

ЗАДАЧА РАЗМЕЩЕНИЯ ЗАКАЗОВ ДЛЯ СЕТИ АПТЕК РЕГИОНА И ЕЕ ПРИБЛИЖЕННОЕ РЕШЕНИЕ

В. С. Сигаев

Информационно-технологическая сервисная компания, 644110, Омск, Россия

УДК 519.87

Рассматривается задача размещения заказов для сети областных аптек. Строится ее математическая модель в виде задачи целочисленного линейного программирования, доказываемая *NP*-трудность данной задачи, предлагается алгоритм ее приближенного решения. Приводятся результаты экспериментальных исследований для задач со случайными исходными данными.

Ключевые слова: дискретная оптимизация, целочисленное программирование, генетический алгоритм.

In this paper we consider an orders allocation problem for regional pharmacy net. Integer linear programming model for the problem is constructed and analyzed, heuristic algorithm is suggested. The *NP*-hardness for this problem is proved. The results of computational experiment for the problem instances with random data are presented.

Key words: discrete optimization, integer programming, genetic algorithm.

Введение. В данной работе рассматривается задача размещения заказов между поставщиками фармацевтической продукции, обслуживающими сети аптек региона [1]. Склады поставщиков находятся в черте города, тогда как аптеки расположены в области на значительном расстоянии от города. Продукцию доставляют в аптеки сами поставщики по заранее известным маршрутам. Обслуживание сети аптек региона происходит в условиях конкуренции поставщиков. Для аптек одного маршрута описанная ситуация создавала определенные трудности, связанные с распределением заказов между поставщиками. С учетом большого числа типов продукции и количества поставщиков сотрудники аптек каждого маршрута были вынуждены формировать заказы таким образом, чтобы сумма этих заказов по данному маршруту и по каждому поставщику была не меньше пороговой суммы, определяемой рентабельностью маршрута поставщика.

Такая схема взаимодействия поставщиков с аптеками требовала модификации, а именно решения следующих задач: 1) упрощение работы аптек при размещении заказов; 2) нахождение оптимального заказа для всей сети аптек; 3) понижение цены продукции. Для решения поставленных задач была предложена концепция электронных торгов специального вида, в основе которой лежит организация регулируемого взаимодействия поставщиков с аптеками. При данном взаимодействии аптеки формируют заявки на основании суммарного прайс-листа поставщиков.

Результатом электронных торгов является документ, содержащий информацию о том, какой поставщик будет обслуживать аптеку данным видом продукции и в каком количестве. Следует отметить, что каждый вид продукции должен поставляться в аптеку только одним поставщиком. Данное условие обусловлено тем, что при наличии однотипного товара

с разными ценами от разных поставщиков необходимо проведение ряда операций по приведению его к одной цене. В случае малочисленного персонала областных аптек это приводит к потере большого количества времени.

Для реализации рассмотренной концепции разработано программное обеспечение “Центр единого заказа”, которое помимо модулей, решающих задачи по обмену информацией между аптеками и поставщиками, включает модуль, оптимизирующий размещение заказов. Следует отметить, что время решения задачи оптимизации было жестко ограничено и составляло 10 мин.

В данной работе сформулирована математическая модель описанной задачи, исследована ее вычислительная сложность и рассмотрен алгоритм поиска приближенного решения.

1. Математическая модель. Прежде чем представить описание математической модели, приведем список входных данных: $T = \{t_1, \dots, t_m\}$ — множество типов продукции; $P = \{p_1, \dots, p_n\}$ — множество поставщиков; $A = \{a_1, \dots, a_k\}$ — множество аптек; $R = \{r_1, \dots, r_{|R|}\}$, $r_i \subseteq A$, $i = 1, \dots, |R|$ — множество маршрутов; Q_{at} — потребности аптеки a в продукции типа t ; c_{pt} — стоимость единицы продукции типа t у поставщика p ; b_{pt} — запас продукции типа t у поставщика p ; R_p — множество маршрутов поставщика p ($R_p \subseteq R$); V_{rpa} — минимальная суммарная стоимость заказа аптек в маршруте r , достаточная для поездки поставщика p в аптеку a . Данная величина определена с учетом рентабельности открытия маршрута r для поставщика p .

