



OSTIS-2012

(Open Semantic Technologies for Intelligent Systems)

УДК 621.876.11 + 519.872.4

АЛГОРИТМЫ УПРАВЛЕНИЯ ЛИФТАМИ С ИСПОЛЬЗОВАНИЕМ СЕМАНТИЧЕСКОЙ ИНФОРМАЦИИ

Кузнецов А.П., Никонов В.Н., Шмарловский А.С., Силивонец М.В.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

kuznap@bsuir.by

nrkn@mail.ru

sas@bsuir.by

a702@tut.by

Рассмотрены наиболее распространенные алгоритмы управления группами лифтов. Анализ эффективности алгоритмов осуществлен с использованием специально разработанной программной платформы. Описаны основные факторы, влияющие на пассажиропоток. Разработан универсальный алгоритм диспетчерского управления группой лифтов, эффективность которого подтверждена при помощи моделирования.

Ключевые слова: алгоритмы управления, лифты, управление группой лифтов, диспетчерское управление.

ВВЕДЕНИЕ

Быстрый рост городов, увеличение ценности земли являются причиной устойчивой тенденции роста этажности городской застройки. По мере увеличения этажности зданий возрастают требования к надежности, комфортности, качеству изготовления, монтажа и эксплуатации лифтов, их энергоэффективности. В крупных зданиях с интенсивным пассажиропотоком эффективность обслуживания пассажиров системой лифтов имеет большое значение. Минимизация времени ожидания лифта после вызова и времени доставки пассажиров на требуемый этаж является актуальной задачей. Несмотря на то, что лифты уже давно стали обычным явлением в повседневной жизни, задача распределения кабин лифта по вызовам таким образом, чтобы время ожидания и поездки было минимальным, далека от тривиальной.

Любую систему, в которой поток требований встречает ограниченные средства их удовлетворения, можно рассматривать как систему массового обслуживания (СМО). Если моменты поступления требований или продолжительность их обслуживания не регламентируются, то при пользовании системой возникают конфликты и образуется очередь. Длина этой очереди зависит от двух характеристик потока требований: интенсивности поступления требований и статистических флуктуаций этой интенсивности [Клейнрок, 1979]. Если интенсивность поступления

требований превышает пропускную способность системы, система не справляется с потоком требований и начинает расти очередь неограниченной длины. Но даже если интенсивность поступления требований меньше пропускной способности системы, очередь может образовываться из-за статистических флуктуаций и внезапного накопления требований. Влияние таких колебаний увеличивается, если средняя нагрузка приближается к пропускной способности системы.

В последнее время получили широкое распространение микропроцессорные системы управления лифтами, позволяющие реализовать достаточно сложные алгоритмы функционирования лифта или группы лифтов, позволяющие минимизировать время ожидания и добиться максимальной производительности с учетом направленности и напряженности пассажиропотока в здании. Существует большое количество типовых алгоритмов диспетчерского управления. Каждый из них имеет свои сильные и слабые стороны, проявляющиеся в различных условиях при использовании в зданиях с различной интенсивностью пассажиропотока. Целью работы является разработка универсального алгоритма, сочетающего в себе наиболее сильные стороны стандартных алгоритмов и обладающего интеллектуальной составляющей, позволяющей оптимизировать работу группы лифтов в контексте текущих условий. При этом алгоритм не должен быть требовательным к вычислительным ресурсам. Оптимальное использование ресурсов группы

лифтов конкретного здания должно достигаться настройкой диспетчерских алгоритмов за счет учета семантической информации. Улучшение качества работы должно обеспечиваться за счет перераспределения назначенных одиночным лифтам вызовов в зависимости от изменения обстановки и текущего характера пассажиропотока. Характер пассажиропотока заранее неизвестен, однако его можно спрогнозировать за счет учета изменений месторасположения некоторых объектов в здании, текущего времени (времена обедов, будний или праздничный день, день недели, сезон года и т. п.), погоды, накопленной статистики и т. д. Анализ этой информации не является тривиальной задачей.

