УДК: 519.854.33

Определение оптимального наряда сил и средств для проведения поисково-спасательных работ

Михненко Н.К.[1]*, Вахромеев П.В.[2], Ктитров С.В.[1]*****

[1]*Национальный исследовательский ядерный университет «МИФИ» (НИЯУ МИФИ), Каширское шоссе, 31, Москва, 115409, Россия*

[2]*Объединение компаний «Радиотехнический институт», г. ул. 8-го Марта, д.10, стр.1, Москва, 127083, Россия*

**e-mail:* abernod137@gmail.com

***e-mail:* vpasha@inbox.ru

****e-mail:* svktitrov@mephi.ru

Аннотация

Рассматривается задача определения оптимального наряда сил и средств для проведения поисково-спасательных работ при допущении, что производится учет только авиационных средств и на каждую зону поиска может быть назначено только одно воздушное судно. Приводится математическая постановка задачи. Для ее решения предлагается двухэтапный метод, причем на каждом этапе требуется решение целочисленной задачи линейного программирования. Разработан и теоретически обоснован алгоритм, позволяющий при определенных условиях уменьшить размерность задач. Для решения задач предлагается использовать симплекс-метод.

Ключевые слова: поисково-спасательные работы, псевдобулевая задача линейного программирования, унимодулярность матриц.

Введение

Одним из существенных факторов успеха поисково-спасательной операции с применением авиационных поисково-спасательных средств является быстрое и своевременное реагирование при максимально эффективном использовании поисковых воздушных судов (ВС) [1].

При организации ПСР основной задачей становится оптимальное назначение ВС на зоны ПСР, причем задача должна быть решена с учетом текущего размещения поисковых средств на аэродромах [2]. Поиск должен быть обеспечен в каждой из зон ПСР, достижимых с заданных аэродромов. Критерием оптимальности является максимизация суммарного времени поиска всеми задействованными средствами. Чтобы обеспечить выполнение обоих условий, предлагается решать задачу в несколько этапов. Задачей первого этапа является максимизация числа ВС, которые можно будет назначить на зоны ПСР с учетом достижимости зон поиска с аэродромов базирования. Необходимость в этом возникает вследствие того, что с точки зрения суммарного времени поиска при определённых условиях может оказаться выгоднее не учитывать какую-либо зону ПСР, что недопустимо.

Постановка задачи

Постановка задачи нахождения максимально возможного наряда сил, который можно будет распределить на зоны ПСР в форме пар «тип ВС/аэродром» (ТА), может быть представлена в следующем виде:

$$f_{вс}(x_{11},..,x_{km}) = \sum_{i=1}^{k}\sum_{j=1}^{m} x_{ij} \to \max \qquad (1)$$

при ограничениях:

$$\begin{cases} \sum_{i=1}^{k} x_{ij} \le 1,\ j = 1,..,m; \\ \sum_{j=1}^{m} x_{ij} \le h_i,\ i = 1,..,k; \\ x_{ij} \in \{0,1\}, \end{cases} \qquad (2)$$

где m – число зон ПСР; k – число различных ТА; h_i – количество ВС $i^{ой}$ ТА; переменная x_{ij} может принимать значения 1 или 0 в зависимости от того, выделено ли ВС $i^{ой}$ ТА на $j^{ую}$ зону ПСР.

После того как найден максимально возможный наряд сил, можно получить условия, формирующие ограничения в задаче линейного программирования для максимизации суммарного времени поиска:

$$f_{вр}(x_{11},..,x_{km}) = \sum_{i=1}^{k}\sum_{j=1}^{m} (t_{ij} \cdot x_{ij}) \to \max \qquad (3)$$

при ограничениях:

$$\begin{cases} \sum_{i=1}^{k} x_{ij} \le 1,\ j = 1,..,m; \\ \sum_{j=1}^{m} x_{ij} \le h_i,\ i = 1,..,k; \\ \sum_{i=1}^{k}\sum_{j=1}^{m} x_{ij} = f_{вс}(x_{11},..,x_{km}); \\ x_{ij} \in \{0,1\}, \end{cases} \qquad (4)$$

где t_{ij} – время поиска в $j^{ой}$ зоне ПСР для $i^{ой}$ ТА, с учетом времени вылета до зоны ПСР и возврата на аэродром; $f_{вс}(x_{11},..,x_{km})$ – значение целевой функции (1).

