

АВТОМАТИЗАЦИЯ ОПРЕДЕЛЕНИЯ КРИТИЧЕСКОГО ПУТИ В ЛОГИСТИЧЕСКОЙ СИСТЕМЕ

В современных условиях продолжительность жизненного цикла продукции становится все короче, а скорость изменений в производственных и логистических технологиях все выше, что предъявляет к логистическим системам материальных грузопотоков дополнительные требования. Они должны быть более гибкими в отношении выполняемых задач, быстро и с малыми затратами времени и ресурсов адаптироваться к постоянно меняющимся запросам в системе. С целью повышения эффективности и качества функционирования логистической системы в работе предлагается производить анализ и синтез сети со сложной топологией методами компьютерного моделирования с применением в качестве средства решения ориентированного графа, по которому определяется критический путь как наиболее эффективный для достижения заданного критерия качества. В результате разработаны алгоритм и процедура оптимизации, позволяющие сократить время, необходимое для перемещения грузопотока от исходного в конечный пункт, согласно критерию качества. Алгоритм реализован с помощью разработанной программы в кодах MatLAB, простота и технологическая прозрачность которой не требуют дополнительного лицензирования. Рассмотрен конкретный пример определения критического пути для подтверждения корректности предложенных технических решений.

Ключевые слова: автоматизация, алгоритм, логистика, ориентированный граф, критический путь, целевой функционал, модель, критерий качества.

В АЖНЕЙШЕЙ задачей, стоящей сегодня перед логистическими компаниями и перед транспортной системой России в целом, является быстрая адаптация к новым, более высоким требованиям рынка к объему и, главное, качеству предоставляемых услуг. Поскольку продолжительность жизненного цикла продукции становится все короче, а скорость изменений в производственных и логистических технологиях все выше, системы материальных грузопотоков должны быть более гибкими в отношении выполняемых ими задач и вносимых новшеств. Они должны быстро и с малыми затратами адаптироваться к постоянно меняющимся запросам рынка. Всем этим требованиям соответствуют автоматические системы, использующие транспорт, работающий без участия человека [1].

Основой таких систем являются различные транспортные средства, такие, например, как робокары, «сердцем» которых являются устройства навигации и связи. Компьютер более высокого уровня обеспечивает диспозицию машин путем оптимизации маршрута движения и временных затрат на выполнение операций [2]. Эта тенденция стимулируется многочисленными инновациями в автоматических транспортных системах [3]. Поэтому повышение эффективности таких логистических систем является важной управленческой и инновационной задачей [4], [5].

Основной моделью большинства логистических схем доставки грузов является ориентированный граф, с помощью которого определяется критический путь как наиболее эффективный с точки зрения заданного критерия качества. В середине 50-х годов XX в. в теории графов сформировалась потоковая задача поиска искомого пути на сетке. Соответствующая методология получила название *сетевое планирование и управление* (СПУ) [6].

В терминологии теории графов сетевым графиком является конечный ориентированный граф без контуров, в котором имеется единственная вершина с отсутствующими прообразами и единственная вершина, не имеющая образов. Сетевым графиком можно назвать ориентированную транспортную сеть с одним входом и одним выходом, в которой нет путей с повторяющимися вершинами. Узлы в этой сетке — *события*, а дуги — *операции* (работы). Последние изображаются *стрелками (дугами)*, которые соединяют *вершины (узлы)*, представляющие события. Начало

и окончание любой работы описываются парой событий, которые называются *начальным* и *конечным* событиями. Для указания конкретной работы используют код работы (i, j) , состоящий из номеров начального (i -го) и конечного (j -го) событий. Таким образом, концы входящих в событие и начала выходящих из него дуг имеют свой кодовый признак (номер) [7], [8].

В сетевом графике реализованы два принципа:

- операция начинается только тогда, когда достигнуто ее начальное событие;
- событие считается достигнутым, если выполнены все операции, которые являются входами.

Для расчета сетевого графика обычно используют два метода:

- **метод критического пути** (*Critical Path Method* (CPM)), где длительность операций однозначно определена и задана конкретными числами [9];
- **метод анализа и просмотра программ** (*Program Evaluation and Review Technique*), где длительность операций не определена, и их задают с помощью определенных вероятностных оценок [10].

