

Алгоритм выбора направления движения кабины лифта

Беликов И.А. Арутюнян А.Р.
Кафедра ЭВМ, ДонНТУ

Abstract

Belikov I.A, Arutyunyan A.R. Cage moving direction choice algorithm. This article contains description of cage moving direction choice algorithm that allow to increase lift carrying capacity.

Введение

Лифт стал неотъемлемой частью искусственно созданной среды обитания человека технократической цивилизации [3]. Лифт представляет собой подъемное оборудование, обслуживающее два или более этажей, включающее кабину для транспортировки пассажиров и/или других грузов, которая движется между жесткими направляющими рельсами, расположенным вертикально или с отклонением от вертикали не более чем на 15° [2].

Лифт является одним из наиболее важных и массовых средств пассажирского транспорта в городах. Роль его непрерывно возрастает в связи с объективной тенденцией повышения этажности строительства. Одной из составных частей лифта является система управления лифтом. На настоящий момент времени система управления лифтом обеспечивает обслуживание требований пассажиров (приказов из кабины или вызовов с этажных постов) решая при этом ряд логических задач, связанных прежде всего с правильным выбором направления движения в зависимости от особенностей работы лифта в различных режимах [1]. При этом, выбор режима представляет собой выбор набора правил, которыми руководствуется система управления лифта при выборе направления движения. Несмотря на ряд преимуществ, такой подход обладает существенным недостатком: следование одному из существующих наборов правил не гарантирует того, что совокупность выбранных направлений будет являться оптимальной или близкой к оптимальной.

Общая методика построения алгоритма управления лифтом, согласно которой предлагается реализация в виде конечного автомата, кратко изложена в работе [5]. Более подробное описание данного подхода приводится в работе [6], реализацию данного подхода с использованием объектно-ориентированного проектирования можно найти в работе [8].

Качество функционирования и эффективность функционирования жилых, административных и общественных зданий и сооружений в значительной мере определяется степенью рациональности решения задач

перевозок грузов и пассажиров на основе применения лифтов [4]. Следовательно, актуальной задачей является разработка алгоритма, который позволит гарантировать, что время удовлетворения требований не будет сильно отличаться от минимально возможного.

Построение алгоритма

В настоящее время существуют следующие способы управления кабиной лифта:

1. Простое управление — управление, при котором регистрируется и выполняется последующая команда управления только после выполнения предыдущей команды;
2. Собирательное управление — смешанное управление, при котором после регистрации одной команды управления могут быть зарегистрированы и последующие; при этом выполнение команд управления происходит в соответствии с заданной программой;
3. Одностороннее собирательное управление по вызовам — собирательное управление по командам управления с посадочных площадок, при котором предусматривается выполнение попутных вызовов только при движении кабины в одном направлении: вверх или вниз;
4. Двустороннее собирательное управление по вызовам — собирательное управление по командам управления с посадочных площадок, при котором предусматривается выполнение попутных вызовов при движении кабины вверх и вниз.

Необходимо отметить, что набор кнопок для управления лифтом не может обеспечить систему управления данными, которых было бы достаточно для решения поставленной задачи. Действительно, для оптимального выбора направления недостаточно информации о требуемом направлении движения, т.к. она не содержит в себе количество этажей, на которое необходимо переместить пассажира, т.е. оценить количество ресурса лифта, необходимого для удовлетворения запроса. Эта проблема может быть решена несколькими способами. Один из них состоит в том, чтобы с помощью видеодатчика определять, кто вошёл, и использует настраиваемую сеть, чтобы выбрать этаж, на котором данный человек обычно выходит [7]. Второй — размещение на каждом этаже не одной или двух кнопок, а пульта управления, который содержит кнопки с номерами всех этажей, что позволит указывать пассажиру этаж назначения еще до того как он войдет в кабину. Кроме того, предлагается (данное требование, в отличие от предыдущего, не является необходимым для решения поставленной задачи) рассматривать лифт не как систему, которая доставляет человека от исходного этажа к конечному, а как транспортное средство, которое движется по определенному маршруту, позволяя

пассажирам заходить в него, если ему по пути. При этом маршрут должен рассчитываться таким образом, чтобы последний из пассажиров был доставлен как можно раньше.

Исходными данными для рассматриваемого алгоритма, является совокупность запросов. Каждый запрос представляет собой пару чисел v_1 и v_2 . v_1 – номер этажа, на котором находится пассажир в текущий момент, v_2 – номер этажа, на который необходимо доставить пассажира.