Для решения задачи размещения заказов необходимо определить значения следующих переменных (маршруты одного поставщика не имеют общих аптек, т.е. $r \cap r' = \emptyset$, $r \in R_p$, $r' \in R_p$, $r' \neq r$): x_{pat} — целочисленная переменная, определяющая количество единиц продукции типа t , доставляемое от поставщика p аптеке a ; l_{pat} — бинарная переменная ($l_{pat} = 1$ в случае, если поставщик p доставляет потребителю a продукцию типа t , $l_{pat} = 0$ в случае, если поставщик p не доставляет продукцию потребителю a); d_{pa} — бинарная переменная ($d_{pa} = 1$ в случае, если поставщик p доставляет продукцию потребителю a , $d_{pa} = 0$ в случае, если поставщик p не доставляет продукцию потребителю a).

В силу описанной выше специфики взаимодействия поставщиков и потребителей любое решение задачи размещения заказов должно принадлежать множеству допустимых решений $\Omega = \{(x_{pat}), (l_{pat}), (d_{pa})\}$, определяемому следующими ограничениями:

$$\sum_{a' \in r} \sum_{t \in T} c_{pt} x_{pa't} \geq V_{rpa} d_{pa}, \quad p \in P, \quad r \in R_p, \quad a \in r; \quad (1)$$

$$\sum_{a \in A} x_{pat} \leq b_{pt}, \quad p \in P, \quad t \in T; \quad (2)$$

$$x_{pat} \leq Q_{at} l_{pat}, \quad t \in T, \quad p \in P, \quad r \in R_p, \quad a \in r; \quad (3)$$

$$\sum_{p \in P} l_{pat} \leq 1, \quad a \in A, \quad t \in T; \quad (4)$$

$$l_{pat} \leq d_{pa}, \quad a \in A, \quad t \in T, \quad p \in P; \quad (5)$$

$$x_{pat} = 0, \quad t \in T, \quad p \in P, \quad r \notin R_p, \quad a \in r; \quad (6)$$

$$x_{pat} \geq 0, \quad d_{pa} \in \{0, 1\}, \quad a \in A, \quad p \in P, \quad t \in T. \quad (7)$$

Неравенство (1) определяет условие, согласно которому по каждому поставщику и по каждому маршруту суммарная стоимость заказа аптек должна быть не ниже минимального порогового значения, установленного поставщиком. При этом данное пороговое значение

зависит от расстояния до последней аптеки маршрута, в которую необходимо доставить товар данному поставщику. Неравенство (2) определяет максимально допустимые общие объемы поставок каждого поставщика по каждому виду продукции. Количество привезенного продукта от любого поставщика не должно превышать потребности аптеки в этом продукте, что учтено в неравенстве (3). Каждый вид продукции должен поставляться в аптеку только одним поставщиком (см. (4)). При описанных выше условиях требуется найти лексикографический оптимум в трехкритериальной задаче оптимизации при указанных ниже критериях.

Первым по значимости критерием является минимизация деффектуры, т. е. количества заказанного, но неполученного товара:

$$\varphi^* = \min_{((x_{pat}), (l_{pat}), (d_{pa})) \in \Omega} \varphi((x_{pat})), \quad (8)$$

где

$$\varphi((x_{pat})) = \sum_{a \in A} \sum_{t \in T} \left(Q_{at} - \sum_{p \in P} x_{pat} \right).$$

С целью минимизации общей стоимости закупки для всей сети аптек рассматривался второй по значимости критерий

$$\psi^* = \min_{((x_{pat}), (l_{pat}), (d_{pa})) \in \Omega} \{ \psi((x_{pat})) \mid \varphi((x_{pat})) = \varphi^* \}, \quad (9)$$

где

$$\psi((x_{pat})) = \sum_{a \in A} \sum_{t \in T} \sum_{p \in P} c_{pt} x_{pat}.$$

Третьим по значимости критерием является минимизация максимального количества поставщиков, обслуживающих аптеку, что обусловлено нежелательностью оформления большого числа фактур, необходимых при регистрации товара:

$$\tau^* = \min_{((x_{pat}), (l_{pat}), (d_{pa})) \in \Omega} \left\{ \max_{a \in A} \left(\sum_{p \in P} d_{pa} \right) \mid \varphi((x_{pat})) = \varphi^*, \psi((x_{pat})) = \psi^* \right\}. \quad (10)$$

2. Сложность задачи. В данном пункте проводится анализ вычислительной сложности задачи размещения заказов.

