

УДК 004.021

ОПТИМИЗАЦИЯ ДВИЖЕНИЯ ПОГРУЗЧИКОВ НА СКЛАДЕ

Безе В.И., Телятников А.О.

Донецкий национальный технический университет, Украина.

В статье предлагается подход к решению задачи передвижения погрузчиков для изъятия-размещения товара на складе. Рассмотрены часто используемые методы оптимизации размещения товара. Для решения данной задачи был использован волновой алгоритм с учетом времени передвижения погрузчиков.

Общая постановка задачи

Данная статья посвящена разработке информационной системы, которая позволяла бы решить задачу маршрутизации погрузчиков на складе. Информационная система должна учитывать некоторые критерии для построения маршрутов, критерии: маршрут должен быть как можно короче, принцип размещения-изъятия товара должен иметь структуру FIFO (первый пришедший изымается первым, это сделано для того чтобы товар долго не оставался на складе). Так же учитывается ограничение – отсутствие столкновения погрузчиков друг с другом при движении по маршруту.

Задача автоматизации определения приемлемого варианта размещения товаров на складе не является новой для торговли и системы материально-технического снабжения. Суть ее заключается в определении оптимальных мест хранения товара. Размещение товаров на складе играет ключевую роль в оптимизации складских процессов. От того, как товар размещен на складе, зависит и сохранение его качества, и скорость отбора [1]. Разработаны различные методы, предлагающие решать эту задачу, однако она актуальна, так как на данный момент размещение товара не идеально и иногда занимает много времени [2].

Определение правил хранения в информационной системе выполняется путем указания для наименования товара параметров «режим хранения» и «область». Эти же параметры устанавливаются и в ячейках. Таким образом, ячейки для размещения единицы хранения определяются при полном совпадении данных параметров [1].

Основные принципы рациональной транспортировки внутри склада, обеспечивающих эффективную грузопереработку, сводятся к следующему [3]:

- транспортировка грузов должна проходить по маршрутам без столкновений с другими погрузчиками;
- эффективные алгоритмы построения оптимальных маршрутов для подъемно-транспортных средств.

Критериями оптимизации такой задачи являются:

- минимальное расстояние перевозки товара;
- выбор ближайшего погрузчика к места отгрузки.

Оптимизация системы хранения

Рассмотрим методы оптимизации, применяемые для поставленной задачи:

1) Алгоритм Кларка-Райта.

Алгоритм Кларка-Райта (Clarke and Wright) [4, 5] – это один из самых известных алгоритмов для решения задачи маршрутизации транспорта (ЗМТ). Его идея основана на процессе слияния мелких маршрутов в более крупные, проводимого до тех пор, пока есть возможность уменьшить суммарную стоимость объезда. Особую роль в этом алгоритме играет понятие “сбережения” (saving) – это снижение общей стоимости решения, получаемое при объединении двух маршрутов. Рассмотрим ситуацию, когда маршрут $(0, \dots, i, 0)$ и маршрут $(0, j, \dots, 0)$ могут быть совмещены в единую последовательность $(0, \dots, i, j, \dots, 0)$. Сбережением является изменение расстояния, равное $s_{ij} = c_{i0} +$

$c_{0j} - c_{ij}$, если оно больше нуля, где c_{ij} – расстояние между соответствующими вершинами. Алгоритм применяется в случаях, когда количество экипажей не определено заранее и его можно вычислять в ходе работы. Его можно использовать как для симметричных задач, так и для несимметричных, но в показано, что качество работы для симметричных случаев заметно ухудшается. Известны два варианта реализации: параллельный и последовательный. В обоих случаях для них требуется предварительное выполнение подготовительного этапа. Он заключается в:

1. Вычислении сбережений $s_{ij} = c_{i0} + c_{0j} - c_{ij}$ для $i, j = 1, \dots, n$ и $i \neq j$.
2. Создании n маршрутов транспортных средств $(0, i, 0)$ для $i = 1, \dots, n$.
3. Сортировке сбережений в порядке убывания.

Рассмотрим сначала параллельный вариант алгоритма:

1. Просматриваем построенный список сбережений с его начала и выполняем следующие действия:
2. Для текущего элемента списка s_{ij} определяем, существуют ли два маршрута, один из которых содержит дугу или ребро $(0, j)$, второй – дугу или ребро $(i, 0)$, и которые могут быть соединены в один общий маршрут.
3. Если такие маршруты найдены, то выполняем их объединение, удаляя $(0, j)$ и $(i, 0)$ а затем добавляя (i, j) .

Последовательный вариант можно представить следующим образом:

1. Для всех маршрутов $(0, i, \dots, j, 0)$ выполним следующие действия:
2. Проведём поиск элемента в списке сбережений s_{ki} или s_{jp} , который может быть использован для объединения текущего маршрута с некоторым, содержащим дугу (ребро) $(k,0)$ или $(0,l)$.
3. Если элемент был найден, то выполним объединение и применим процедуру ещё раз к новому маршруту.
4. Если сбережение найти не удалось, то перейдём к следующему маршруту и продолжим работу.
5. Завершим работу, когда объединения более невозможны.