Цель обоих методов одинакова и состоит в определении минимальной продолжительности выполнения всех операций проекта. Оказалось, что для определения минимального срока выполнения всего проекта, графической и математической моделью которого является сетевой график, нужно найти в конфигурации ту длину пути, которая соединяет начальный узел с конечным. Этот путь называют **критическим путем**. Операции, которые в него входят, называют **критическими**. Они должны быть выполнены в определенный срок, поскольку их задержка приведет к увеличению длительности проекта. Все остальные операции не критические. Они имеют резерв времени. Следовательно, основное внимание должно быть сосредоточено на критических работах [11], [12].

В большинстве известных методов вычисление критического пути производится табличным способом (с применением табличного процессора) посредством вычисления ранних и поздних времен наступления событий в узлах сетевого графика. Способ этот рутинный и затратный по времени, к тому же он не обладает универсальностью для различных структур сетевой модели. Указанные обстоятельства, а также значительно возросшая мощность современных вычислительных сред, в частности среды MatLAB (version 7.9.0.529, R2009b, august 12, 2009, license number: 161051), подтолкнули авторов к созданию универсального алгоритма вычисления критического пути на основе компьютерной модели сетевого графика. С этой целью для построения сетевой модели используется математический аппарат дискретного программирования, в котором искомые переменные принимают только целочисленные или логические (булевы или бинарные) значения: нуль или единицу [13].

Математическую модель сетевой задачи построим на примере сетевого графика (рис. 1), заданного в виде ориентированного графа, который состоит из девяти узлов (событий) и 13 дуг (операций). Нужно найти критический путь от узла № 1 к узлу № 9.

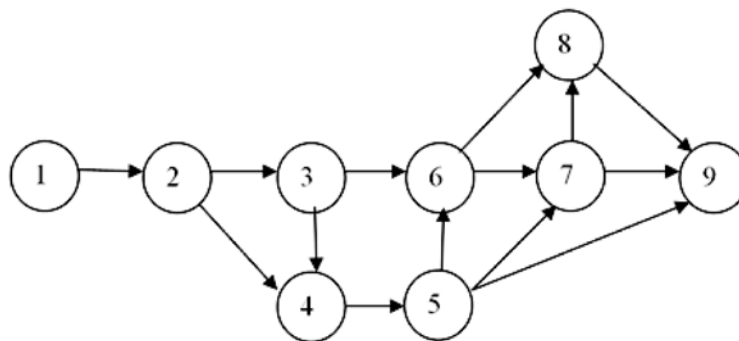


Рис. 1. Исходный ориентированный граф

Численные значения продолжительности операций (дуг), а также индексов, соответствующих началам и концам операций (дуг), приведены в табл. 1. Неизвестными в этой задаче являются

бинарные переменные x_{ij} , имеющие смысл наличия или отсутствия факта принадлежности операции (дуги) критическому пути.

Таблица 1

Исходные данные по операциям сетевого графика

Начало i	1	2	2	3	3	4	5	5	5	6	6	7	7	8
Конец j	2	3	4	4	6	5	6	7	9	7	8	8	9	9
Продолжительность T_{ij}	3	8	5	5	6	9	3	2	5	4	7	6	5	10

Тогда математическую модель задачи можно записать как совокупность следующих составляющих:

– целевой функции

$$F = \sum_{i,j} T_{ij} x_{ij} \rightarrow \min; \quad (1)$$

– ограничения-баланса (в каждом узле)

$$A_2 - A_1 = B \in \begin{cases} 1, & k = 1; \\ 0, & 1 < k < 9; \\ -1, & k = 9; \end{cases} \quad (2)$$

– входного потока

$$\sum_{j=2}^9 x_{ij} = \begin{cases} x_{ij} = 1, & j = k \\ x_{ij} = 0, & j \neq k \end{cases} = A_1, \quad k \in [1, 9]; \quad (3)$$

