

УДК

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ПОИСКА ОПТИМАЛЬНОГО ПУТИ ДЛЯ ГОРНОСПАСАТЕЛЕЙ НА ОСНОВЕ МОДЕЛИ ГОРНОЙ ВЫРАБОТКИ БОЛЬШОЙ ПРОТЯЖЕННОСТЬЮ ВО ВРЕМЯ АВАРИЙНОЙ СИТУАЦИИ

Черненко В.А., Фонов А.М.

*Донецкий Национальный Технический Университет
кафедра автоматизированных систем управления
E-mail: chernenko.valeriya@gmail.com*

Описано представление модели горных выработок на графе. Реализован поиск оптимального пути в графе муравьиным алгоритмом. Построен кратчайший маршрут движения горноспасателей на графе, а так же маршрут движения по точкам. Рассчитан кратчайший путь полного обхода шахтной выработки.

Угольная шахта – сложная производственная система с особо опасными условиями эксплуатации, где существуют не только случайные изменения геологических и других природных условий, но и нарушения правил безопасного ведения горных работ, отказ техники, и другие нежелательные явления. Несмотря на принятые меры безопасности, аварии на горном предприятии неизбежны. Применение компьютерных систем и разработка программного обеспечения дает возможность качественно нового решения сложных проблем горного производства. [2]

Угольная промышленность Украины является потенциально опасной отраслью народного хозяйства страны. В связи с увеличением глубины разработки, протяженности действующих горных выработок, конвейерных линий и энергетических кабельных сетей, с ростом метановыделения и энергоемкости применяемых машин и оборудования опасность возникновения аварий в угольных шахтах остается довольно высокой. Главной задачей охраны труда на шахтах при возникновении аварийной ситуации является обеспечение безопасного выхода людей на свежую струю и на поверхность как можно быстрее. Поэтому была поставлена задача нахождения оптимального безопасного маршрута поиска горнорабочих в выработках большой протяженности во время аварийной ситуации. [1]

Разработанные модели и информационно-аналитические технологии дадут возможность усовершенствовать процесс безопасной эвакуации горняков из горной выработки большой протяженности во время аварийной ситуации, что является задачей государственной важности.

Представление горных выработок на графе. Горные выработки можно представить, используя ориентированный граф. Горные работы ведутся по определенной системе и планированию, используя планограмму развития горных работ. Эта планограмма отображается на планах горных работ по каждому пласту, на котором ведутся работы. Развитие горных работ осуществляется за определенный интервал времени (t) и в определенном месте полезного ископаемого, т.е., осуществляется во времени и в пространстве. Данная информация отображается на маркшейдерских планах горных работ в виде продвижения каждой выработки за определенное время.

Модель горной выработки на плоскости представляет собой граф (сеть маршрутов, рис.1). Путь в графе - это последовательность вершин (без повторений), в которой любые две соседние вершины смежные, причем каждая вершина является одновременно концом одной дуги и началом следующей дуги. Взвешенный граф - это граф, некоторым элементам которого (вершинам, ребрам или дугам) сопоставлены числа. Числа-пометки носят различные названия: вес, длина, стоимость. Длина пути во взвешенном (связном) графе - это сумма длин (весов) тех ребер, из которых состоит путь. Расстояние между вершинами – это длина кратчайшего пути [7].