Методы решения поставленных задач

Предлагается к рассмотрению 2 метода, позволяющие решить поставленные задачи:

- симплекс-метод;
- методы, основанные на алгоритмах решения задач булевой выполнимости (SAT).

Симплекс-метод – это итеративный процесс направленного решения системы уравнений по шагам, который начинается с опорного решения и в поисках лучшего варианта движется по угловым точкам области допустимого решения, улучшающих значение целевой функции до тех пор, пока целевая функция не достигнет оптимального значения.

Необходимо подчеркнуть, что поставленные задачи принадлежат к классу псевдобулевых (ПБ) задач линейного программирования (ЗЛП). В большинстве случаев, для решения подобного класса задач применяются алгоритмы, основанные на симплекс-методе, вследствие неприменимости его самого. Но, как будет показано далее, при решении поставленных задач симплекс-метод всегда позволяет получить оптимальное решение.

В основе SAT методов лежит сведение ПБ ограничений к конъюнктивной нормальной форме (КНФ) с целью дальнейшего поиска решений, которые удовлетворяли бы этой форме и обеспечивали бы все большее значение целевой функции.

В последние десятилетия, определители булевой выполнимости (SAT Solvers), базирующиеся на SAT методах, добились значительного прогресса. Рекордная производительность современных SAT Solvers открывает новые перспективы для их применения там, где раньше это считалось возможным лишь условно. В частности, начиная с 2008 года, the Eclipse platform полагается на SAT4j, как на программный продукт, от которого зависят другие составляющие данной платформы (software dependencies) [3].

Но, при использовании SAT-алгоритмов для решения тех или иных задач, существует вероятность того, что время, затрачиваемое на поиск оптимального решения, будет экспоненциальным. Это объясняется природой NP-полноты SAT задач. В данной статье демонстрируется случай неприменимости SAT подхода для решения поставленных задач на примере наилучшего в настоящее время SAT Solver в категории ПБ ЗЛП [4]: Nagoya Pseudo-Boolean Solver (NaPS) v. 1.02b [5].

Алгоритм, позволяющий уменьшить размерность задачи

В процессе анализа поставленных задач было выявлено, что при выполнении определенных условий, связанных с входными данными, размерность задачи можно уменьшить. Эти условия легли в основу разработанного алгоритма, который представлен далее:

Пусть задано k ТА и m зон ПСР, при этом можно выделить зоны, достижимые только одной $i^{ой}$ ТА. Введем следующие обозначения:

- Z_i – число зон, достижимых $i^{ой}$ ТА;

- M_i – число зон, достижимых только $i^{ой}$ ТА;

- H_i – число ВС $i^{ой}$ ТА.

Рассмотрим алгоритм на примере $i^{ой}$ ТА.

1. Если $H_i \ge Z_i$ и $M_i > 0$, то на зоны, достижимые только $i^{ой}$ ТА назначаются ВС.
2. Если $H_i < Z_i$, то на зоны, достижимые только $i^{ой}$ ТА назначаются ВС при условии, что они входят в последние H_i элементов списка зон, достижимых $i^{ой}$ ТА. Данный список упорядочен по возрастанию времени поиска.
3. Если после применения правила п.2 остались зоны, достижимые только данной ТА, на которые не были распределены ВС, из упорядоченного списка п.2 удаляется зона ПСР, которая достигается такой $j^{ой}$ ТА, для которой выполняется условие: $H_j \ge Z_j$, $j = 1,..,k$, $j \ne i$. Затем п.2 повторяется.
4. П.3 повторяется, пока будут оставаться нераспределенные ВС $i^{ой}$ ТА или будет возможно выполнение п.3.

Обоснование применимости алгоритма:

1) Если $H_i \ge Z_i$, то при назначении ВС на зоны, достижимые только $i^{ой}$ ТА, число оставшихся ВС будет не меньше числа оставшихся зон ПСР, достижимых данной ТА, что в любом случае означает отсутствие потери наилучшего решения.

2) Пусть есть $i^{ая}$ ТА, у которой $M_i > 0$ и данные зоны ПСР входят в последние H_i элементов списка зон, достижимых $i^{ой}$ ТА (таблица 1). Данный список упорядочен по возрастанию времени поиска.