Совокупность запросов можно представить в виде двух взвешенных ориентированных графов G_1 и G_2 . При этом каждая вершина v_i графов G_1 и G_2 соответствует i -тому этажу, ребра G_1 – запросам, для которых выполняется неравенство $v_1 > v_2$, а ребра G_2 – запросам, для которых выполняется неравенство $v_1 < v_2$. Веса ребер $(v_i; v_j)$ графов G_1 или G_2 принимаются равными количеству запросов $v_i; v_j$, т.е. запросов с одного и того же этажа на другой.

Для достижения большей наглядности и простоты изложения целесообразно воспользоваться понятием разреза графа. Разрезом $(S; V - S)$ графа $G = (V; E)$ называется разбиение его вершин на два множества [9]. Рассмотрим разрезы $C_i = (V_i; V - V_i)$ графа $G_1 = (V; E)$, где $V_i = \{v_j : j \leq i\}$. Пусть суммарный вес ребер, пересекающих разрез C_i равен W_i , а максимальная вместимость кабины лифта M , тогда $K_i^1 = \left\lceil \frac{W_i}{M} \right\rceil$ – минимальное число раз, которое кабина должна пройти по направлению вниз между этажом i и $i+1$, при условии, что граф является связным.

Аналогично можно получить минимальное число раз, которое кабина должна пройти между этажом i и $i+1$ в направлении вверх K_i^2 .

В случае, когда граф является несвязным, исходя из того, что кабина должна пройти по маршруту каждого из запросов минимум один раз, то она должна пройти от самого нижнего этажа, с которого или на который выполнен запрос, до самого верхнего этажа с или на который сделан запрос минимум один раз. Таким образом, кабина также должна пройти минимум один раз между каждой парой этажей i и $i+1$, для которых получено $K_i^{1,2} = 0$.

Вычисляя, таким образом, K_i^1 и K_i^2 для i от 1 до $N-1$, где N – число этажей, можно получить число раз, которое кабина должна пройти между каждой парой соседних этажей. При этом, исходя из того, что кабина должна вернуться в исходное положение, число раз, которое кабина будет проходить вверх, должно быть равно числу раз, которое кабина пройдет вниз. Поэтому минимальное число раз K_i , которое кабина должна пройти между двумя соседними этажами вверх/вниз, равно максимальному из двух значений K_i^1 и K_i^2 .

По вычисленным значениям K_i можно сформировать маршрут движения кабины, используя метод «выметания». При построении маршрута используются две последовательности K_i , первая анализируется и модифицируется при движении вниз, а вторая – при движении вверх. Построение маршрута выполняется следующим образом:

- При движении выделяются следующие события:
 - Событие 1 на текущем участке последовательности $K_i < K_{i+1}$.
 - Событие 2 на текущем участке последовательности $K_i > K_{i+1}$ или достигнут конец последовательности.
- Выметание производится, начиная с K_1 , до тех пор, пока не пуст стек, в который изначально добавляется позиция номер 1 (первый этаж), при направлении движения вверх.
- При прохождении через элемент последовательности, его значение уменьшается на единицу.
- При возникновении события 1 в стек добавляется номер текущей позиции, и текущее направление движения
- При возникновении события 2 необходимо извлечь из стека номер позиции, вернуться на нее, после чего продолжить движение в сохраненном направлении, если не произошло одно из перечисленных выше событий
- При возвращении в позицию 1 необходимо повторно выполнять перечисленные действия, до тех пор, пока все запросы не будут удовлетворены.

При выполнении указанных действий поддерживается следующий инвариант:

Инвариант 1. Последовательности K содержат только одну подпоследовательность, ограниченную нулями или концом последовательности, состоящую из ненулевых элементов.

Инициализация. Справедливость инварианта следует из метода построения последовательностей K .

Сохранение. Если в стеке сохранен хотя бы один номер элемента, то исходя из признака, по которому в стек добавлен этот номер, значение текущего уменьшаемого элемента больше одного, и, следовательно, он не станет нулевым. Если же стек пуст, значение текущего элемента последовательности будет равно значению первого элемента последовательности в начале выполнения алгоритма. Если значения первого элемента в начале выполнения алгоритма было равно одному, то все элементы до текущего станут нулевыми, в противном случае они будут больше нуля.

Инвариант 2. При движении по полученному маршруту кабина лифта всегда будет максимально заполненной.

Согласно методу построения последовательности K_i выбирается как максимум из K_i^1 и K_i^2 . При этом можно добавить фиктивные запросы между соседними этажами так, что бы минимальное значение из K_i^1 и K_i^2 увеличилось до K_i . Т.о. K_i можно условно считать количеством запросов, попавших в соответствующий разрез.