– выходного потока

$$\sum_{i=1}^9 x_{ij} = \begin{cases} x_{ij} = 1, & i = k \\ x_{ij} = 0, & i \neq k \end{cases} = A_2, \quad k \in [1, 9]; \quad (4)$$

– граничного условия $x_{ij} \in [0, 1] \geq 0, \quad i = 1, \dots, 9; \quad j = 2, \dots, 9,$

где x_{ij} — бинарные переменные состояния, фиксирующие факт принадлежности каждой из заданной в табл. 1 операции (дуги) критическому пути; i, j — индексы, соответствующие началам и концам дуг; T_{ij} — продолжительность операции (дуги) с индексами (i, j) ; k — номер (индекс) узла сетевого графика.

Левая часть ограничения (3) представляет сумму входящих потоков узла, т. е. суммируются потоки по тем дугам, концы которых совпадают с поточным узлом. Аналогично, левая часть ограничения (4) представляет собой сумму выходящих из узла потоков, т. е. суммируются потоки по тем дугам, начала которых совпадают с поточным узлом.

Таким образом, требуется найти вектор неизвестных из целевой функции (1), чтобы выполнялись условия сохранения баланса потоков для каждого узла в выражении (2):

- для узла-источника — $Выход - Вход = 1$;
- для промежуточных узлов — $Выход - Вход = 0$;
- для узла-стока — $Выход - Вход = -1$;
- все неизвестные — неотрицательные переменные.

Требуется среди множества решений системы (1) — (4) найти такое неотрицательное решение, которое составляет минимум целевой функции (1).

Реализация в среде MatLAB. Для решения задачи дискретного программирования в MatLAB используется универсальная функция `linprog`. Область поиска для нее задается следующими условиями [14]:

$$A * x \leq b \text{ — линейные неравенства (A — матрица, b — вектор);}$$

$A_{eq} * x = b_{eq}$ — линейные уравнения (A_{eq} — матрица, b_{eq} — вектор);
 $lb \leq x \leq ub$ — ограничения на координаты (lb — вектор нижних границ и ub — вектор верхних границ).

Целевая функция $f^T * x$ в `linprog` задается вектором коэффициентов f .

Форма обращения к этой функции:

$[x, J] = \text{linprog}(f, A, b, A_{eq}, b_{eq}, lb, ub)$,

где J — значение целевого функционала.

Для обращения к функции `linprog` подготовим входную информацию:

– вектор коэффициентов целевой функции

$f = [3 \ 8 \ 5 \ 5 \ 6 \ 9 \ 3 \ 2 \ 5 \ 4 \ 7 \ 6 \ 5 \ 10]$;

– массив номеров узлов

$Yz = [1; 2; 3; 4; 5; 6; 7; 8; 9]$;

– массив индексов i — начало дуг (операций)

$Dn = [1; 2; 2; 3; 3; 4; 5; 5; 5; 8; 6; 7; 7; 8]$;

– массив индексов j — концов дуг (операций)

$Dk = [2; 3; 4; 4; 6; 5; 6; 7; 9; 7; 8; 8; 9; 9]$;

– матрица коэффициентов системы линейных равенств (уравнений баланса), иначе, матрица инцидентий

$A_{eq} = A2 - A1$,

где $A1 = []$ — матрица коэффициентов линейных ограничений потоков по входу (вычисляется автоматически по заданному массиву D_k индексов j -концов дуг путем циклических итераций);

$A2 = []$ — матрица коэффициентов линейных ограничений потоков по выходу (вычисляется автоматически по заданному массиву D_n индексов i -начала дуг путем циклических итераций);

– вектор свободных (правых) членов системы линейных равенств баланса

$b_{eq} = [1; 0; 0; 0; 0; 0; 0; 0; -1]$;

– вектор нижних границ переменных

$lb = \text{zeros}(14, 1)$;

– вектор верхних границ переменных

$ul = []$.

По синтаксису функции `linprog` находится минимум. В случае максимизации критерия необходимо сменить знак коэффициентов целевой функции f либо поставить знак «минус» перед именем этого вектора:

$[x, J] = \text{linprog}(-f, [], [], A_{eq}, b_{eq}, lb, [])$.