Таблица 1

Упорядоченный по возрастанию времени поиска список

Номер зоны ПСР в списке	1	…	$Z_i - H_i + 1$	…	Z_i
Время поиска	t_1	…	$t_{Z_i - H_i + 1}$	…	t_{Z_i}

Пусть, такая зона одна и ее индекс: $Z_i - H_i + 1$. Допустим, что в оптимальном решении не происходит назначения ВС на данную зону, а вместо этого происходит назначение на некую зону $j = 1,..,(Z_i - H_i)$, причем она, очевидно, достигается как минимум двумя ТА, иначе приходим к противоречию. Назначение ВС на данную зону должно иметь своей целью оптимизацию наряда сил ТА, смежных с данной относительной зоны j с целью нахождения наибольшего суммарного времени поиска во всех зонах ПСР. Фактически при таком назначении освобождается 1 ВС у ТА, ВС которой необходимо было бы назначить на данную зону, если бы мы выделили ВС $i^{ой}$ ТА на $j^{ую}$ зону ПСР. Стоит упомянуть, что если бы не было такой необходимости, то вновь пришли бы к противоречию относительно оптимального решения. Ведь тогда не было бы необходимости назначать ВС $i^{ой}$ ТА на данную зону ПСР и фактически терпеть издержки в суммарном времени поиска, т.к. $t_{Z_i - H_i + 1} \succ t_{Z_i - H_i}$.

Но данная необходимость говорит об оптимальности выбора данной зоны ПСР для этой ТА, ведь если это было бы не так, то данная зона была бы просто проигнорирована для нее. Фактически приходим к тому, что заданная зона оптимальна для ТА смежной с $i^{ой}$ относительно $j^{ой}$ зоны ПСР и назначив ВС на эту зону изначально мы приходим к издержкам по времени, как относительно $i^{ой}$ ТА, так и смежной с ней. Приходим к противоречию.

Теперь допустив, что количество подобных зон ПСР больше 1, и проделав аналогичные рассуждения, также придем к противоречию.

Таким образом п.2 справедлив.

3) Пусть есть $i^{ая}$ ТА, у которой $M_i > 0$ и данные зоны ПСР не входят в последние H_i элементов списка зон, достижимых $i^{ой}$ ТА (таблица 1). Данный список упорядочен по возрастанию времени поиска.

Найдем для данной ТА зону, достижимую еще как минимум одной ТА (*промежуточная*). Начинаем обход по всем ТА, которыми можно достигнуть *промежуточной* зоны, за исключением $i^{ой}$, пока не встретим такую (j), для которой $H_j \geq Z_j$. Если это так, то это означает, что на *промежуточную* зону всегда можно назначить ВС найденной ТА, при этом не нарушив ограничения на количество ВС, налагаемое на нее. Вычеркиваем *промежуточную* зону из рассмотрения для $i^{ой}$ ТА, путем удаления ее из списка зон, достижимых $i^{ой}$

ТА. Повторяем данную последовательность действий, пока хотя бы одна из зон, достижимых только $i^{ой}$ ТА не начнет входить в последние H_i элементов соответствующего списка. Исходя из постановки задачи, а именно того, что поиск должен быть обеспечен в каждой из зон ПСР, а также того, что данная зона имеет максимальное время поиска среди зон, достижимых только $i^{ой}$ ТА, приходим к тому, что для оптимального распределения наряда сил, необходимо определить ВС $i^{ой}$ ТА на данную зону.

Доказательство применимости симплекс-метода

В соответствии с постановкой задачи 1 этапа, ограничения (2) можно разделить на 2 непересекающихся множества: одно множество относится к ограничениям каждой из заданных ТА, накладываемым на число ВС, участвующих в поиске потерпевших бедствие, а второе – к ограничениям на число ВС, назначенным на определенную зону ПСР. Определим многогранник $R_1(G) = \{x : Gx \le b, x \ge 0\}$, представляющий допустимое множество для поставленной задачи ЛП и воспользуемся следующей теоремой [6]:

Теорема 1. Если матрица G вполне унимодулярна, то для любого целочисленного вектора b все вершины многогранника $R_1(G)$ целочисленны.