Инициализация. Т.к. начальное положение кабины – первый этаж, то K_1 соответствует числу запросов сделанных с первого этажа. Т.к. K_1 больше нуля, кабина может быть полностью заполнена.

Сохранение. Событие 1 свидетельствует о наличии запросов из текущей вершины, которые позволят загрузить кабину после возврата.

Событие 2 свидетельствует о наличии запросов в направлении вниз, которые позволяют загрузить кабину.

В случае, когда не происходит ни одно из перечисленных событий, смежные элементы последовательности равны, что, в свою очередь, свидетельствует о том, что каждому из запросов, направленных на данный этаж, соответствует запрос, направленный с данного этажа, что позволяет сохранять неизменной загруженность кабины.

Инвариант 2 позволяет утверждать, что каждое уменьшение значений K_i соответствует удалению запроса из соответствующего разреза. Таким образом, значения элементов последовательности после модификации соответствуют количествам запросов, оставшихся в соответствующих разрезах. При выполнении алгоритма значения элементов уменьшаются, при этом согласно инварианту 1, ненулевые элементы не будут разорваны на несколько частей подстроками, состоящими из нулей, что в свою очередь, позволяет уменьшить все значения элементов K_i до нулевого значения.

Результаты моделирования

Результаты моделирования систем управления, действующих по набору правил «Нормальная работа» и разработанному алгоритму, следующие:

- При генерации случайных запросов с равномерным распределением вероятности появления запросов с каждого этажа, предложенный алгоритм уступает классическому алгоритму. На рисунке 1 изображена зависимость суммарного числа этажей, на которое необходимо переместить всех пассажиров, от времени, при использовании предложенного алгоритма. Результаты применения стандартного алгоритма в этом случае приведены на рисунке 2.
- При генерации тестов с фиксированной вероятностью появления одного из трех видов запросов отмечается превосходство предложенного алгоритма. На рисунке 3 изображена зависимость

суммарного числа этажей, на которое необходимо переместить всех пассажиров, от времени, при использовании предложенного алгоритма. Результаты применения стандартного алгоритма в этом случае приведены на рисунке 4.

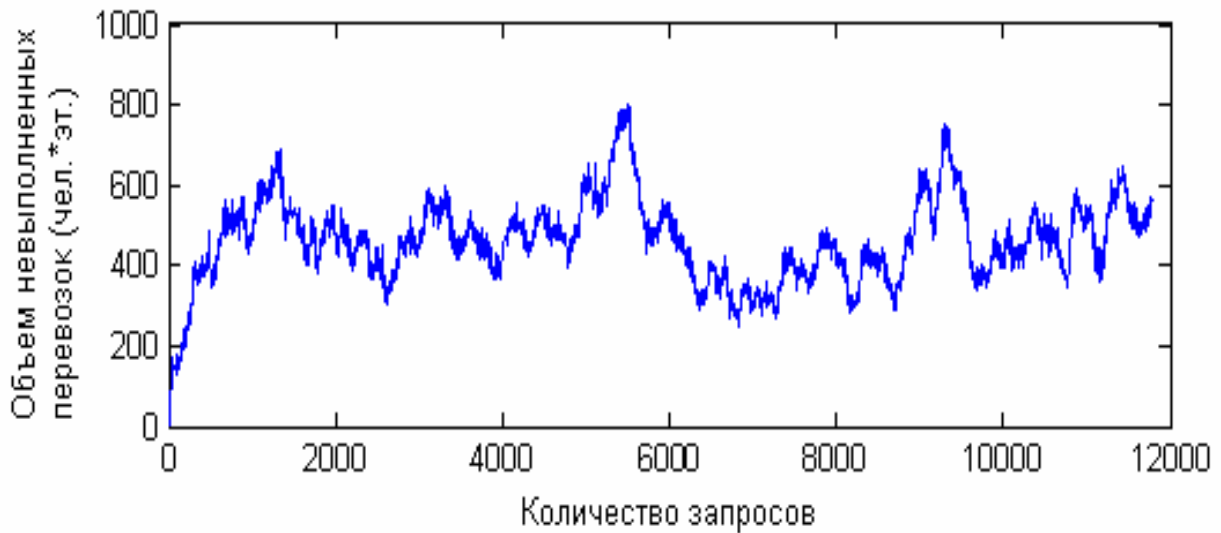


Рисунок 1 - Объем невыполненных перевозок при случайном наборе запросов, полученный при использовании предложенного алгоритма

