельности энергетического предприятия, показатели, используемые в оценке экономической эффективности, рассчитываются по данным бухгалтерской отчетности.

Таблица 3 - Показатели оценки социальной эффективности

3. Оценка социальной эффективности		
3.1. Оценка степени надежности энергоснабжения потребителей		
Частота отказов генерирующего оборудования, 1/год	<u>Число отказавшего оборудования</u> Общее число оборудования	Показывает количество отказавшего оборудования в единицу времени (в данном случае за календарный год).
Длительность перерывов электропитания потребителей, ч	Сумма времени, необходимого для ликвидации причины перерыва электроснабжения, и времени, необходимого на восстановление электроснабжения для рассматриваемой технической установки	Перерывы электроснабжения потребителей нарушают нормальную жизнь городов и промышленных предприятий, приводят к большим народнохозяйственным убыткам, а поэтому нежелательны.
Величина недоотпуска энергетической продукции потребителям, кВт.ч	Математическое ожидание недоотпуска энергии потребителям за конкретный промежуток времени	Величина недоотпуска энергетической продукции потребителям. Различают два вида недоотпуска: 1) при эксплуатации, когда возникает отказ оборудования, приводящий к снижению производительности; 2) при прогнозируемых отказах, когда есть вероятность снижения производительности.
Коэффициент резерва генерирующих мощностей.	$K_{рез} = (N_{max.дост} - P_{max}) / P_{max}$ , где $N_{max.дост}$ – максимально доступная мощность энергосистемы; $P_{max}$ – максимальная нагрузка энергосистемы	Под резервом генерирующей мощности понимают генерирующую мощность, которая может быть реализована за определенный период времени.
Коэффициент накопленного износа основных средств	<u>Сумма начисленной амортизации</u> Первоначальная стоимость основных средств	Исходные данные для расчета - бухгалтерская отчетность. Показатель условный, так как зависит от способа начисления амортизации
Коэффициент обновления основных средств	<u>Стоимость новых основных средств</u> Стоимость основных средств на конец года	Исходные данные для расчета - бухгалтерская отчетность. Повышение коэффициента обновления основных средств означает увеличение в общем

		объеме парка машин и оборудования новых, более эффективных.
Коэффициент выбытия основных средств	$\frac{\text{Стоимость выбывших основных средств}}{\text{Стоимость основных средств на начало года}}$	Исходные данные для расчета – бухгалтерская отчетность. Чем больше коэффициент выбытия основных фондов, тем меньше сроки службы элементов основных фондов и наоборот.
3.2. Оценка социальной эффективности управления персоналом		
Текущность персонала, %	$\frac{\text{Количество уволенных сотрудников}}{\text{Среднеписочная численность}} * 100\%$	Характеризует движение человеческих ресурсов из штата в штат предприятия
Коэффициент создания новых рабочих мест в регионе, %	$\frac{\text{Созданные рабочие места в регионе}}{\text{Число замещенных рабочих мест}} * 100\%$	Отражает степень увеличение рабочих мест и сокращение безработицы в данном регионе
Убытки от аварий и отказов по вине персонала, тыс.руб.	Σ(полный ущерб от аварий, прямые потери организации, затраты на ликвидацию и расследование аварий, затраты, понесенные вследствие гибели и травматизма людей, косвенный ущерб)	Отражает стоимость убытков от аварий, отказов оборудования, связанных с прямой виной обслуживающего персонала
Охрана труда и обеспечение безопасности, руб./руб.	$\frac{\text{Затраты на охрану труда и обеспечение безопасности}}{\text{Выручка}}$	Характеризует систему мероприятий по обеспечению безопасности жизни и здоровья работников в процессе трудовой деятельности