Вычислительные результаты сравнения показывают, что параллельный вариант алгоритма даёт результаты лучше, чем последовательный.

2) Волновой алгоритм:

1. Из начальной точки распространяется волна в 4-е стороны. Выбранный элемент создает новый фронт волны. На рисунке цифрами обозначены номера фронтов волны. Каждый из элементов первого фронта волны будет, является источником вторичной волны (рисунок 1). Элементы второго фронта волны будут генерировать волну третьего фронта и т.д. Процесс формирования волн продолжается, пока не будет достигнут конечный элемент [6].

Правила для построения волны:

- волна ставится только точке со значением (-1);
- выбранная точка проверяется на наличие в ней другого погрузчика в конкретное время, если другой погрузчик был найден в этой точке, то ставится не номер волны, а значение

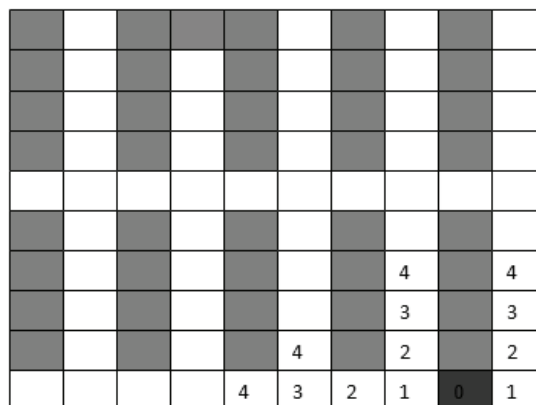


Рисунок 1. Распространение волны

	16		18		12		10		10
	15		17		11		9		9
	14		16		10		8		8
	13		15		9		7		7
13	12	13	14	-2	8	7	6	7	6
	11		15		7		5		5
	10		16		6		4		4
	9		-2		5		3		3
	8		6		4		2		2
8	7	6	5	4	3	2	1	0	1

Рисунок 2. Результаты действия

(-2), иначе ставится волна.

2. На втором этапе волнового алгоритма строится сама трасса (рисунок 2). Ее построение необходимо осуществлять в соответствии со следующим правилом, при движении от конечного элемента к начальному номер фронта волны должны уменьшаться [6].

Выберем волновой алгоритм, так как при помощи этого алгоритма находится самый короткий маршрут.

Формальная постановка задачи

Имеется n_p зон погрузки и отгрузки в которых сосредоточены поступившие товары, их наименования $N_{n_1}, \dots, N_{n_{n_p}}$ и количество A_1, \dots, A_{n_p} . И имеются $n*m*p$ (n – количество стеллажей; m – количество ячеек; p – количество ярусов) мест размещения $B_{1,1,1}, \dots, B_{n,m,p}$ для размещения товара. Также имеются два типа погрузчиков, первый тип может размещать товары на высоту $1, \dots, p_1$, а второй на высоту $1, \dots, p_2$. Необходимо составить план размещения товара на складе.

Построим схему склада (рисунок 1) MAP_{ij} , $i=1, \dots, n*2$, $j=1, \dots, m+4$:

$$MAP_{ij} = \begin{cases} -3, & \text{если на высоте } i \text{ стоит стеллаж} \\ -1 & \end{cases}, \quad (1)$$

где n – количество стеллажей, m – количество ячеек.

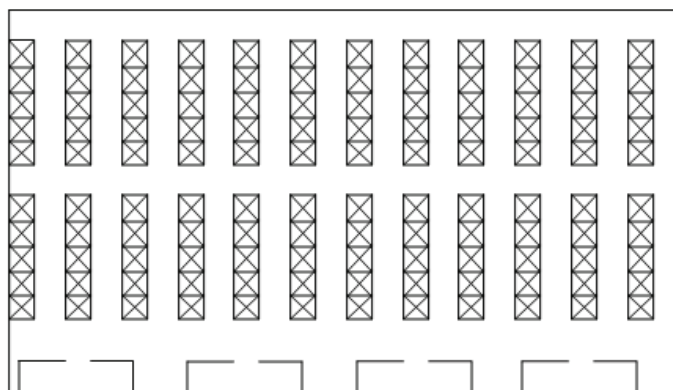


Рисунок 3. Схема склада

После применим волновой алгоритм, который заключается в распространении волны от начальной точки и заканчивается при достижении конечной точки, и получим обновленную схему склада (рисунок 4) со значениями (-3 – стоит стеллаж, -2 – перекрыто другим погрузчиком, -1 – не рассматривается, 0 – точка начала пути, >0 – проставленные волны).

Далее двигаясь от конечной точки с постепенным уменьшением волны, составим маршрут движения (рисунок 5).