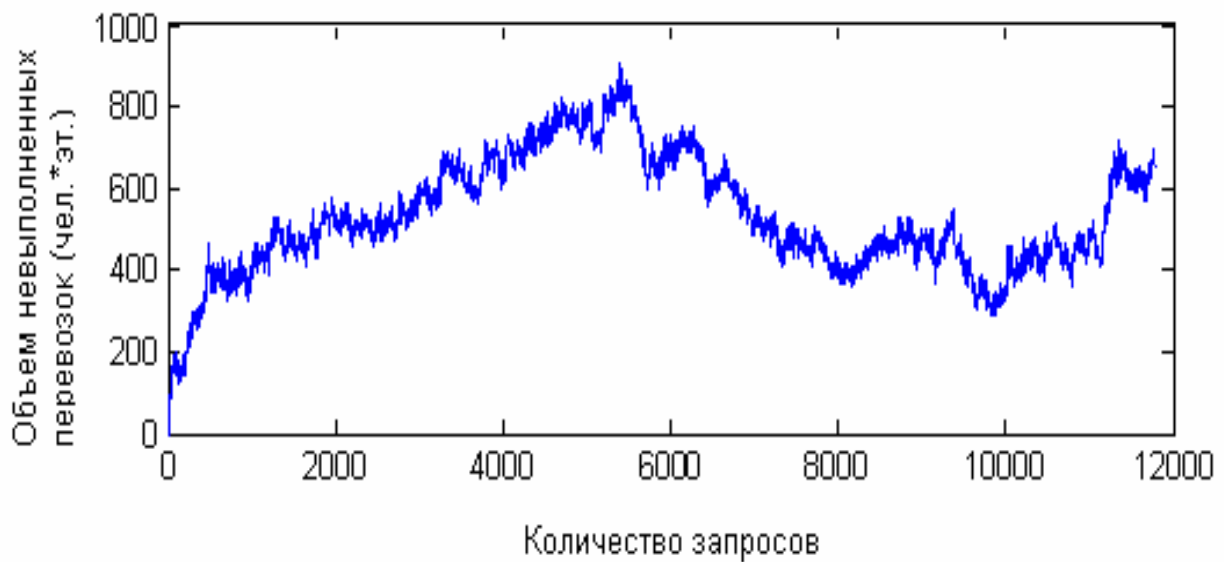


Рисунок 2 - Объем невыполненных перевозок при случайном наборе запросов, полученный при использовании стандартного алгоритма

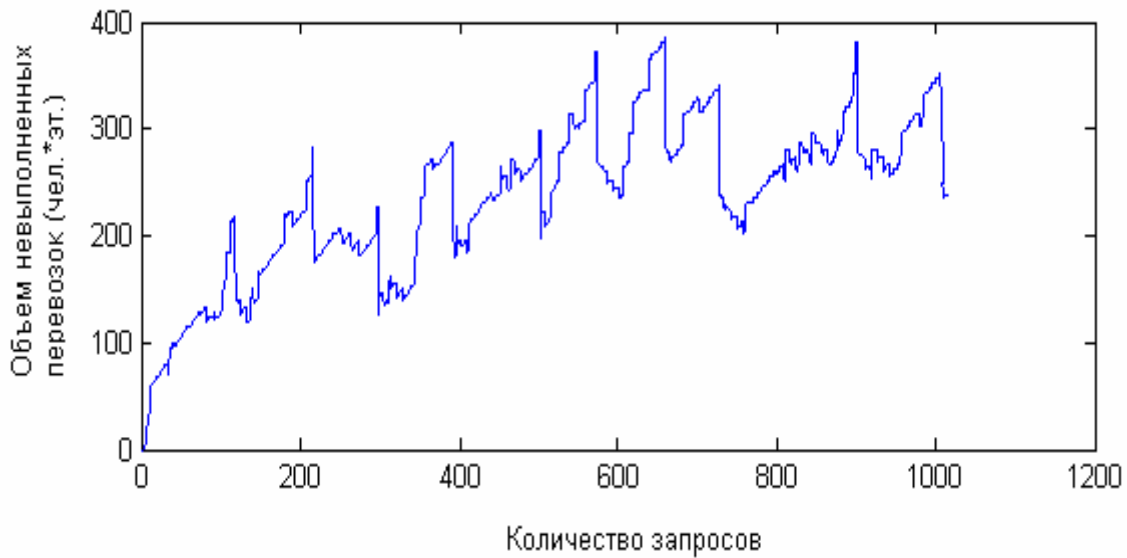


Рисунок 3 - Объем невыполненных перевозок при фиксированном наборе запросов, полученный при использовании предложенного алгоритма