Для реализации алгоритма разработана программа в кодах MatLAB и получены следующие результаты.

Матрица коэффициентов системы линейных ограничений по потокам (в ориентированном графе — матрица инцидентий):

$$A_{eq} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & -1 \end{bmatrix},$$

Вектор переменных состояния (дуг)

$x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12} \ x_{13} \ x_{14}] = [1 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0]$.

Длина минимального пути — критического пути $J_{\min} = 22$.

Если в качестве критерия оптимизации задается максимальное значение целевого функционала J , то вектор переменных состояния (дуг)

$$x=[x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8 \ x_9 \ x_{10} \ x_{11} \ x_{12} \ x_{13} \ x_{14}]=[1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1],$$

а значение функционала J соответствует длине максимального пути: $J_{\max} = 48$.

Вычислительный алгоритм позволяет получить вектор состояния x для любых промежуточных значений J в диапазоне от 22 до 48 путем введения ограничений-неравенств. Расчетные значения для различных J приведены в табл. 2.

Таблица 2

Значения вектора состояния для экстремальных и промежуточных значений J

Экстремальные и промежуточные значения	Значения вектора состояния													
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}
$L_{\max} = 48$	1	1	0	1	0	1	1	0	0	1	0	1	0	1
$L_{\min} = 22$	1	0	1	0	0	1	0	0	1	0	0	0	0	0
$J = 25$	1	0	1	0	0	1	0	1	0	0	0	0	1	0
$J = 30$	1	1	0	1	0	1	0	0	1	0	0	0	0	0
$J = 35$	1	0	1	0	0	1	0	1	0	0	0	1	0	1
$J = 40$	1	0	1	0	0	1	1	0	0	1	0	1	0	1
$J = 45$														

Анализ результата. Найденные критические пути в выделенных графах табл. 3 представляют оценку максимальной продолжительности по вектору состояния, равную 48 ч.

Таблица 3

Операции с выделенными столбцами критических работ

Начало i	1	2	2	3	3	4	5	5	5	6	6	7	7	8
Конец j	2	3	4	4	6	5	6	7	9	7	8	8	9	9
Продолжительность T_{ij}	3	8	5	5	6	9	3	2	5	4	7	6	5	10

Выводы

1. Разработан эффективный алгоритм оптимизации временных затрат и материальных ресурсов в системах материальных грузопотоков экстремальными методами на основе компьютерных моделей для применения в логистических системах.

2. Предложена модель для выполнения технологических расчетов, представленная уравнениями целевой функции, ограничениями баланса (в каждом узле), входным и выходным потоками при соответствующих ограничениях. Введены бинарные переменные состояния, что позволило фиксировать принадлежность каждой заданной операции (дуги) соответствующему критическому пути и использовать для расчетов функции линейного программирования, содержащиеся в инструментарию Optimization Toolbox среды MatLAB.

3. Корректность и эффективность предложенных решений подтверждена на конкретном примере автоматизации расчета критического пути с полным анализом возможных вариантов с размерностью вектора состояния $\dim(X) = 14$.

4. Приведенная модель расчета, реализованная в кодах MatLAB, может быть использована не только для автоматизации логистических систем на водном транспорте, но и в телекоммуникационных системах пакетной передачи данных [3], в системах оптимизации маршрутных карт российских регионов [15], а также в других системах автоматизации и управления высокой размерности [15], [16] различного назначения.