1. Система управления группой лифтов

Лифты представляют собой системы массового обслуживания, в которых алгоритмы диспетчерского управления используются для распределения кабин по вызовам, полученным от пассажиров. Здание определяется как система из N этажей и L кабин лифта. Кабины с индексом $\{1, 2, \dots, L\}$ либо стоят на этажах $\{1, 2, \dots, N\}$, либо движутся вверх или вниз. Вызовы с этажа инициируются пассажирами, желаемое направление движения определяется использованием соответствующей кнопки. Вызовы из кабины инициируются пассажирами, которые вошли в лифт и нажимают кнопку этажа внутри кабины. Вызванный этаж соответствует этажу, на котором пассажир хочет выйти из лифта. Вызовы с этажа регистрируются контроллером, управляющим работой группы лифтов. Контроллер назначает кабину для обслуживания данного вызова на основе диспетчерского алгоритма. Назначение может быть задержано до тех пор, пока не появится лифт с достаточными возможностями для обслуживания вызова. Вызовы из кабины немедленно попадают на контроллер лифта, в задачу которого входит сортировка вызовов, назначенных групповым контроллером, и формирование внутренней последовательности обслуживания. Если назначено обслуживание хотя бы одного вызова, лифт движется в направлении целевого этажа, то есть первого вызова в последовательности обслуживания. Обобщенная структурная схема данной системы приведена на рис. 1.

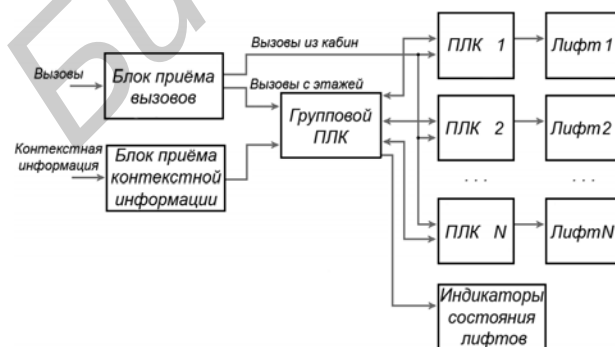


Рисунок 1 – Обобщенная структурная схема системы управления группой лифтов

Диспетчерское управление лифтами осуществляется на двух уровнях: контроллер группы лифтов и контроллер лифта. Контроллер группы лифтов отвечает за работу всех групп лифтов и осуществляет глобальное планирование. Как только вызов назначен для обслуживания определенным лифтом, соответствующий контроллер лифта выполняет локальное планирование в рамках одного лифта. Диспетчерское управление на уровне контроллера лифта обычно осуществляется классическим способом обслуживания в одном направлении: лифт отвечает на все вызовы в текущем направлении своего движения до тех пор, пока последний этаж по вызовам не окажется этажом изменения направления движения. После этого лифт начинает идти в противоположном направлении, обслуживая все вызовы по ходу нового направления движения до тех пор, пока вновь не будет достигнут этаж изменения направления движения.

2. Программный комплекс

Для определения значений параметров эффективности диспетчерских алгоритмов проведено моделирование системы управления группой лифтов. Проведение эксперимента с реально существующей группой лифтов требует больших затрат, поэтому была разработана библиотека классов C++, моделирующая группу лифтов, и создана компьютерная программа на ее основе. Интерфейс программы (рис. 2) предусматривает возможность ввода такой информации как количество лифтов и этажей в здании, скорость лифта, время нахождения лифта на этаже с открытыми дверями, время открывания дверей, направление и интенсивность пассажиропотока в рассматриваемый период, максимально возможное количество людей в здании.