Утверждение. *Задача (1)–(10) является NP-трудной.*

Доказательство. Рассмотрим частный случай задачи (1)–(10), когда имеются только один поставщик и только две аптеки, находящиеся на двух его маршрутах. Запасы поставщика состоят из одного продукта каждого типа: $|T| \geq 3$. Потребности первой аптеки состоят из одного продукта каждого типа, кроме второго, а потребности второй аптеки состоят из одного продукта каждого типа, кроме первого. Стоимости продукции первого и второго типа равны нулю, а стоимости продукции остальных видов являются целыми положительными числами. Минимальная суммарная стоимость заказа по маршруту равна половине стоимости всей продукции поставщика (для упрощения в доказательстве индекс p опускается):

$$|T| \geq 3, \quad |P| = 1, \quad A = \{a_1, a_2\}, \quad R = \{r_1, r_2\}, \quad r_1 = \{a_1\}, \quad r_2 = \{a_2\}; \quad (11)$$

$$b_t = 1, \quad t \in T; \quad (12)$$

$$c_{t_1} = c_{t_2} = 0, \quad Q_{a_1 t_1} = Q_{a_2 t_2} = 1, \quad Q_{a_1 t_2} = Q_{a_2 t_1} = 0; \quad (13)$$

$$Q_{at} = 1, \quad c_t \in Z^+, \quad a \in A, \quad t \in T \setminus \{t_1, t_2\}; \quad (14)$$

$$V_{ra} = \frac{1}{2} \sum_{t \in T} c_t, \quad a \in A, \quad r \in R. \quad (15)$$

При таких исходных данных оптимальное решение со значением φ^* , равным $|T| - 2$, можно получить при условии, что поставщик полностью использовал возможности своего склада, частично удовлетворив потребности каждой из аптек. Если не существует решения, при котором поставщик полностью использует возможности своего склада, то оптимальным решением со значением φ^* , равным $|T| - 1$, является следующее:

$$x_{a_1 t} = l_{a_1 t} = 1, \quad t \in T \setminus \{t_2\}, \quad x_{a_2 t} = l_{a_2 t} = 0, \quad t \in T \setminus \{t_1\}, \quad x_{a_1 t_2} = l_{a_1 t_2} = x_{a_2 t_1} = l_{a_2 t_1} = 0, \quad (16)$$

т. е. поставщик полностью удовлетворил потребности только одной из аптек, в данном случае a_1 . Таким образом, для решения задачи (1)–(10) при предположениях (11)–(15) необходимо определить, существует ли такое решение $((x_{pat}^*), (l_{pat}^*), (d_{pa}^*)) \in \Omega$, что $\varphi((x_{pat}^*)) = |T| - 2$, а в случае его отсутствия выдать решение (16). Ниже показано, что поиск допустимых решений $((x_{pat}), (l_{pat}), (d_{pa})) \in \Omega$, для которых $\varphi((x_{pat})) = |T| - 2$, эквивалентен ответу “да” в задаче “РАЗБИЕНИЕ” [2], которая является *NP*-трудной.

Если $\varphi((x_{pat})) = |T| - 2$, то поставщику необходимо развезти весь товар, имеющийся у него в наличии, т. е.

$$\sum_{a \in A} l_{pat} = 1, \quad t \in T \setminus \{t_1, t_2\}, \quad l_{pa_1 t_1} = l_{pa_2 t_2} = 1, \quad l_{pa_1 t_2} = l_{pa_2 t_1} = 0,$$