Матрица G принимает следующий вид (таблица 2):

Таблица 2

Общий вид матрицы G для задачи 1 этапа

Первое множество ограничений	1	…	1	0	…	0	…	0	…	0
	0	…	0	1	…	1	…	0	…	0
	0	…	0	0	…	0	…	1	…	1
Второе множество ограничений	1	…	0	1	…	0	…	1	…	0
	0	…	0	0	…	0	…	0	…	0
	0	…	1	0	…	1	…	0	…	1

Матрица G является вполне унимодулярной, что вытекает из нижеследующей теоремы [5]:

Теорема 2. Пусть G – матрица с элементами 0, $+1$, -1, каждый столбец которой содержит не более двух отличных от нуля элементов; G вполне унимодулярна в том и только в том случае, если все ее строки можно распределить по двум непересекающимся множествам с соблюдением следующих условий:

(1) если два ненулевых элемента одного и того же столбца имеют одинаковый знак, то один из них принадлежит первому множеству, а другой второму;

(2) если два ненулевых элемента одного и того же столбца имеют противоположные знаки, то оба они содержатся или в первом множестве, или во втором.

Из теоремы 1 следует, что симплекс-метод пригоден для выполнения поставленной задачи.

До сих пор рассматривалось подтверждение применимости симплекс-метода для поставленной задачи первого этапа. Рассмотрим теперь его применимость к задачи второго этапа (3), (4). В этом случае матрица G принимает следующий вид (таблица 3):

Таблица 3

Общий вид матрицы G для задачи 2 этапа

Первое множество ограничений	1	…	1	0	…	0	…	0	…	0
	0	…	0	1	…	1	…	0	…	0
	0	…	0	0	…	0	…	1	…	1

Продолжение таблицы 3

Второе множество ограничений	1	…	0	1	…	0	…	1	…	0
	0	…	0	0	…	0	…	0	…	0
	0	…	1	0	…	1	…	0	…	1
Ограничение на назначаемый наряд сил	1	…	1	1	…	1	…	1	…	1

Для доказательства применимости симплекс-метода в этом случае необходимо показать, что матрица (таблица 3) вполне унимодулярна.

Воспользуемся следующими двумя теоремами [7]:

Теорема 3 (Хеллер). Пусть $A = (a^i_j)$ − матрица с элементами 0, $+1$, и -1; для двух векторов-строк a^i и a^j положим $a^i > a^j$, когда каждый ненулевой элемент a^j_k строки a^j равен элементу a^i_k строки a^i, стоящему в том же столбце. Если

(1) $a^i_k = a^j_k \neq 0$ при каком-либо $k \Rightarrow a^i > a^j$ или $a^j > a^i$.

(2) $a^i_k = -a^j_k \neq 0$ при каком-либо $k \Rightarrow a^i > -a^j$ или $a^j > -a^i$,

то матрица A вполне унимодулярна.

Теорема 4. Пусть A и B – две матрицы, элементами которых являются 0 и 1 и которые удовлетворяют условию (1) предыдущей теоремы; тогда матрица C, полученная объединением строк матриц A и B, вполне унимодуярна.

Рассмотрим последнюю из этих двух теорем: пусть матрица A состоит из строк первого множества ограничений и сроки ограничения на назначаемый наряд сил (см. таблицу 3), а значит имеет следующий вид (таблица 4):

Таблица 4

Вид матрицы A

1	…	1	0	…	0	…	0	…	0
0	…	0	1	…	1	…	0	…	0
0	…	0	0	…	0	…	1	…	1

1	…	1	1	…	1	…	1	…	1

Матрица B состоит из строк второго множества ограничений и сроки ограничения на назначаемый наряд сил (см. таблицу 3) и соответственно будет выглядеть следующим образом (таблица 5):

Таблица 5

Вид матрицы B

1	…	0	1	…	0	…	1	…	0
0	…	0	0	…	0	…	0	…	0
0	…	1	0	…	1	…	0	…	1
1	…	1	1	…	1	…	1	…	1

Обе эти матрицы удовлетворяют теореме 3, а значит матрица C, полученная объединением строк матриц A и B, вполне унимодулярна. Матрица C примет следующий вид (таблица 6):

Таблица 6

Вид матрицы C

1	…	1	0	…	0	…	0	…	0
0	…	0	1	…	1	…	0	…	0
0	…	0	0	…	0	…	1	…	1

1	…	0	1	…	0	…	1	…	0
0	…	0	0	…	0	…	0	…	0
0	…	1	0	…	1	…	0	…	1
1	…	1	1	…	1	…	1	…	1
1	…	1	1	…	1	…	1	…	1

Удаляя одну из двух последних строк, т.к. они являются повторяющимися и удаление одной из них не окажет влияния на унимодуярность итоговой матрицы, приходим к матрице вида *таблицы 3*. Значит данная матрица вполне унимодулярна.