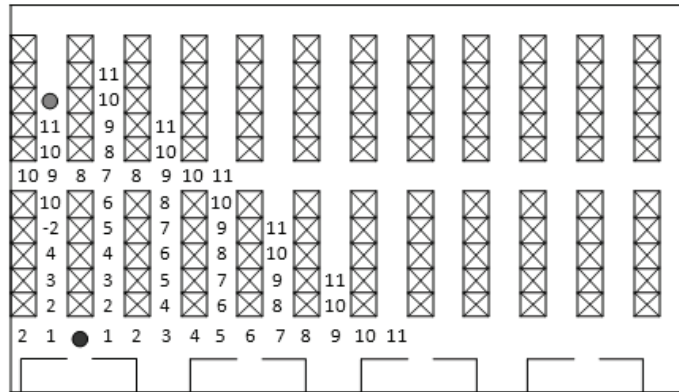


Рисунок 4. Обновленная схема склада

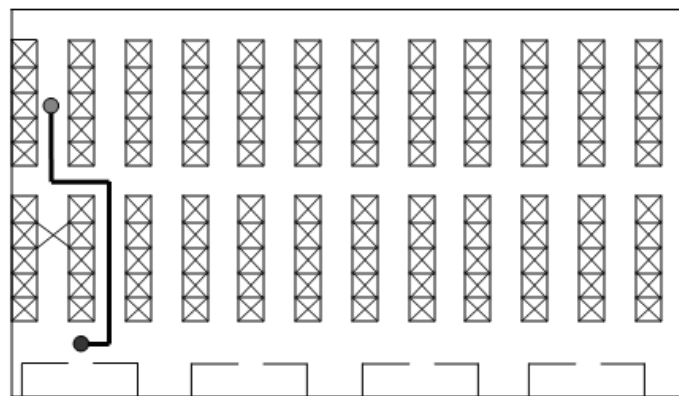


Рисунок 5. Маршрут движения

Далее двигаемся от конечной точки к следующей и так далее пока не построим полный маршрут для выполнения поставленных задач.

Пошаговое описание алгоритма:

Шаг 1. Берем погрузчик (np), который раньше всех освободится $\min(Tp[1..n])$. Выбираем зону прихода ($n1$) и берем пришедший товар ($t1$), и также выбираем зону отгрузки ($n2$) и запоминаем, какой товар ($t2$) необходимо в нее привезти.

Шаг 2. Применим волновой алгоритмом для построения маршрута движения погрузчика:

1. Найдём маршрут из начальной точки погрузчика к зоне прихода и увеличим время, формула (2)

$$Tp[np] = Tp[np] + Tdv, \tag{2}$$

где Tdv – время которое было потрачено на передвижения между двумя точками.

2. Далее находим передвижение из зоны прихода к нужной нам ячейке с минимальным расстоянием до нее, для размещения товара и увеличим время, формула (4), если такого маршрута не было найдено конечной точкой становится начальная.
3. Далее находим передвижение из ячейки, куда был размещен товар к нужной нам ячейке с минимальным расстоянием до нее, для изъятия товара и увеличим время, формула (2), если такого маршрута не было найдено конечной точкой становится начальная.
4. Далее находим передвижение из ячейки, из которой был изъят товар к зоне отгрузки и увеличим время, формула (2).

Шаг 3. По полученным результатам мы заносим в БД результаты размещения (если было размещение) и изъятия (если было изъятие) товара.

Шаг 4. Если вещь товар был размещен и изъят, то заканчиваем работу, иначе возвращаемся к шагу 1.

Заключение

Преимущества волнового алгоритма в том, что с его помощью можно найти маршрут в любом лабиринте и с любым количеством перекрытых путей (стеллажей, стоящих погрузчиков). При помощи этого алгоритма находится самый короткий маршрут. Недостаток алгоритма – требуется много памяти, для хранения карты.

Литература

- [1] Оптимизация размещения товара в распределительном. Электронный ресурс. Режим доступа: <http://www.axelot.ru/Company/publ/optimisation>.
- [2] Основы технологии складских операций на предприятиях оптовой торговли. Электронный ресурс. Режим доступа: http://otherreferats.allbest.ru/management/00119631_0.html.
- [3] Разработка системы складирования центре. Электронный ресурс. Режим доступа: <http://virgos.ru/poleznye-stati>.
- [4] Алгоритмы решения задачи маршрутизации транспорта. Электронный ресурс. Режим доступа: <http://www.marigostra.ru/materials/disser.pdf>.
- [5] Clarke G. Scheduling of vehicles from a central depot to a number of delivery points / G. Clarke, J.W. Wright / Operations Research. - 1964. - № 12 - P. 568–581.
- [6] Волновой алгоритм. Нахождение пути в лабиринте. Электронный ресурс. Режим доступа: <http://orionxl.ru/volnovojs-algoritm.html>.