$$l_{pat} = x_{pat}, \quad a \in A, \quad t \in T, \quad p \in P,$$

что не нарушает ограничение (4). Отметим, что с учетом $l_{pa_1 t_1} = l_{pa_2 t_2} = 1$ из ограничения (5) следует $d_{pa} = 1$, $a \in A$. Совокупность ограничений (7), (14), (13) и $l_{pat} = x_{pat}$, $a \in A$, $t \in T$, $p \in P$ обеспечивает выполнение ограничения (3). Из условий (2) и $\varphi((x_{pat})) = |T| - 2$ следует, что $x_{a_1 t} + x_{a_2 t} = 1$, $t \in T$, т. е. любой товар $t \in T$ будет поставляться либо аптеке a_1 , либо аптеке a_2 . В результате ограничения (2)–(5) выполняются при предположениях (11)–(15), а ограничение (1) имеет следующий вид: $\sum_{t \in T} c_t x_{a_1 t} \geq \frac{1}{2} \sum_{t \in T} c_t$, $\sum_{t \in T} c_t x_{a_2 t} \geq \frac{1}{2} \sum_{t \in T} c_t$. Выполнение данных неравенств возможно только в случае, если

$$\sum_{t \in T} c_t x_{a_1 t} = \sum_{t \in T} c_t x_{a_2 t} = \frac{1}{2} \sum_{t \in T} c_t.$$

Существование допустимых решений при таком ограничении соответствует тому случаю, когда задача “РАЗБИЕНИЕ” [2] имеет ответ “да”.

3. Генетический алгоритм. В данной работе рассматривается генетический алгоритм (ГА), принадлежащий семейству эвристических случайных методов [3]. В генетическом алгоритме — упрощенной модели эволюции в природе — используется как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде.

Процесс работы ГА представляет собой последовательную смену популяций, состоящих из фиксированного числа особей — пробных точек пространства решений, причем каждая особь представлена строкой символов некоторого алфавита, называемой генотипом. В рассматриваемом случае множеством генотипов является $\Gamma = \{((w_{at}), (\pi_{pu}))\}$, где $w_{at} \in P$ — поставщик, удовлетворяющий потребности аптеки a в продукции типа t . В каждой строке матрицы (π_{pu}) для соответствующего поставщика p фиксируется порядок перебора маршрутов данного поставщика при использовании в функции декодирования. Данный порядок определяется перестановкой, которая записана в строке p матрицы (π_{pu}) . Функция декодирования решения $\Omega' = \zeta(\Gamma)$, где $\Omega' \subseteq \Omega$, вычисляется по следующей схеме:

- 1) для каждого $p \in P$ и для каждого $r \in R_p$ в порядке $\pi_{p1}, \dots, \pi_{p|R_p|}$ выполнять шаги 2–13;
- 2) определить наибольшее по мощности множество \tilde{A} , такое что $\tilde{A} \subseteq r$ и $\max_{a \in \tilde{A}} V_{rpa} \leq \sum_{t \in T} c_{pt} \min(\sum_{a \in \tilde{A}} Q_{at}, b_{pt})$ (в множество \tilde{A} входят те аптеки маршрута r , в которые поставщику p выгодно везти продукцию);
- 3) для каждого $a \in \tilde{A}$ в порядке возрастания величины V_{rpa} выполнять шаги 4–8;
- 4) положить $d_{pa} := 0$;
- 5) для каждого $t \in T$ выполнять шаги 6–8;
- 6) положить $x_{pat} := \min(Q_{at}, b_{pt})$;
- 7) обновить остатки поставщика $b_{pt} := b_{pt} - x_{pat}$;
- 8) если $x_{pat} > 0$, то положить $l_{pat} := 1$, $d_{pa} := 1$, иначе $l_{pat} := 0$;
- 9) для каждого $a \in r \setminus \tilde{A}$ выполнять шаги 10–13;
- 10) положить $d_{pa} := 0$;
- 11) для каждого $t \in T$ выполнять шаги 12–13;
- 12) положить $x_{pat} := 0$;
- 13) положить $l_{pat} := 0$;
- 14) результатом работы функции является $((x_{pat}), (l_{pat}), (d_{pa}))$.

Недостатком данного метода является то, что существует множество частных задач, для которых $\Omega' \subset \Omega$ и часть решений не может быть получена при декодировании генотипов. Однако сравнительно небольшая трудоемкость позволяет исследовать за отведенное время большое количество генотипов, тем самым увеличивая вероятность нахождения решения приемлемого качества.

Обозначим через $\Phi(\gamma) = (\varphi(\zeta(\gamma)), \psi(\zeta(\gamma)), \tau(\zeta(\gamma)))$ функцию приспособленности генотипа $\gamma \in \Gamma$.