Из теоремы 1 следует, что симплекс-метод пригоден для выполнения задачи второго этапа.

Проверка эффективности алгоритма, позволяющего уменьшить размерность задачи

В качестве метода, на примере которого проводилась бы проверка эффективности разработанного алгоритма, предлагается метод зондирования решений (МЗР) [8]. МЗР − это метод решения ПБ ЗЛП. В процессе использования данного метода осуществляется контролированный перебор всех решений задачи и поиск оптимального. Вследствие того, что МЗР имеет экспоненциальную сложность – он является особо чувствительным к уменьшению размерности задачи. Последний фактор и обусловил выбор этого метода.

Тесты проводились на ЭВМ с процессором Intel Core i7-5700HQ в среде разработки Visual Studio 2015 в режиме сборки с оптимизацией и без отладочной информации. В результате их проведения были получены следующие результаты по времени поиска оптимального решения (таблица 7):

Таблица 7

Среднее время (в сек.) нахождения оптимального решения без использования разработанного алгоритма (МЗР) и с его использованием (модифицированный МЗР)

Размерность задачи (число переменных)	Число ТА	Число зон ПСР, достижимых только одной ТА	Среднее время (сек.)	
			МЗР	Модифицированный МЗР (ММЗР)
6	6	6	0,016	0,003
14	9	5	0,224	0,009
22	9	4	1,795	0,136
30	9	3	11,919	2,013
38	9	2	78,908	29,149

Продолжение таблицы 7

Размерность задачи (число переменных)	Число ТА	Число зон ПСР, достижимых только одной ТА	Среднее время (сек.)	
			МЗР	Модифицированный МЗР (ММЗР)
46	9	1	790,582	378,605
54	9	0	4753,946	4760,791

Ниже приводится график, иллюстрирующий вышеуказанные результаты (см. рис. 1):

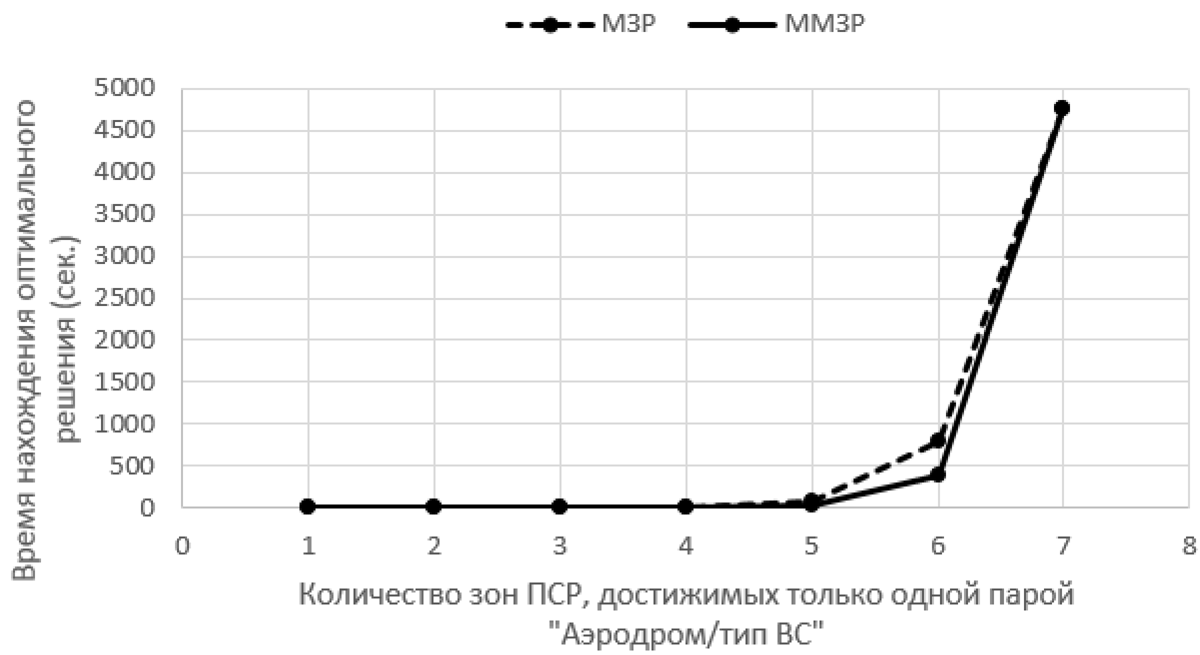

Рисунок 1. График, иллюстрирующий полученные результаты (таблица 7).