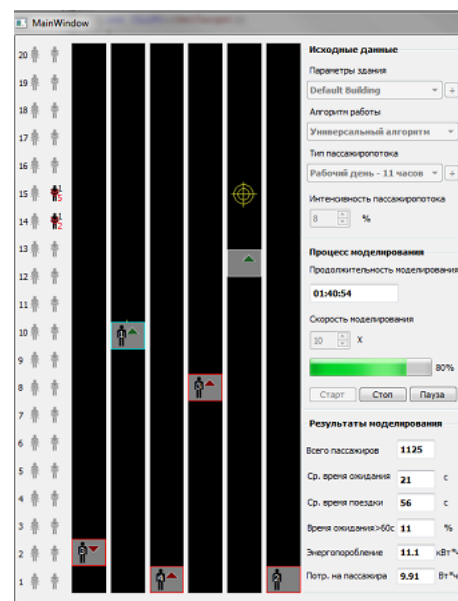


Рисунок 2 – Интерфейс программы

Скорость прибытия пассажиров моделируется

как случайная величина с распределением Пуассона. Программа производит моделирование работы системы управления исходя из перечисленных входных параметров и генерируемых потоков пассажиров. Результатом работы программы является расчет следующих параметров системы:

- общее количество пассажиров, воспользовавшихся лифтом;
- среднее время ожидания кабины лифта;
- среднее время нахождения в лифте;
- процент пассажиров, которые не были обслужены по истечении одной минуты;
- энергопотребление, в том числе в расчете на одного пассажира.

Графический пользовательский интерфейс позволяет строить графики мгновенного и усредненного энергопотребления, усредненного времени ожидания и доли пассажиров, не обслуженных в течение минуты.

3. Обзор наиболее распространенных алгоритмов диспетчерского управления

Ниже приведены алгоритмы, получившие наибольшее распространение в системах диспетчерского управления на уровне контроллера группы лифтов.

Преимуществами кругового алгоритма является простота реализации, равномерная загрузка каждого лифта, а также обеспечение приемлемого обслуживания пассажиров в условиях неинтенсивной загрузки. Основной целью круговой системы при диспетчерском управлении группой лифтов является достижение равной нагрузки на каждый лифт. Вызовы распределяются по мере их поступления последовательным образом по отдельным лифтам. Вызов 0 назначается к обслуживанию кабиной 0, вызов 1 – кабиной 1, вызов L – кабиной 0, вызов $L+1$ – кабиной 1 и т. д.

Частным случаем кругового алгоритма является диспетчерское управление при максимальном потоке вверх. Отличием данного варианта является использование особой стратегии парковки: если групповой контроллер обнаруживает простаивающий лифт, он инициирует вызов с первого этажа для отправки кабины на этаж с предположительно максимальным потоком пассажиров, для уменьшения времени ожидания будущих пассажиров.

В некоторых зданиях используют идею зонирования, заключающуюся в разделении здания на несколько прилегающих друг к другу зон, причем каждый из лифтов обслуживает вызовы с этажей только той зоны, которая назначена для обслуживания данным лифтом.

Алгоритм трех переходов [Ронг и др, 2003] представляет собой частный случай алгоритма управления группой лифтов с оценкой времени

прибытия. Он используется для определения последовательности обслуживания вызовов с этажей. Вызовы прохода 1 (P1) лифт может обслужить по пути следования в настоящий момент времени. Вызовы прохода 2 (P2) лифт может обслужить после того, как один раз изменит направление движения. Вызовы прохода 3 (P3) лифт может обслужить после того, как дважды изменит направление движения. Вызовы группы P1 сортируются от ближайшего к дальнему относительно текущего положения лифта и выполняются в этом порядке. Если нет вызовов группы P1, выполняются вызовы группы P2 от дальнего к ближнему. При отсутствии вызовов первых двух групп выполняются вызовы группы P3 в порядке от дальнего к ближнему.