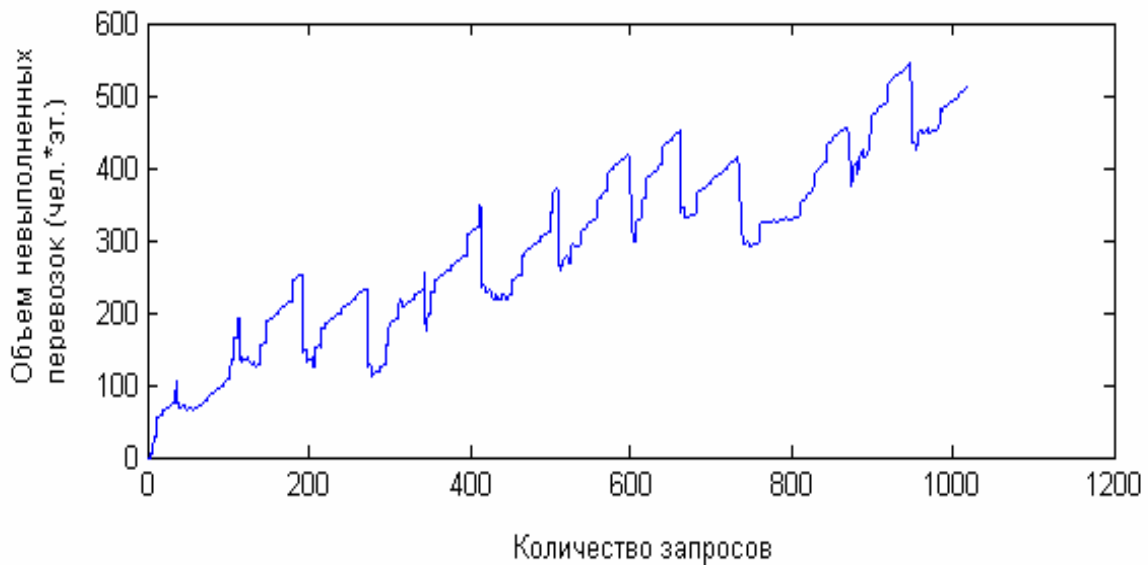


Рисунок 4 - Объем невыполненных перевозок при фиксированном наборе запросов, полученный при использовании стандартного алгоритма

Выводы

Из приведенных выше результатов видно, что при наличии случайного набора запросов, эффективности стандартного и предложенного алгоритмов примерно одинаковы. При этом амплитуда колебаний объема невыполненных перевозок, при использовании стандартного алгоритма, больше, чем при использовании предложенного

алгоритма, что можно объяснить тем, что время от времени появляются наборы запросов, с которыми стандартный алгоритм справляется более или менее эффективно, чем предложенный алгоритм. При фиксированном наборе запросов (именно тех, с которыми стандартный алгоритм справляется менее эффективно), стандартный алгоритм перестает обеспечивать удовлетворение всех запросов.

Литература

1. Лифты. Учебник для вузов /под общей ред. Д.П.Волкова - М.: изд-во АСВ, 1999. - 480 стр. с илл.
2. Яновски Л. Проектирование механического оборудования лифтов. Третье издание: -М.: Монография. Издательство АСВ, 2005, -336 с
3. Гидравлические лифты. Учебное пособие под общей редакцией Г. Г. Архангельского. -М.: изд-во АСВ, 2002 - 346 стр. с иллюстрациями.
4. Архангельский Г.Г. Ионов А.А. Основы расчета и проектирования лифтов. Основы расчета и проектирования лифтов. Учебное пособие. –М. : МИСИ, 1985, 74с.
5. Кнут Д. Искусство программирования. Т. 1: Основные алгоритмы. М.: Вильямс, 2001. – 712 с.
6. Шалыто А.А. Switch-технология. Алгоритмизация и программирование задач логического управления. СПб.: Наука, 1998. – 628 с.
7. Ю. Ф. Голубев. Нейронные сети в мехатронике. //Фундаментальная и прикладная математика, том 11(2005), № 8, стр 81-103
8. Л. А. Наумов, А. А. Шалыто, Искусство программирования лифта. Объектно-ориентированное программирование с явным выделением состояний // «Информационно-управляющие системы». 2003. №6, с. 38–49.
9. Т. Кормен Ч.Лейзейрсон, Р Ривест. Алгоритмы: построение и анализ. М: МЦНМО, 2000. –960с, 263 ил.

Дата поступления в редакцию 10.10.2007