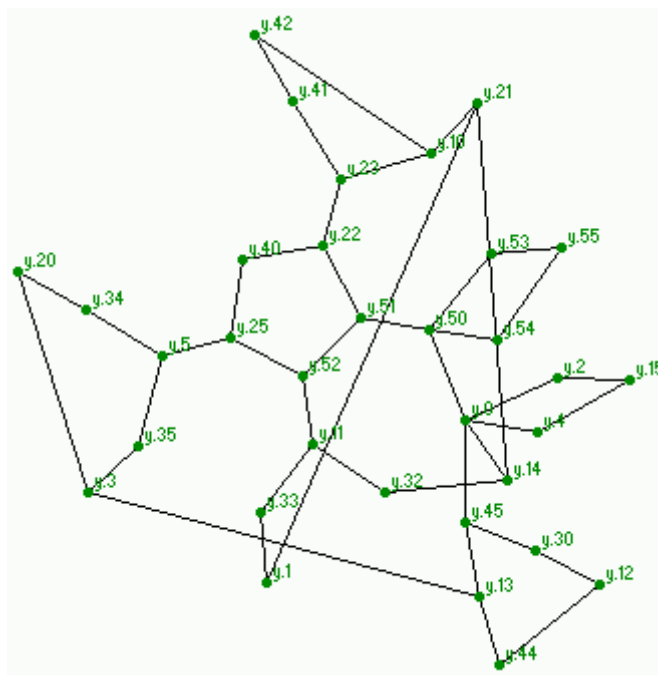


Рисунок 1. Модель горной выработки на плоскости

Постановка задачи. Дана модель горной выработки, которая состоит из n точек проведения горных работ, точки заданы координатами (x, y) . Во время аварии место возникновения и распространения аварии известно (x', y') . Спасатели должны обойти всю выработку, за кратчайшее время t исключая те места, где произошел взрыв, и спасти горнорабочих.

Необходимо найти наиболее выгодный путь обхода всех точек на графе, который реализует модель горной выработки. Точками обхода могут выступать предполагаемые точки нахождения людей в выработке, в роли весов дуг можно использовать количество людей и расстояния до них.

Обзор существующих методов решения поставленной задачи.

Проблему коммивояжера можно представить в виде модели на графе, то есть, используя вершины и ребра между ними. Таким образом, вершины графа (на рис. 1) соответствуют модели шахты (города), а ребра между вершинами и — пути сообщения между этими городами. Каждому ребру можно сопоставить критерий выгодности маршрута, который можно понимать как, например, расстояние между точками. Маршрутом (также гамильтоновым маршрутом) называется маршрут на таком графе, в который входит по одному разу каждая вершина графа. Задача заключается в отыскании

кратчайшего маршрута, обойдя все точки в выработке.

Задачу такого вида можно решить следующими алгоритмами:

- полный перебор
- случайный перебор
- жадные алгоритмы
 - метод ближайшего соседа
 - метод включения ближайшего города
 - метод самого дешёвого включения
- метод минимального остовного дерева
- метод имитации отжига.

Можно предложить следующую простую схему решения задачи коммивояжера: сгенерировать все $n!$ возможных перестановок вершин полного графа, подсчитать для каждой перестановки длину маршрута и выбрать кратчайший. Однако, $n!$ с ростом n растёт быстрее, чем любой полином от n , и даже быстрее, чем n^n . Таким образом, решение задачи коммивояжера методом полного перебора оказывается практически неосуществимым, даже при достаточно небольших n [8].

Все эффективные (сокращающие полный перебор) методы решения задачи коммивояжера — методы эвристические. В большинстве эвристических методов находится не самый эффективный маршрут, а приближённое решение. Зачастую востребованы так называемые *any-time* алгоритмы, то есть постепенно улучшающие некоторое текущее приближенное решение.

На практике применяются различные модификации более эффективных методов:

- метод ветвей и границ
- метод генетических алгоритмов
- алгоритм муравьиной колонии [9].

Наиболее подходящим алгоритмом для решения поставленной задачи является алгоритм муравьиной колонии, один из эффективных полиномиальных алгоритмов для нахождения приближённых решений задачи коммивояжера, а также аналогичных задач поиска маршрутов на графах.

Алгоритм муравьиной колонии. Идея муравьиного алгоритма – моделирование поведения муравьёв, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи и адаптироваться к изменяющимся условиям, находя новый кратчайший путь. При своём движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути. Это элементарное правило поведения и определяет способность муравьёв находить новый путь, если старый оказывается недоступным [9].