Генетический алгоритм, разработанный в данной работе, включает следующие этапы (операторы *Init*, *Selection*, *Crossover*, *Mutation*, *LCS* описаны далее):

1. Построить популяцию $\Pi = \text{Init}(N)$, где N — размер популяции ($N \leq |\Gamma|$, $\Pi \subseteq \Gamma$).
2. До тех пор пока не выполнен критерий остановки, выполнять шаги 2.1, ..., 2.3:
 - 2.1. Выбрать родительские особи $\gamma^1, \gamma^2 := \text{Selection}_s(\Pi)$.
 - 2.2. Создать особи $\hat{\gamma}^1$ и $\hat{\gamma}^2$: $(\hat{\gamma}^1, \hat{\gamma}^2) := \text{Crossover}(\gamma^1, \gamma^2)$.
 - 2.3. Для $\gamma \in \{\hat{\gamma}^1, \hat{\gamma}^2\}$:
 - 2.3.1. Вычислить $\gamma' := \text{LCS}(\text{Mutation}(\gamma))$.
 - 2.3.2. Выбрать $\gamma^w \in \Pi$, такое что $\Phi(\gamma^w) = \text{lexmax}(\Phi(\Pi))$.
 - 2.3.3. Если $\Phi(\gamma^w) = \text{lexmax}(\Phi(\{\gamma', \gamma^w\}))$ и $\gamma' \notin \Pi$, то заменить γ^w на γ' .
3. Результатом работы алгоритма является $\gamma^* \in \Pi$, такое что $\Phi(\gamma^*) = \text{lexmin}(\Phi(\Pi))$.

В рассматриваемом случае для создания популяции Π используется процедура $Init(N)$, которая последовательно заполняет популяцию Π новыми генотипами. Для генерации нового генотипа выбирается множество поставщиков $P' \subseteq P$. Сначала в качестве P' последовательно берутся различные одноэлементные множества из P , затем различные двухэлементные множества из P и т. д. В случае если необходимо выбрать множество P' , которое по количеству элементов больше или равно множеству P , берется все множество P . После выбора нового множества P' создается генотип γ' : для каждого $a \in A$ и $t \in T$ выбирается элемент w'_{at} с равномерным распределением на множестве P' ; перестановка (π'_{pu}) для каждого $p \in P$ генерируется случайным образом. Если $\gamma' \notin \Pi$, то генотип γ' добавляется в популяцию Π , иначе выполняется переход на этап генерации нового генотипа. Процесс заполнения популяции прекращается, когда $|\Pi| = N$.

Для выбора номера очередного “родительского” генотипа используется так называемый оператор s -турнирной селекции, в котором из текущей популяции независимо извлекаются s особей с равномерным распределением и из них выбирается особь, имеющая лексикографически минимальное значение функции приспособленности Φ .

К генотипам выбранных особей применяется оператор $Crossover$, заменяющий часть генов одного генотипа на гены другого. В рассматриваемом случае после применения оператора кроссинговера $(\gamma', \gamma'') := Crossover(\bar{\gamma}, \underline{\gamma})$ с вероятностью $1 - P_{cross}$ получаем $\gamma' = \bar{\gamma}$, $\gamma'' = \underline{\gamma}$, а с вероятностью P_{cross} имеем $\gamma' = ((w'_{at}), (\pi'_{pu}))$ и $\gamma'' = ((w''_{at}), (\pi''_{pu}))$, где

$$w'_{at} = \begin{cases} \bar{w}_{at}, & \text{если } t \in \{t_1, \dots, t_{v_a-1}\}, \\ \underline{w}_{at}, & \text{если } t \notin \{t_1, \dots, t_{v_a-1}\}, \end{cases} \quad w''_{at} = \begin{cases} \underline{w}_{at}, & \text{если } t \in \{t_{v_a}, \dots, t_m\}, \\ \bar{w}_{at}, & \text{если } t \notin \{t_{v_a}, \dots, t_m\}, \end{cases}$$

позиция v_a для каждого $a \in A$ выбрана с равномерным распределением на $\{1, \dots, m\}$.