4. Итерационный алгоритм

Одним из резервов улучшения качества работы является перераспределение уже назначенных одиночным лифтам вызовов в зависимости от изменения обстановки. Все множество M запросов, поступающих с различных этажей здания, разбивается на такие непересекающиеся подмножества M_i ($i = 1 \dots n$, где n – число лифтов в группе), чтобы максимальное время обработки i -м лифтом подмножества запросов M_i с учетом уже имеющегося внутреннего задания было минимальным:

$$T = \max \{T_i; i = 1, 2, \dots, n\}, \quad (1)$$

где T – общее время обслуживания всех запросов;

T_i – время обслуживания i -м лифтом подмножества запросов M_i .

Поставленную задачу можно отнести к классу комбинаторных задач, для решения которых существует ряд методов. Однако все эти методы так или иначе требуют полного перебора возможных вариантов, число которых составляет

$$C = (2^n - 1)^m, \quad (2)$$

где $m = |M|$ – число запросов.

Максимальное значение $m_{\max} = 2K - 1$, где K – число этажей в здании. Поэтому:

$$C = (2^n - 1)^{2K-1}. \quad (3)$$

К примеру, если число этажей в здании 20, а число лифтов 8, получаем:

$$C = (2^8 - 1)^{39} \approx 2^{312}. \quad (4)$$

Очевидно, что для большинства контроллеров такой полный перебор в реальном времени изменения ситуации (изменения множества запросов, внутренних заданий и положения лифтов) невозможен. Рассмотрим другой подход к решению проблемы комбинаторного целераспределения,

основывающийся на стратегии распределенного или коллективного принятия решений [Каляев, 2001]. В подходе используется итерационный алгоритм оптимизации, суть которого заключается в последовательном выборе каждым объектом, входящим в группу, такой цели, которая дает экстремальное приращение целевого функционала при фиксированном выборе всех остальных объектов группы. Итерационный процесс оптимизации продолжается до тех пор, пока в двух последовательных циклах итерации значение целевого функционала не изменяется. Для учета происходящих изменений ситуации процедура периодически повторяется заново, что позволяет оперативно оптимизировать распределение.

Описанная выше процедура является асимптотически сходящейся в смысле определения Ляпунова, причем число итерационных циклов D , за которые данная процедура сходится

$$D \leq \frac{|T_{i\dot{a}}^{\max} - T_{i\dot{a}}^{\min}|}{\Delta T_{\min}}, \quad (5)$$

где $T_{o\dot{o}}^{\max}$ – максимально возможное значение времени обслуживания;

$T_{o\dot{o}}^{\min}$ – минимально возможное значение времени обслуживания;

ΔT_{\min} – минимально возможное изменение времени обслуживания при перераспределении запросов между лифтами.

Значения $T_{o\dot{o}}^{\max}$, $T_{o\dot{o}}^{\min}$ и ΔT_{\min} можно оценить, исходя из следующих соображений:

1. Максимально возможное время обслуживания может быть в том случае, если все запросы последовательно будет обслуживать единственный лифт, причем в наихудшем случае общее число обслуживаемых запросов будет $2K$, где K – число этажей в здании. Поэтому:

$$T_{o\dot{o}}^{\max} \leq 2\tau(K + k \cdot K) = 2\tau K(1 + k). \quad (6)$$

2. Минимально возможное время обслуживания будет достигаться в случае, если каждый из лифтов выполняет единственный запрос, перемещаясь при этом на один этаж:

$$T_{o\dot{o}}^{\min} \geq \tau(1 + k). \quad (7)$$

3. Минимально возможное изменение времени обслуживания при перераспределении запросов достигается в случае, если один лифт «забирает» у другого один запрос. При этом общее время обслуживания в минимальном случае будет уменьшаться на время посадки-высадки пассажиров $k\tau$ или время, в течение которого лифт проезжает один этаж τ , в зависимости от того, какая из этих величин будет меньше для конкретного здания:

$$\Delta T_{\min} \geq \min\{k\tau, \tau\}. \quad (8)$$

Отсюда получаем

$$D \leq \frac{(2\tau K(1 + k) - \tau(1 + k))}{\min\{k\tau, \tau\}} = \frac{(1 + k)(2K - 1)}{\min\{k, 1\}}. \quad (9)$$