Алгоритм:

1. Создаём муравьёв.

- Стартовая точка, куда помещается муравей, зависит от ограничений, накладываемых условиями задачи. Потому что для каждой задачи способ размещения муравьёв является определяющим. Либо все они помещаются в одну точку, либо в разные с повторениями, либо без повторений.
- На этом же этапе задаётся начальный уровень феромона. Он инициализируется небольшим положительным числом для того, чтобы

на начальном шаге вероятности перехода в следующую вершину не были нулевыми.

2. Ищем решения.

- Вероятность перехода из вершины i в вершину j определяется по следующей формуле:

$$P_{ij}(t) = \frac{\tau_{ij}(t)^\alpha \left(\frac{1}{d_{ij}}\right)^\beta}{\sum_{j \in \text{allowed nodes}} \tau_{ij}(t)^\alpha \left(\frac{1}{d_{ij}}\right)^\beta}, \quad (1)$$

где $\tau_{ij}(t)$ – уровень феромона; d_{ij} – эвристическое расстояние; α, β – константные параметры. При $\alpha = 0$ выбор ближайшей точки наиболее вероятен, то есть алгоритм становится жадным. При $\beta = 0$ выбор происходит только на основании феромона, что приводит к субоптимальным решениям. Поэтому необходим компромисс между этими величинами, который находится экспериментально.

3. Обновляем феромон.

- Уровень феромона обновляется в соответствии с приведённой формулой:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k \in \text{Colony that used edge}(i,j)} \frac{Q}{L_k}, \quad (2)$$

где ρ – интенсивность испарения, $L_k(t)$ – цена текущего решения для k -ого муравья, а Q – параметр, имеющий значение порядка цены оптимального решения, то есть $Q/L_k(t)$ – феромон, откладываемый k -ым муравьём, использующим ребро (i, j) [8].

Реализация. Была создана программа GACO (Graph Ant Colony Optimizer) – программа, выполняющая поиск минимального цикла Гамильтона в полном неориентированном взвешенном графе. Данная программа имеет клиент-серверную архитектуру. Клиент – C# - программа, которая выполняет считывание, хранение и начальную визуализацию входных данных. Сервер – М-программа, которая выполняет поиск минимального цикла Гамильтона с помощью алгоритм муравьиных колоний.

Итак, заданы начальные параметры: максимальное число эпох – 100, количество муравьев – 200, коэффициент испарения – 0.1, альфа – 1, бета – 5, модель горной выработки представлена виде графа с известными координатами.

На форме отображается список точек, с помощью которых можно моделировать аварийные ситуации на шахте. При выборе определенных точек, будет считаться, что в этих точках произошла авария, и спасателям не стоит туда заходить. Эти точки в построении оптимального маршрута участия не принимают (рис. 2). После выбора аварийных точек происходит выполнение муравьиного алгоритма на сервере. Рис. 3 демонстрирует выполнение программы, представлен оптимальный путь поиска горнорабочих. На рис. 4 представлен маршрут движения спасателей по точкам.

Работоспособность программы протестирована. Работает адекватно. Аварийные точки в алгоритме не участвуют. [3-6]