Формирование набора (π'_{pu}) потомка происходит следующим образом: для каждого поставщика $p \in P$ набор (π'_{pu}) заполняется элементами первого родителя до позиции μ_p , после чего дополняется отсутствующими элементами в порядке их следования во втором родителе:

$$\pi'_{pu} = \begin{cases} \bar{\pi}_{pu}, & \text{если } u \in \{1, \dots, \mu_p - 1\}, \\ \hat{\pi}_{pu}, & \text{если } u \notin \{1, \dots, \mu_p - 1\}, \end{cases} \quad \pi''_{pu} = \begin{cases} \underline{\pi}_{pu}, & \text{если } u \in \{\mu_p, \dots, |R_p|\}, \\ \check{\pi}_{pu}, & \text{если } u \notin \{\mu_p, \dots, |R_p|\}, \end{cases}$$

$$\hat{\pi}_{pl} = \underline{\pi}_{py'_l}, \quad y'_l = \min(i \in \{1, \dots, |R_p|\} | \underline{\pi}_{pi} \notin \{\pi'_{p1}, \dots, \pi'_{pl-1}\}),$$

$$\check{\pi}_{pl} = \bar{\pi}_{py''_l}, \quad y''_l = \min(i \in \{1, \dots, |R_p|\} | \bar{\pi}_{pi} \notin \{\pi''_{p1}, \dots, \pi''_{pl-1}\}).$$

Значение μ_p для каждого $p \in P$ выбрано с равномерным распределением на $\{1, \dots, |R_p|\}$. Таким образом, набор (w'_{at}) потомка, полученного после операции скрещивания, для каждой аптеки $a \in A$ содержит элементы как одного родителя, так и второго [4].

Оба полученных после кроссинговера генотипа поступают на вход оператора мутации, действие которого на наборе (π_{pu}) состоит в последовательном переборе элементов π_{pu} , каждый из которых с вероятностью P_{mut} меняется местами с некоторым элементом $\pi_{p\tilde{u}}$, где \tilde{u} — случайно выбранная позиция.

Затем полученные генотипы подаются на вход оператора локальной корректировки решения LCS . Оператор локальной корректировки успешно применяется в качестве одного из этапов генетического алгоритма [5]. На данном этапе в генотипах $\rho' = ((w'_{at}), (\pi'_{pu}))$ и $\rho'' = ((w''_{at}), (\pi''_{pu}))$ определяются элементы w_{at} , для которых при декодировании генотипа $x_{w_{at}} = 0$. Для таких элементов проводится повторная процедура мутации при условии, что

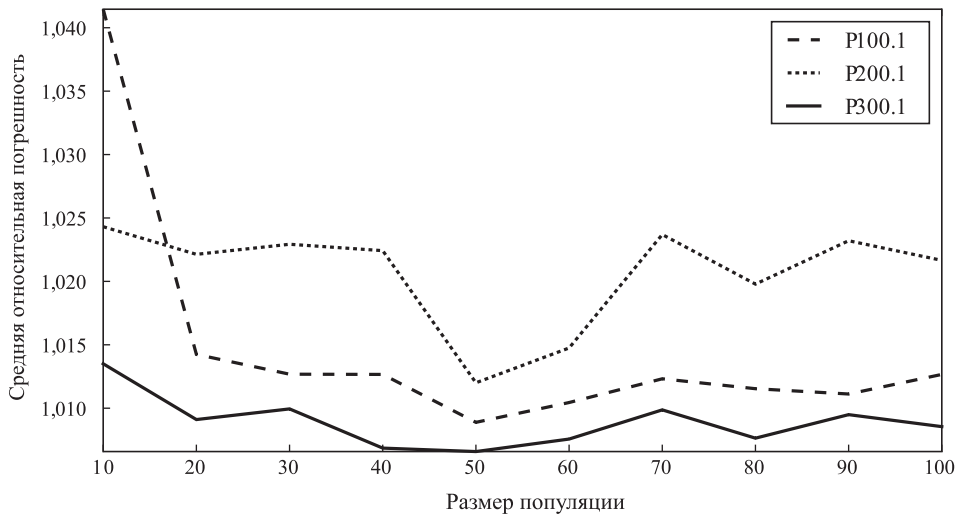


Рис. 1. Погрешность, определенная для различных размеров популяции

поставщики, которые уже обслуживают данную аптеку, имеют приоритет при выборе замены. Целью разработки данного оператора являлась необходимость распределения продукции, которую не могут привезти установленные поставщики, между другими поставщиками.