Описанная выше процедура оптимизации распределения запросов между лифтами обладает рядом преимуществ по сравнению со стандартными процедурами, основывающимися на полном переборе. Значительно снизится число анализируемых вариантов распределения запросов:

$$C = D \cdot n \cdot m_{ig}^j \leq D \cdot n \cdot K, \quad (10)$$

где $m_{ig}^j = |M_{ig}^j|$ – число запросов, включенных в множество допустимых запросов для i -го лифта на j -ом шаге итерации.

Подставляя (9) в (10), получаем

$$C \leq \frac{(k + 1)(2K - 1)}{\min\{k, 1\}} \cdot n \cdot K. \quad (11)$$

Если, например, $K=20$, $n=8$, $k=1$, получаем: $C \leq 24960$, то есть число анализируемых вариантов по сравнению со случаем полного перебора снижается приблизительно в $3,3 \cdot 10^{89}$ раз. Это позволяет реализовать описанную процедуру в реальном времени изменения ситуации с помощью большинства распространенных вычислительных устройств. Разумеется, такое существенное сокращение анализируемого числа вариантов достигается за счет отказа от достижения глобального оптимума целевой функции (достигается некоторый локальный оптимум). Однако следует отметить, что из-за быстрого и непредсказуемого изменения ситуации (например, появление новых вызовов) и неизвестного заранее точного времени посадки пассажиров по уже имеющимся вызовам, глобальный оптимум теряет свою актуальность. В таких условиях имеет смысл как можно чаще искать некий «рациональный» оптимум, что и обеспечивает предложенный итерационный алгоритм.

В программе моделирования реализован модуль сбора статистики, учитывающий информацию о каждом пассажире. Групповой контроллер реальной лифтовой системы такой информации иметь не может, однако ему доступна информация о времени регистрации и выполнения вызовов. Анализ этой информации может с определенной долей достоверности помочь определить характер пассажиропотока. Качественные критерии работы итерационного алгоритма определяются выбором критерия передачи вызова от более загруженного лифта к менее загруженному. Рассмотрены и промоделированы следующие варианты:

- Оптимизация общего времени выполнения вызовов. Условием перестановки в данном случае является возможность минимизации общего времени выполнения активных вызовов. То есть из всех вариантов перестановки двух вызовов выбирается тот, который обеспечивает

максимальное уменьшение общего времени выполнения текущих вызовов. Перестановка осуществляется для всех лифтов по очереди.

- Оптимизация общего времени выполнения вызовов с сортировкой. В отличие от предыдущего варианта для перестановки на каждом шаге выполняется сортировка лифтов по времени выполнения назначенных им вызовов. Для перестановки выбираются вызовы, назначенные самому загруженному лифту.

- Оптимизация конкретного вызова по времени выполнения. У лифта с наибольшей загрузкой выбирается вариант перестановки какого-либо вызова по критерию максимального уменьшения времени выполнения вызова, то есть из всех вариантов перестановки выбирается пара лифтов, которая дает максимальное уменьшение времени выполнения вызова. Данная оптимизация рассмотрена как с контролем общего времени выполнения вызовов (в качестве условия перестановки используется условие, запрещающее в результате перестановки увеличивать общее время выполнения вызовов), так и без него.

- Оптимизация конкретного вызова по пройденному расстоянию. К описанному выше добавляется требование перестановки вызовов только в случае уменьшения пути, который должна пройти кабина лифта для выполнения вызова.