Наряду с показателями оценки экономической эффективности для анализируемых объектов рассматривают и социальную эффективность, так как показатели экономической эффективности выражают лишь отдельные характеристики результативности деятельности электроэнергетических организаций. Рассматривают социальную эффективность управления персоналом организации и социальную эффективность удовлетворения конечных потребностей общества. Успех любой организации во многом зависит от коллектива с его профессионализмом, самоотдачей, потенциалом. Понятие социальной эффективности управления персоналом подразумевает гармоничное развитие каждого сотрудника как личности, а именно совершенствование его профессионализма, стимулирование социальной активности, формирование благоприятного психологического климата. Оценка социальной эффективности ориентирована на результативность удовлетворения социальных потребностей сотрудников и достижение за счет этого целей организации. Оценка социальной эффективности управления персоналом проявляется в возможности достижения позитивных изменений в организации, таких как сокращение текучести кадров, создание новых рабочих мест, создание мероприятий по предотвращению аварий по вине персонала, разработка мероприятий по охране труда, его безопасности и другие.

Оценка социальной эффективности удовлетворения конечных потребностей общества определяется несколькими факторами: 1) надежное и бесперебойное электрической и тепловой энергии; 2) реализация инвестиционных программ по замене производственных мощностей; 3) использование передовых технологий, направленных на обеспечение энергосбережения и высоких экологических показателей; 4) выстраивание эффективной системы управления предприятием. Степень надежности энергоснабжения достаточно сильно зависит от установленного на энергетическом предприятии оборудования. Поэтому следует учитывать частоту отказов генерирующего оборудования, длительность перерывов электропитания потребителей, величину недоотпуска энергетической продукции потребителям, коэффициент резерва генерирующих мощностей, а также показатели, характеризующие основные средства (коэффициент накопленного износа основных средств, коэффициент обновления основных средств, коэффициент выбытия основных средств).

Таблица 4 - Показатели оценки экологической эффективности

4. Оценка экологической эффективности		
Экологические платежи, тыс. руб.	Расчет экологических платежей производится предприятием в зависимости от объема загрязняющих веществ, попадающих в окружающую среду.	Характеризует суммарные затраты на экологические платежи
Доля экологических платежей в себестоимости продукции, дол. или %	<u>Экологические платежи</u> Себестоимость продукции	Характеризует удельный вес объема экологических платежей в себестоимости продукции
Расходы, связанные с повышением экологичности производства, тыс. руб.	Суммарные затраты экологической направленности	Характеризуют годовой объем расходов на повышение экологичности производства

С целью компенсации вреда, наносимого окружающей среде во время осуществления хозяйственной деятельности организациями, государством установлены экологические платежи. Экологические платежи берутся за следующие виды негативного воздействия на окружающую среду: выброс разнообразных загрязняющих веществ в атмосферу, сброс в подземные и поверхностные водоемы загрязняющих веществ, размещение отходов. Величина экологических платежей равна сумме платежей за загрязнение. Поэтому оценка деятельности энергетического субъекта по экологическим показателям очень важна.

В целом каждое из четырех направлений представляет собой интегральный показатель, рассчитывать который предлагается с помощью формулы средней арифметической взвешенной, присвоив предварительно на основе экспертного метода весовые коэффициенты каждому из предложенных в направлении показателей. Сумма весовых коэффициентов по каждому направлению, должна быть равна 1. Производить оценку интегральных показателей предлагается ежеквартально, по полугодиям и ежегодно. Результаты следует анализировать в динамике.

Таким образом, в ходе проведенных исследований установлено, что с целью повышения эффективности деятельности энергетических субъектов, привлечения

инвестиций в эту отрасль целесообразным является использование предложенной модели оценки. Она представляет собой комплексный показатель, включающий группу интегральных показателей по следующим направлениям: энергетическому, экономическому, социальному, экологическому. Предложенный подход к оценке эффективности деятельности энергетических субъектов позволяет сформировать информационную базу, которую можно использовать для текущего и перспективного анализа, а также инвестиционной привлекательности организации.