4. Вычислительный эксперимент. Оптимизационный модуль был запрограммирован на языке C++. Поиск решения в данном модуле основан на генетическом алгоритме, описанном выше. Все задачи, на которых проводились эксперименты, генерировались таким образом, чтобы они максимально соответствовали реальной обстановке: $100 \leq |T| \leq 300$, $|P| = 10$, $|A| = 50$, остальные параметры выбирались случайным образом. На первом этапе вычислительных экспериментов необходимо определить основные настройки генетического алгоритма, к числу которых в данном случае относятся следующие: размер популяции, размер турнира селекции, вероятность кроссинговера и вероятность мутации. По каждому из предложенных параметров генерировались три задачи со значениями $|T| \in \{100, 200, 300\}$. Для сбора статистических данных на каждой из задач было проведено 30 запусков генетического алгоритма. Показателем качества полученных решений на каждой из задач являлась средняя погрешность значения функции $\varphi(x)$ относительно оптимального значения φ^* , т. е. $(\sum_{i=1}^{30} \varphi_i) / (30 * \varphi^*)$, где φ_i — значение, полученное на i -м запуске генетического алгоритма. В качестве оптимального решения использовалось решение, полученное с помощью коммерческого пакета CPLEX 22.9.2 для решения задач частично целочисленного линейного программирования.

Размер популяции, при котором получают решения с лучшим значением целевой функции φ , определялся на построенных случайным образом задачах P100.1 ($|T| = 100$), P200.1 ($|T| = 200$), P300.1 ($|T| = 300$). На каждой из этих задач проведены запуски генетического алгоритма с размерами популяций 10, 20, ..., 100. На рис. 1 приведена средняя относительная погрешность по критерию $\varphi(x)$. Видно, что алгоритм позволяет получить приемлемые результаты в широком диапазоне значений N . Далее было выбрано значение $N = 50$.

На задачах P100.2 ($|T| = 100$), P200.2 ($|T| = 200$), P300.2 ($|T| = 300$) определялся размер турнира s . На каждой из задач проводились запуски генетического алгоритма с размерами турнира 1, 2, ..., 7. На основе анализа полученных результатов, приведенных на рис. 2, для дальнейших экспериментов было выбрано значение $s = 4$.