Моделирование показало, что оптимизация конкретного вызова по времени выполнения значительно превосходит оптимизацию по общему времени вызовов. Это объясняется тем, что оптимизация общего времени вызовов в обстановке быстрого притока новых пассажиров, а следовательно, быстрого изменения общей картины вызовов не способна обеспечить приемлемого результата. Ситуация меняется столь быстро, что долгосрочное прогнозирование теряет смысл. При этом краткосрочный прогноз при условии оптимизации каждого конкретного вызова дает положительный результат. Оптимизация вызова по пройденному расстоянию показывает худшее быстрое действие, но меньшие энергозатраты по сравнению с оптимизацией вызова по времени выполнения. В дальнейшем эти алгоритмы упоминаются как итерационный алгоритм с оптимизацией скорости и итерационный алгоритм с оптимизацией экономичности соответственно. Алгоритм, обеспечивающий экономичность, получается из алгоритма оптимизации скорости с помощью дополнительного условия перестановки вызовов в итерации. Вариант с коэффициентом, регулирующим приоритет производительности или экономичности, реализуется за счет переключения алгоритмов с заданной периодичностью.

Для определения критериев оценки алгоритмов использовался типичный для офисного здания 11-часовой прогон, сочетающий в себе интенсивный поток вверх в начале прогона, интенсивный поток вниз в конце, постоянный межэтажный поток невысокой интенсивности и суперпозицию потоков

вверх и вниз высокой интенсивности в середине прогона (табл. 1). В рассматриваемом здании 12 этажей, 4 лифта, высота этажа 4 метра, пиковая интенсивность прибытия пассажиров 8 человек в минуту.

Таблица 1 – Результаты моделирования

| Алгоритм | $T_{ож},$ с | $T_{п},$ с | $\%_{>мин}$ | $P_{на}$ пасс., Вт |
|-------------------------------------|----------------|---------------|-------------|--------------------------|
| Кольцевой алгоритм | 48 | 46 | 17 | 12,1 |
| Алгоритм трех переходов | 24 | 46 | 10 | 12,4 |
| Максимальный поток вверх | 23 | 45 | 13 | 16,1 |
| Алгоритм зонирования | 33 | 49 | 20 | 11,7 |
| Итерационный алгоритм | 12 | 46 | 2 | 11,3 |
| Итерационный алгоритм (экономичный) | 15 | 47 | 5 | 10,4 |

Результаты моделирования демонстрируют преимущество итерационного экономичного алгоритма над ближайшим по результатам на 11% по экономичности и на 9% по производительности над наиболее быстрым из стандартных – алгоритмом трех переходов. Итерационный алгоритм, адаптированный по скорости, имеет преимущество на 15% по производительности над алгоритмом трех переходов и на 3,4% по экономичности над алгоритмом зонирования. Итерационный алгоритм с оптимизацией по экономичности на 6,5% менее производителен, однако на 8% экономичнее алгоритма, оптимизированного по скорости.

5. Использование семантической информации

Важным преимуществом итерационного алгоритма является возможность значительно влиять на его качественные показатели за счет изменения или расширения условий перестановки вызова от более загруженного лифта к менее загруженному. Это позволяет гибко настроить алгоритм в зависимости от характера пассажиропотока и учесть закономерности его изменения. Отдельные здания могут иметь этажи с повышенной интенсивностью пассажиропотока: например, первый этаж; этаж, на котором располагаются объекты общественного питания – во время обеденного перерыва; этажи, на которых имеются переходы, соединяющие несколько смежных зданий; открытые либо закрытые смотровые площадки высотных зданий и т. д. Отдельные этажи имеют постоянные либо временные (зависящие от времени суток или погодно-климатических условий – как, например, посещаемость открытой смотровой площадки и т. п.) всплески интенсивности заданий.

При наличии информации о повышенной интенсивности вызовов с таких этажей не

составляет труда повысить приоритет этих вызовов, введя в условие перестановки вызовов на очередном шаге итерации пониженные весовые коэффициенты для менее приоритетных вызовов. Таким образом, у приоритетных вызовов появится преимущество распределения по отношению к наименее удаленным и загруженным лифтам. Также возможна отправка части свободных лифтов на приоритетные этажи даже при отсутствии в конкретный момент времени вызовов с этих этажей.