### Список литературы:

1. Обзор «Электроэнергетика в России. Итоги 2018 года и прогноз развития». [Электронный ресурс] - Режим доступа: <https://aftershock.news/?q=node/766436&full>
2. Топ 10 стран по производству электроэнергии. [Электронный ресурс] - Режим доступа: <https://zen.yandex.ru/media/zavodfoto/top-10-stran-po-proizvodstvu-elektroenergii-5c651ec36acc00ac2deac4>.
3. Энергетика России. [Электронный ресурс] - Режим доступа - [https://ru.wikipedia.org/wiki/Энергетика\\_России](https://ru.wikipedia.org/wiki/Энергетика_России)
4. Егоров А.В. Исследование проблем увеличения эффективности выполнения работ энергопредприятия // Молодой ученый. – 2018. - №42. – С.198 – 200. [Электронный ресурс] - Режим доступа: <https://moluch.ru/archive/228/53191/> :
5. Методические указания по регулированию тарифов с применением метода доходности инвестированного капитала // Приказ ФСТ России от 30.03.2012 № 228-э. [Электронный ресурс] - Режим доступа : [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_128373/a9d44de9042709b257773da27099203b53bdc641/](http://www.consultant.ru/document/cons_doc_LAW_128373/a9d44de9042709b257773da27099203b53bdc641/)
6. Гаврилович Е.В., Данилов Д.И., Шевченко Д.Ю. «Умные сети» Smart Grid – перспективное будущее энергетической отрасли России // Молодой ученый. – 2016. - №28.2. – С. 55-59. [Электронный ресурс] - Режим доступа: <https://moluch.ru/archive/132/36972/>
7. Мансуров Р.Е. Экспертный метод при формировании набора показателей конкурентоспособности энергетических предприятий// Административно-управленческий портал. [Электронный ресурс] - Режим доступа: <http://www.aup.ru/articles/marketing/42.htm>
8. Федеральный закон "Об энергосбережении и о повышении энергетической эффективности и о внесении изменений в отдельные законодательные акты Российской Федерации" от 23.11.2009 N 261-ФЗ. [Электронный ресурс] - Режим доступа: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_93978/](http://www.consultant.ru/document/cons_doc_LAW_93978/)

### References

1. Overview "Electricity in Russia. Results of 2018 and development forecast." [Electronic resource] - Access mode: <https://aftershock.news/?q=node/766436&full>
2. Top 10 countries for electricity production. [Electronic resource] - Access mode: <https://zen.yandex.ru/media/zavodfoto/top-10-stran-po-proizvodstvu-elektroenergii-5c651ec36acc00ac2deac4>.
3. Energy of Russia. [Electronic resource] - Access mode - [https://ru.wikipedia.org/wiki/Energy\\_Russia](https://ru.wikipedia.org/wiki/Energy_Russia)
4. Egorov A.V. The study of the problems of increasing the efficiency of the energy enterprise // Young scientist. - 2018. - No. 42. - S.198 - 200. [Electronic resource] - Access mode: <https://moluch.ru/archive/228/53191/> :
5. Guidelines for tariff regulation using the method of return on invested capital // Order of the Federal Tariff Service of Russia dated 03.30.2012 No. 228-e. [Electronic resource] - Access mode: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_128373/a9d44de9042709b257773da27099203b53bdc641/](http://www.consultant.ru/document/cons_doc_LAW_128373/a9d44de9042709b257773da27099203b53bdc641/)

6. Gavrilovich E.V., Danilov D.I., Shevchenko D.Yu. "Smart grids" Smart Grid - a promising future for the energy industry in Russia // Young scientist. - 2016. - No. 28.2. - pp. 55-59. [Electronic resource] - Access mode: <https://moluch.ru/archive/132/36972/>
  7. Mansurov R.E. The expert method in the formation of a set of indicators of competitiveness of energy enterprises // Administrative and administrative portal. [Electronic resource] - Access mode: <http://www.aup.ru/articles/marketing/42.htm>
  8. Federal law "On energy conservation and on improving energy efficiency and on amendments to certain legislative acts of the Russian Federation" dated 11.23.2009 N 261-ФЗ. [Electronic resource] - Access mode: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_93978/](http://www.consultant.ru/document/cons_doc_LAW_93978/).
-