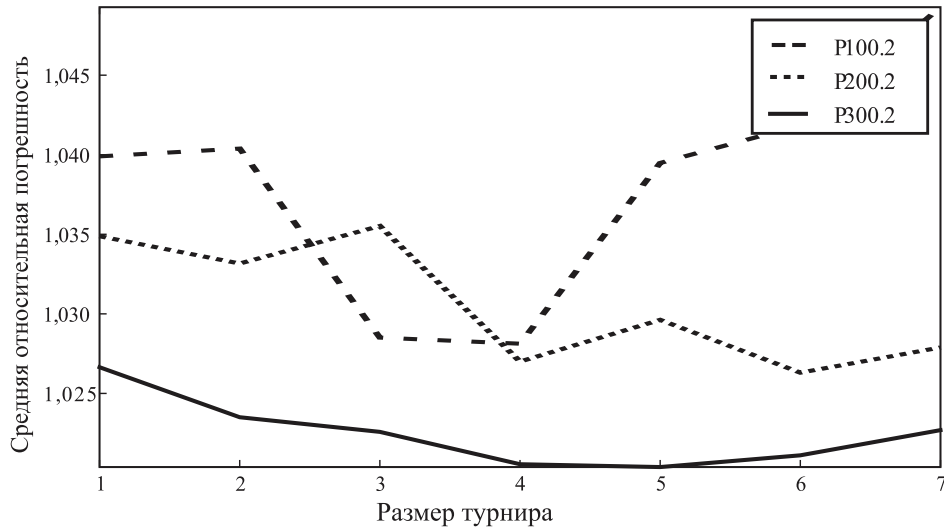


Рис. 2. Погрешность, определенная для различных размеров турнира

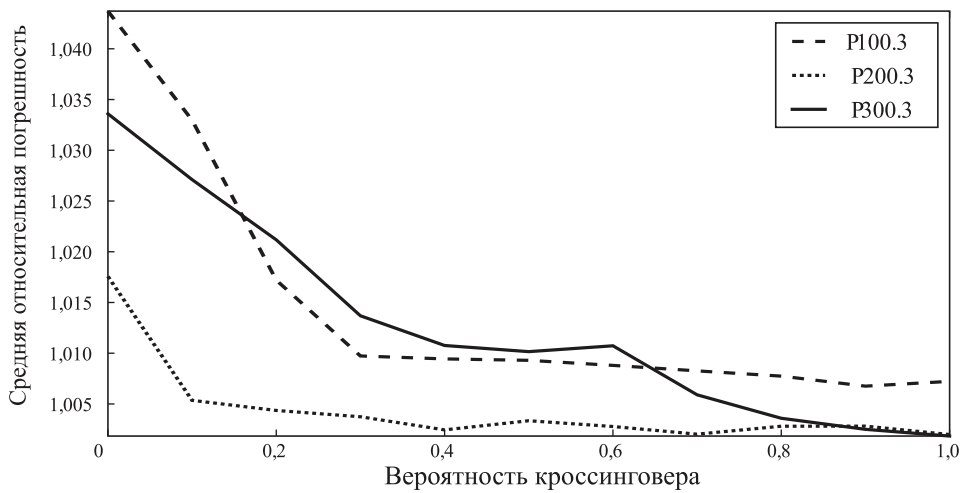


Рис. 3. Погрешность, определенная для различных вероятностей кроссинговера



Рис. 4. Погрешность, определенная для различных вероятностей мутации

При выборе значений вероятности использования оператора скрещивания P_{cross} и оператора мутации P_{mut} на каждом из запусков соответствующие параметры устанавливались равными 0,1; 0,2; ...; 1,0. Результатом вычислительных экспериментов является выбор $P_{cross} = 1,0$ и $P_{mut} = 0,2$ (рис. 3, 4).

Для определения степени влияния процедуры локальной корректировки решения *LCS* на значение целевой функции каждого генотипа проведены дополнительные эксперименты с использованием *LCS* и без использования. Для этого для каждой из задач $P100.5$ ($|T| = 100$), $P200.5$ ($|T| = 200$), $P300.5$ ($|T| = 300$) было проведено 25 запусков генетического алгоритма с использованием *LCS* и без использования. На основе полученных результатов с помощью статистического критерия Манна — Уитни [6] показано, что использование процедуры *LCS* в ГА статистически значимо (с уровнем значимости 0,01) улучшает значение целевой функции генотипа, полученного в результате работы ГА.

Следует отметить, что время поиска оптимального решения задач с использованием программного модуля CPLEX 22.9.2 составило: для задач с $|T| = 100$ — порядка 5 мин, для задач с $|T| = 200$ — порядка 10 мин, для задач с $|T| = 300$ — более 25 мин.

Заключение. В работе построена математическая модель задачи размещения заказов для сети аптек региона. Доказана *NP*-трудность рассмотренной задачи. Проведенные исследования показали перспективность использования генетического алгоритма при поиске приближенного решения задачи размещения заказов для сети аптек региона в системах поддержки принятия решений.

Список литературы

1. СИГАЕВ В. С. Задача размещения заказов для сети аптек региона // Тез. докл. 4-й Всерос. конф. “Проблемы оптимизации и экономические приложения”, Омск, 29 июня – 4 июля 2009 г. Омск: Полиграф. центр КАН, 2009. С. 244.
2. ГЭРИ М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. М.: Мир, 1982. 416 с.
3. HOLLAND J. Adaptation in natural and artificial systems. S. 1.: Univ. of Michigan Press, 1975.
4. REEVES C. R. Genetic algorithms for the operations researcher // INFORMS J. Comput. 1997. V. 9, N 3. P. 231–250.
5. ЕРЕМЕЕВ А. В. Генетический алгоритм для задачи о покрытии // Дискретный анализ и исследование операций. Сер. 2. 2000. Т. 7, № 1. С. 47–60.
6. MANN H. B., WHITNEY D. R. On a test of whether one of two random variables is stochastically larger than the other // Ann. Math. Statistics. 1947. N 18. P. 50–60.

*Сигаев Вячеслав Сергеевич — ведущ. инженер-программист
ООО “Информационно-технологическая сервисная компания”;
e-mail: sigus@mail.ru*

Дата поступления — 8.11.2010