СПИСОК ЛИТЕРАТУРЫ

1. *Гузаиров М. Б.* Моделирование транспортной сети поставок в строительной индустрии / М. Б. Гузаиров, В. А. Тарасова // Вестник Уфимского государственного авиационного технического университета. — 2008. — Т. 10. — № 2. — С. 58–63.
2. Разработка технологических дорожных карт // Российские нанотехнологии. — 2009. — Т. 4. — № 3–4. — С. 10–12.
3. *Слядников Е. Е.* Моделирование распределенных информационно-телекоммуникационных систем пакетной передачи данных / Е. Е. Слядников // Известия Томского политехнического университета. — 2008. — Т. 313. — № 5. — С. 55–60.
4. *D'Ambrosio C.* Mathematical programming techniques in water network optimization / C. D'Ambrosio, A. Lodi, S. Wiese, C. Bragalli // European Journal of Operational Research. — 2015. — Vol. 243. — Issue 3. — Pp. 774–788.
5. *Reich D.* A linear programming approach for linear programs with probabilistic constraints / D. Reich // European Journal of Operational Research. — 2013. — Vol. 230. — Issue 3. — Pp. 487–494.
6. *Оре О.* Теория графов / О. Оре. — М.: Либроком, 2009. — 354 с.
7. *Бояринцева Т. И.* Теория графов: методические указания к выполнению домашнего задания по курсу «Дискретная математика» / Т. И. Бояринцева, А. А. Мاستихина. — М.: МГТУ им. Н. Э. Баумана, 2014. — 40 с.
8. *Рассел Д.* Диаграмма состояний (теория автоматов) / Д. Рассел. — Изд-во: VSD, 2012. — 96 с.
9. *Харари Ф.* Теория графов / Пер. с англ. и предисл. В. П. Козырева / Под ред. Г. П. Гаврилова. — Изд. 2-е / Ф. Харари. — М.: Едиториал УРСС, 2003. — 296 с.
10. *Этов В. И.* 08.01-01В.254. Параллельный алгоритм для определения критического пути в диаграмме PERT. Paralleel algorithm for determining critical paths in PERT CHART. Migita Masahiro, Tada Akio, Itokawa Tsuyoshi, Nakamura Ryozo. Joho Shori Gakkai Ronbunshi=IPSJ J. 2006. 47, N I 7. — С. 2212–2223, 13 Ил. Библ. 6. яп.; Рез. англ. // Автоматика и вычислительная техника. — 2008. — № 1–3. — С. 254–254.
11. *Сахаров В. В.* Алгоритм оптимального планирования группового взаимодействия роботов / В. В. Сахаров, А. А. Чертков, Д. С. Тормашев // Морской вестник. — 2014. — № 4 (52). — С. 119–122.
12. *Зарипова Э. Р.* Дискретная математика: в 3 ч. / Э. Р. Зарипова, М. Г. Кокотчикова. — М.: Изд-во РУДН, 2013. — Ч. III. — 179 с.
13. *Калмыков Г. И.* Древесная классификация помеченных графов / Г. И. Калмыков. — М.: ФИЗМАТЛИТ, 2003. — 192 с.
14. *Мироновский Л. А.* Введение в MATLAB: учеб. пособие / Л. А. Мироновский, К. Ю. Петрова. — СПб.: СПбГУАП, 2005. — 122 с.
15. *Вайнберг Аллен А.* Графы для анализа структурных соотношений между переменными и их приложение к изучению российских регионов / Аллен А. Вайнберг // Прикладная эконометрика. — 2008. — № 4. — С. 42–70.
16. *Бурков В. Н.* Теория графов в управлении организационными системами / В. Н. Бурков, А. Ю. Заложнев, Д. А. Новиков. — М.: СИНТЕГ, 2001. — 124 с.

AUTOMATION TO DEFINE CRITICAL WAY IN LOGISTICAL SYSTEM

In modern conditions the product's life cycle duration is getting short, and changes the rate in the production and logistics technologies all above applies to the logistic material flow of additional requirements. They need to be more flexible with regard quick to tasks and reduction time investment and funds with adaptation to the constantly changing demands on the system. For efficiency and quality rise the analysis and synthesis with complex logistics networks topology is realized by a computer modeling methods for networks with a directed graph that identifies critical path. Effective processes estimation is executed. As the result the algorithm and time amount optimization procedure are considered that takes to move freight from the starting point to the final destination at the specified quality criteria. Algorithm implement realization program in MatLAB codes is constructed. According to the program and algorithm critical path definition example is discussed which is the maximum target functionality is considered.

Keywords: automation, algorithm, logistic, directed graph, critical path, the target functionality, model quality criterion.