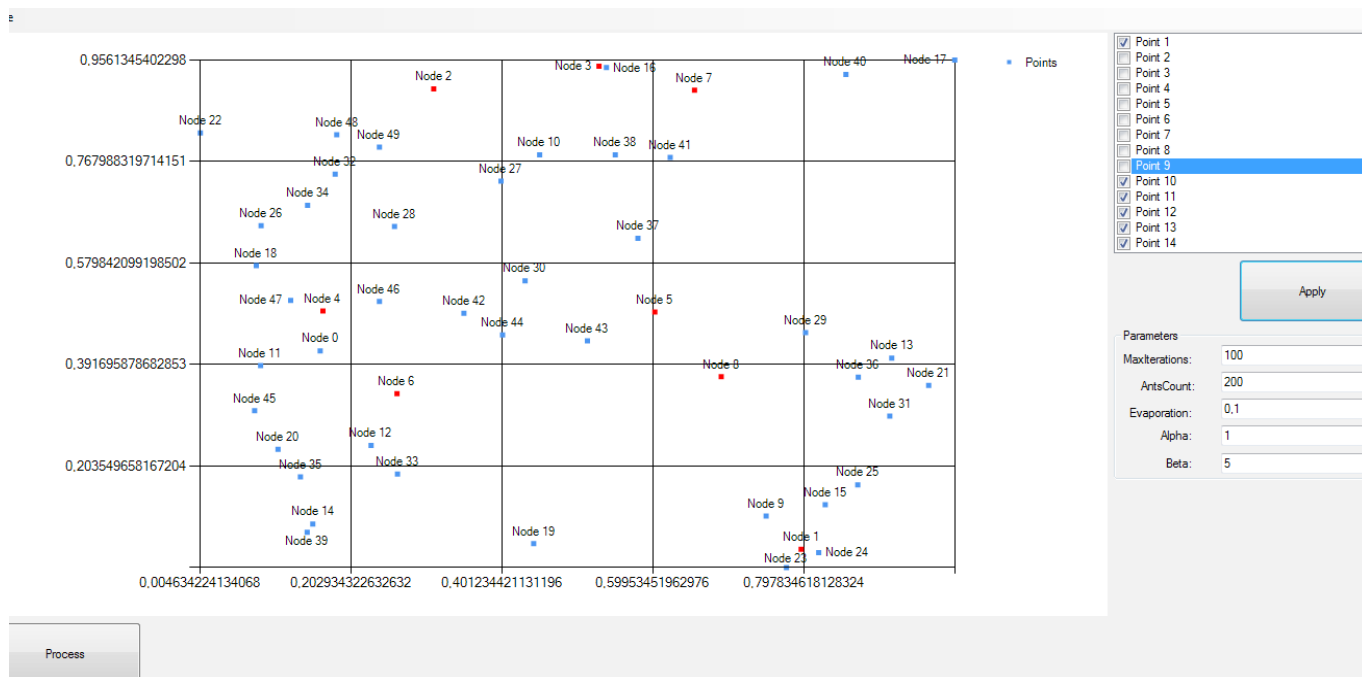


Рисунок 2. Модель горной выработки с аварийными участками

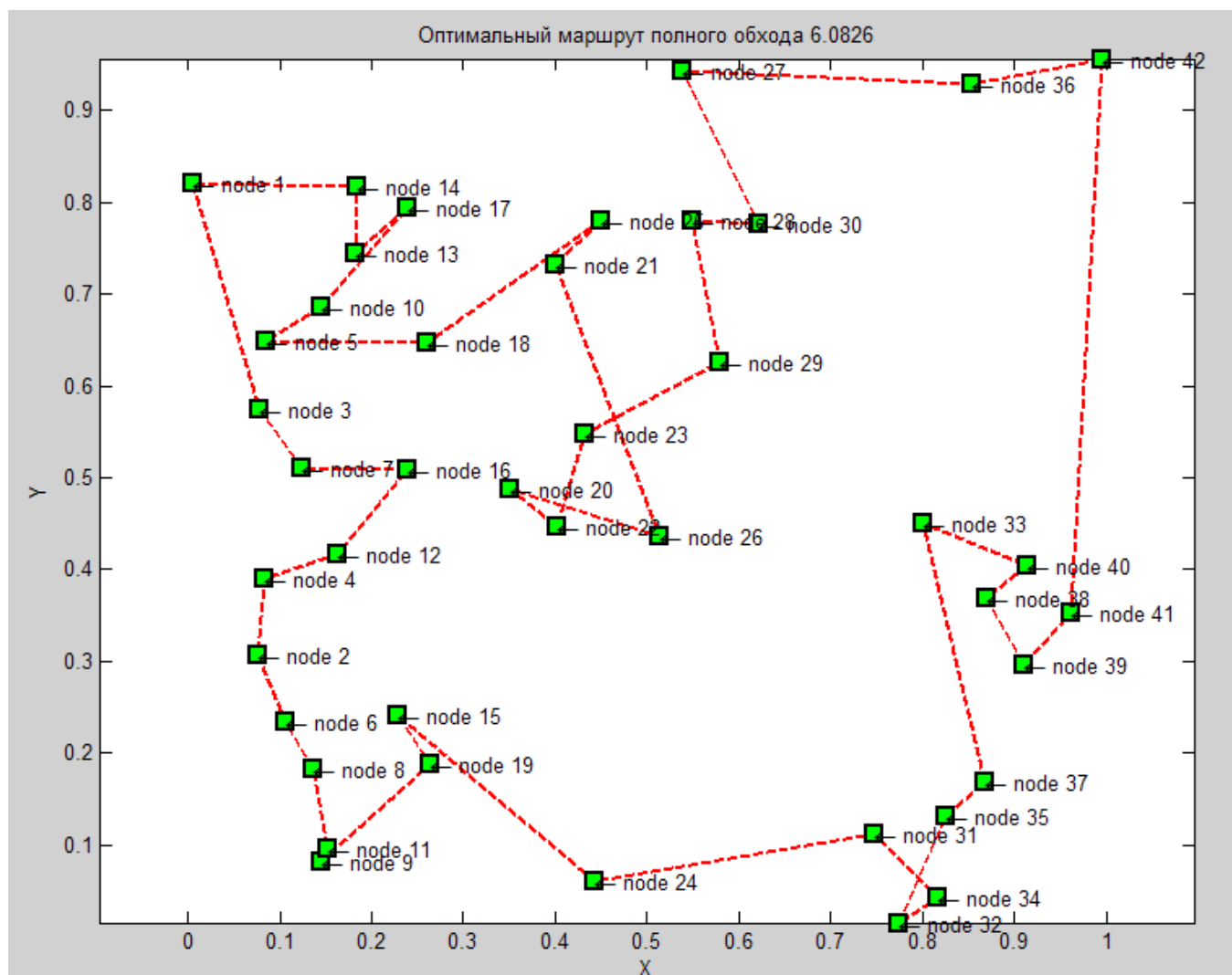


Рисунок 3. Построение кратчайшего пути

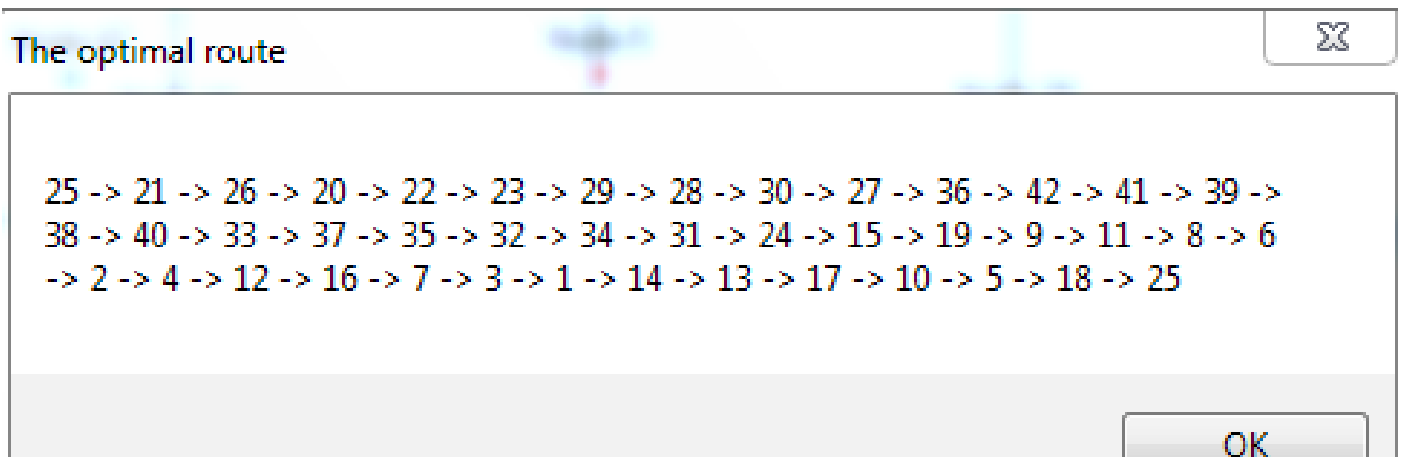


Рисунок 4. Оптимальный путь движения горноспасателей

Выводы

Главной задачей охраны труда на шахтах при возникновении аварийной ситуации является обеспечение безопасного выхода людей на свежую струю и на поверхность как можно быстрее. Человеческая жизнь бесценна, поэтому спасение людей при авариях на шахтном производстве является неотъемлемой частью угольной промышленности.

Разработана программа GACO (Graph Ant Colony Optimizer) –выполняющая поиск минимального цикла Гамильтона в полном неориентированном взвешенном графе. Данная программа имеет клиент-серверную архитектуру.

Определен оптимальный безопасный маршрут поиска горнорабочих в выработках большой протяженности во время аварийной ситуации, с помощью муравьиного