На основе полученных результатов можно сделать выводы:

- время выполнения метода зондирования решений без учета алгоритма и с ним сильно возрастает с ростом размерности задачи, что подтверждает его экспоненциальную сложность;
- разработанный алгоритм, при условии присутствия зон ПСР, удовлетворяющих условиям, предусмотренным данным алгоритмом, действительно уменьшает время нахождения оптимального решения;
- разработанный алгоритм, при условии отсутствия зон ПСР, удовлетворяющих условиям, предусмотренным данным алгоритмом, увеличивает время нахождения оптимального решения. По сравнению с общим временем выполнения, данное увеличение незначительно.

Оценка эффективности выбранных подходов к решению поставленных задач

Примеры, на которых проводилась оценка эффективности выбранных подходов можно разделить на 2 группы:

1) с постепенным увеличением числа ТА при неизменном числе наряда сил ВС и числа зон ПСР;
2) с постепенным увеличением числа зон ПСР при неизменном числе ТА и наряде сил ВС.

Подобное разделение объясняется следующей формулой:

$$\text{сложность пространства решений: } \sim O((k+1)^{m}), \tag{5}$$

которая демонстрирует скорость возрастания пространства решений в зависимости от входных данных.

Тесты проводились на ЭВМ с процессором Intel Core i7-5700HQ в среде разработки Visual Studio 2015 в режиме сборки с оптимизацией и без отладочной информации. В результате их проведения были получены следующие результаты по времени поиска оптимального решения (таблица 8, таблица 9):

Таблица 8

Среднее время (в сек.) нахождения оптимального решения с помощью симплекс-метода и NaPS v. 1.02b на примерах 1 группы

Размерность задачи (число переменных)	Число ТА	Среднее число ВС каждой из ТА	Число зон ПСР	Среднее время (сек.)	
				Симплекс-метод	NaPS v. 1.02b
5	1	5	5	< 0,003	< 0,003
10	2	5	5	< 0,003	< 0,003
15	3	5	5	< 0,003	0,008
20	4	5	5	< 0,003	0,012
25	5	5	5	< 0,003	0,024
30	6	5	5	< 0,003	0,044
35	7	5	5	< 0,003	0,112
40	8	5	5	< 0,003	0,156
45	9	5	5	< 0,003	0,443

Продолжение таблицы 8

Размерность	Число ТА	Среднее	Число зон	Среднее время (сек.)

задачи (число переменных)		число ВС каждой из ТА	ПСР	Симплекс-метод	NaPS v. 1.02b
50	10	5	5	< 0,003	0,464
55	11	5	5	< 0,003	0,816
60	12	5	5	< 0,003	1,136
65	13	5	5	< 0,003	1,668
70	14	5	5	< 0,003	2,116
75	15	5	5	< 0,003	3,264
80	16	5	5	< 0,003	4,480
85	17	5	5	< 0,003	9,432
90	18	5	5	< 0,003	9,856
95	19	5	5	< 0,003	20,348
100	20	5	5	< 0,003	20,731
105	21	5	5	< 0,003	27,952
110	22	5	5	< 0,003	42,444

Таблица 9

Среднее время (в сек.) нахождения оптимального решения с помощью симплекс-метода и NaPS v. 1.02b на примерах 2 группы

Размерность задачи (число переменных)	Число ТА	Среднее число ВС каждой из ТА	Число зон ПСР	Среднее время (сек.)	
				Симплекс-метод	NaPS v. 1.02b
12	12	5	1	< 0,003	0,004
24	12	5	2	< 0,003	0,020
36	12	5	3	< 0,003	0,052
48	12	5	4	< 0,003	0,288
60	12	5	5	< 0,003	1,136
72	12	5	6	< 0,003	4,429
84	12	5	7	< 0,003	19,673
96	12	5	8	< 0,003	105,012
108	12	5	9	< 0,003	1297,753

Ниже приводятся графики, иллюстрирующие вышеуказанные результаты для Nagoya-Pseudo Boolean Solver (1.02b) (см. рис. 2, см. рис. 3):