REFERENCES

1. Guzairov, M. B., and V. A. Tarasova. "Modelirovanie transportnoj seti postavok v stroitelnoj industrii". *Vestnik UGATU* 10.2 (2008): 58–63.
2. "Razrabotka tehnologicheskikh dorozhnykh kart". *Rossijskie nanotehnologii* 4.3-4 (2009): 10–12.
3. Sljadnikov, E. E. "Simulation of distributed information telecommunication systems with batch communication." *Bulletin of the Tomsk Polytechnic University* 313.5 (2008): 55–60.
4. D'Ambrosio, C., A. Lodi, S. Wiese, and C. Bragalli. "Mathematical programming techniques in water network optimization." *European Journal of Operational Research* 243.3 (2015): 774–788.
5. Reich, Daniel. "A linear programming approach for linear programs with probabilistic constraints." *European Journal of Operational Research* 230.3 (2013): 487–494.
6. Ore, O. *Teorija grafov*. Izdatelstvo: Librokom, 2009: 354 p.
7. Bojarinceva, T. I., and A. A. Mastihina. "Teorija grafov: metodicheskie ukazaniya k vypolneniju domashnego zadaniya po kursu «Diskretnaja matematika»". M.: MGTU im. N. Je. Baumana, 2014: 40 p.
8. Rassel, Dzhessi. "Diagramma sostojanij (teorija avtomatov)". Izd-vo: VSD, 2012: 96 p.
9. Harari, F. "Teorija grafov" / Per. s angl. i predisl. V. P. Kozyreva. Pod red. G. P. Gavrilova. Izd. 2-e. M.: Editorial URSS, 2003: 296 p.
10. Jetov, V. I. "08.01-01V.254. Parallelnyj algoritm dlja opredelenija kriticheskogo puti v diagramme PERT. Paralleel algorithm for determining critical paths in PERT CHART. Migita Masahiro, Tada Akio, Itokawa Tsuyoshi, Nakamura Ryojo. Joho Shori Gakkai Ronbunshi=IPSJ J. 2006. 47, NI 7, C. 2212–2223, 13 II. Bibl. 6. Jap.; Rez. Angl." *Avtomatika i vychislitel'naja tehnika* 1-3 (2008): 254–254.
11. Saharov, V. V., A. A. Chertkov, and D. S. Tormashev. "Algoritm optimal'nogo planirovanija gruppovogo vzaimodejstvija robotov". *Morskoj Vestnik* 4(52) (2014): 119–122.
12. Zaripova, Je. R., and M. G. Kokotchikova. "Diskretnaja matematika". *Chast III. Teorija grafov*. M.: Izd-vo RUDN, 2013: 179 p.
13. Kalmykov, G. I. "Drevesnaja klassifikacija pomechennykh grafov". M.: FIZMATLIT, 2003: 192 p.
14. Mironovskij, L. A., and K. Ju. Petrova. "Vvedenie v MATLAB. Uchebnoe posobi". SPb.: SPbGUAP, 2005: 122 p.
15. Allen, A. Vajnberg. "Graphical Methods of Structural Relations between Variables and their Application to Russian Regions (Part Two)". *Applied econometrics* 4 (2008): 42–70.
16. Burkov, V. N., A. Ju. Zalozhnev, and D. A. Novikov. "Teorija grafov v upravlenii organizacionnymi sistemami". M.: Sinteg, 2001: 124 p.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Чертков Александр Александрович – кандидат технических наук, доцент.
ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»
chertkov51@mail.ru
Вардомская Анна Александровна – аспирант.
ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»
kaf_osnivr@gumrf.ru
Дмитриев Александр Александрович – аспирант.
ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»
kaf_osnivr@gumrf.ru

INFORMATION ABOUT THE AUTHORS

Chertkov Alexandr Alexandrovich – Candidate of Engineering, associate professor.
Admiral Makarov State University of Maritime and Inland Shipping
chertkov51@mail.ru
Vardomskaya Anna Alexandrovna – Post-graduate student.
Admiral Makarov State University of Maritime and Inland Shipping
kaf_osnivr@gumrf.ru
Dmitriev Alexandr Alexandrovich – Post-graduate student.
Admiral Makarov State University of Maritime and Inland Shipping
kaf_osnivr@gumrf.ru