алгоритма. Построен маршрут движения спасателей на карте, а так же движение по точкам. Разработанные модели и информационно-аналитические технологии дадут возможность усовершенствовать процесс безопасной эвакуации горняков из горной выработки большой протяженности во время аварийной ситуации, что является задачей государственной важности.

Список источников

- [1] Смоланов С.М. Основи гірничорятувальної справи (навчальний посібник для студентів гірничих спеціальностей вищих навчальних закладів) / С.М. Смоланов, В.І. Голінько, Б.А. Грядущий. - Дніпропетровськ, видавництво НГУ.- 2002.- 267 с.
- [2] Светличный В.П. Обобщенный алгоритм формирования с помощью ЭВМ оптимальных путей движения людей из шахты при возникновении аварийной ситуации / В.П. Светличный, И.Е. Кокоулин, В.А. Хижняк // Известия ВУЗ. Горный журнал. - 1979. - № 10. - С. 26 – 30.
- [3] В.М. Бондарев «Основы программирования» 1998 г., 368 с. «Феникс»
- [4] Ф.А. Новиков «Дискретная математика для программистов» С.-Петербург, 2002 г. 304 с., ил., изд. дом «Питер».
- [5] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley, 1989.

-
- [6] Светличный В.П. Обобщенный алгоритм формирования с помощью ЭВМ оптимальных путей движения людей из шахты при возникновении аварийной ситуации / В.П. Светличный, И.Е. Кокоулин, В.А. Хижняк // Известия ВУЗ. – 1979. – № 10. – С. 26.
- [7] В.А. Черненко, А.М. Фонов. Алгоритм поиска безопасных маршрутов вывода людей из шахты во время аварийных ситуаций. Інформаційні управляючі системи та комп'ютерний моніторинг (ІУС – 2012) / Матеріали ІІ міжнародної науково – технічної конференції студентів, аспірантів та молодих вчених. – Донецьк, ДонНТУ – 2012.
- [8] Чураков Михаил Якушев Андрей, Муравьиные алгоритмы, 2006. <http://rain.ifmo.ru/cat/data/theory/unsorted/ant-algo-2006/article.pdf>
- [9] С.Д. Штовба Муравьиные алгоритмы. http://www.serhiy-shtovba.narod.ru/doc/Shtovba_Ant_Algorithms_ExponentaPro_2003_3.pdf