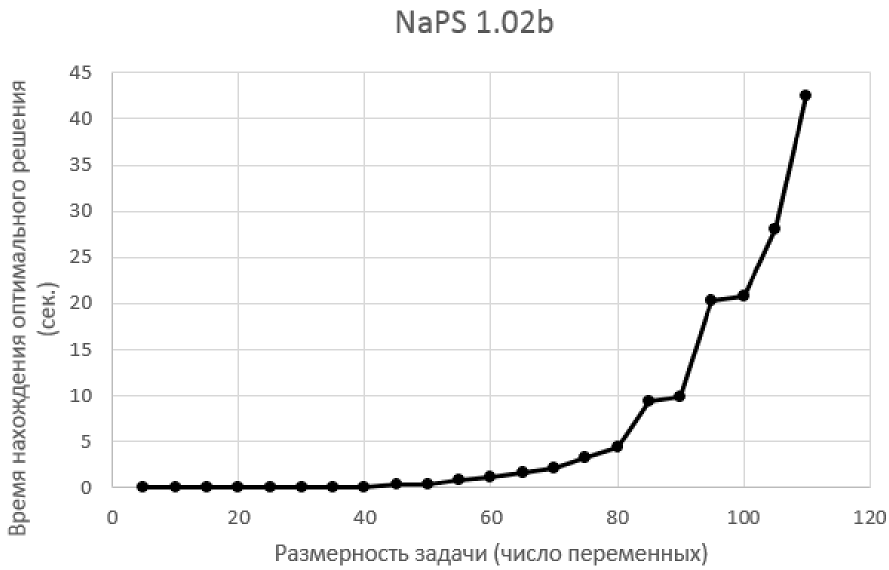

Рисунок 2. График, иллюстрирующий полученные результаты NaPS 1.02b для 1 группы ограничений (таблица 8).

Рисунок 3. График, иллюстрирующий полученные результаты NaPS 1.02b для 2 группы ограничений (таблица 9).

На основе полученных результатов можно сделать выводы:

- для примеров первой группы, т.е. для случая полиномиального возрастания пространства решений (5), время нахождения оптимального решения с помощью Nagoya Pseudo-Boolean Solver (1.02b) имеет полиномиальную зависимость от входных данных;

- для примеров второй группы время нахождения оптимального решения с помощью Nagoya Pseudo-Boolean Solver (1.02b) имеет экспоненциальную зависимость от входных данных, что делает не применимым SAT подход для решения поставленных задач;
- продемонстрирована высокая эффективность симплекс-метода для решения поставленных задач.

Заключение

Рассматривалась задача определения оптимального наряда сил и средств для проведения поисково-спасательных работ при допущении, что производится учет только авиационных средств и на каждую зону поиска может быть назначено только одно воздушное судно.

Были достигнуты следующие результаты:

1) Разработан и теоретически обоснован алгоритм, позволяющий при определенных условиях уменьшить размерность задачи.
2) Продемонстрирована неприменимость NaPS v. 1.02b для решения поставленных задач.
3) Доказана применимость симплекс-метода для решения поставленных задач.

4) Подтверждена высокая эффективность симплекс-метода для решения поставленных задач.

На основе полученных результатов предлагается использовать симплекс-метод для решения подобного класса задач при условии гарантированного получения с его помощью целочисленных решений.

Библиографический список

1. МЧС России, URL: http://www.mchs.gov.ru/dop/sily/Aviacija
2. Немудрый К.В. Аэродромы и аэропорты как один из элементов системы региональной авиации России // Труды МАИ, 2014, № 75: http://www.mai.ru/science/trudy/published.php?ID=49715
3. About Sat4j // Sat4j – the Boolean satisfaction and optimization library in Java, URL: http://www.sat4j.org/allabout.php.
4. Pseudo-Boolean Competition 2016: satisfaction and optimization track: ranking of solvers. Centre de Recherche en Informatique de Lens, URL: http://www.cril.univ-artois.fr/PB16/results/ranking.php?idev=81
5. NaPS (Nagoya Pseudo-Boolean Solver). Sakai Lab./Seki Lab, URL: www.trs.cm.is.nagoya-u.ac.jp/NaPS
6. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация: Алгоритмы и сложность. - М.: Мир, 1984. – 510 с.
7. Берж К. Теория графов и ее применения / Под редакцией Вайнштейна И. А. – М: Изд-во Иностранной литературы, 1962. – 320 с.

8. Загребаев А.М., Крицына Н.А., Кулябичев Ю.П., Шумилов Ю.Ю. Методы математического программирования в задачах оптимизации сложных технических систем. - М.: МИФИ, 2007. – 332 с.