В табл. 2 приведены результаты моделирования работы итерационного алгоритма в режиме максимального потока вверх с учетом приоритета вызовов с первого этажа и без учета приоритетов. Интенсивность прибытия пассажиров при моделировании составляет 6 человек в минуту.

Таблица 2 – Результаты моделирования

| Алгоритм | $T_{ож, с}$ | $T_{п, с}$ | $\%_{>мин}$ | $P_{на пасс, Вт}$ |
|--------------------------|-------------|------------|-------------|-------------------|
| Равный приоритет вызовов | 22 | 58 | 9 | 15,2 |
| Приоритет первого этажа | 7 | 49 | 0 | 19,5 |

Как видно из таблицы, учет особенностей пассажиропотока обеспечил прирост производительности группы лифтов на 30% при увеличении затрат электроэнергии на 20%.

ЗАКЛЮЧЕНИЕ

Разработана платформа для оценки эффективности алгоритмов диспетчерского управления. Разработана методика, позволяющая оценить эффективность реализованного алгоритма для условий конкретного здания и найти разумный компромисс между экономичностью и качеством обслуживания пассажиров. Рассмотрены основные современные алгоритмы управления группами лифтов. Осуществлено исследование достоинств и недостатков данных алгоритмов в различных условиях работы с разными типами пассажиропотоков. Разработан эффективный итерационный алгоритм диспетчерского управления с возможностью регулирования приоритетов производительности или экономичности, а также гибкой подстройки под характер пассажиропотока за счет регулирования коэффициентов приоритетности вызовов с различных этажей.

Полученные результаты моделирования показывают, что система управления лифтами выигрывает от перехода к наиболее подходящему алгоритму в условиях доминирующего в данное время пассажирского потока. Для потоков малой интенсивности использование специально разработанных алгоритмов не дает существенного преимущества по сравнению с простейшими алгоритмами, однако с ростом этажности здания, количества лифтов и интенсивности

пассажиропотока эти преимущества становятся достаточно существенными.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

[Каляев, 2001] Каляев И.А. Использование принципов коллективного принятия решений при управлении группой автоматических лифтов. // Мехатроника, №4, 2001.

[Клейнрок, 1979] Клейнрок Л. Теория массового обслуживания. Пер. с англ./Пер. И.И. Грушко; ред. В.И. Нейман. – М.: Машиностроение, 1979.

[Ронг и др, 2003] Ронг А., Хаконен Х., Ладелма Р. Алгоритм управления группой лифтов на основе расчетного времени прибытия (ETA) с более точной оценкой // Технический отчет 584. Центр компьютерных наук г. Турку (TUCS). Турку, Финляндия, 2003.

ELEVATOR CONTROL ALGORITHMS USING SEMANTIC INFORMATION

Kuznetsov A.P., Nikonov V.N.,
Shmarlouski A.S., Silivonets M.V.

*Belarusian State University of Informatics and
Radioelectronics, Minsk, Republic of Belarus*

kuznap@bsuir.by

nrkn@mail.ru

sas@bsuirby

a702@tut.by

INTRODUCTION (ВВЕДЕНИЕ)

There are a lot of algorithms to control a group of elevators. But almost all of them are developed for specific buildings conditions. The universal control algorithm possibilities of which are extended due to application of intelligent control systems has been considered in this article.

MAIN PART

Standard algorithms of supervisory control have both advantages and disadvantages manifesting in different conditions. The purpose of our development is a universal algorithm combining all advantages of standard algorithms having an intelligent component for optimization of operation of a group of elevators in current conditions.

Improvement of operation is provided by the redistribution of calls assigned for a single elevator which depends on conditions change out on the current character of passenger traffic.

CONCLUSION

Mostly applied control algorithms for controlling groups of elevators have been considered with the help of the developed software platform we have carried out analysis. The main factors affecting the passenger traffic are described. A universal algorithm for supervisory control of a group of elevators has been developed and simulations have proved the effectiveness of this algorithm.