

ОПТИМИЗАЦИЯ ВРЕМЕНИ ОЖИДАНИЯ ПРИ ДИСПЕТЧЕРИЗАЦИИ ГРУППЫ ЛИФТОВ НА ОСНОВЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Струц Д. А.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: dmitry.a.struts@gmail.com

Рассматривается использование генетического алгоритма для решения проблемы кодирования в диспетчеризации кабин. Предлагаемый метод уменьшает среднее время ожидания и использует более простой подход кодирования, который более дешевый с точки зрения вычислительной стоимости.

ВВЕДЕНИЕ

Элемент управления лифтами является одной из важных проблем вертикальных транспортных систем в зданиях. С точки зрения качества обслуживания два основных параметра для оптимизации, - время ожидания и время в пути.

I. ПАРАМЕТРЫ ВРЕМЕНИ И ТРАНСПОРТНАЯ МОДЕЛЬ

С точки зрения качества обслуживания эффективный алгоритм систем управления лифтовой группой должен посылать кабины лифтов на зарегистрированные вызовы, минимизируя время ожидания пассажиров. Время ожидания - время с регистрации вызова и до достижения кабиной заданного этажа. Время в пути начинается с прибытия лифта на этаж с пассажиром и заканчивается при прибытии на место назначения. Оценка времени поездки является более сложной, поскольку неопределенность из-за зависимости от входящих вызовов и их неизвестных направлений может существенно повлиять на нее. Путешествие - общее время с регистрации вызова до достижения места назначения и рассчитывается как сумма времени поездки (TT) и времени ожидания (WT).

$$JT = WT + TT \quad (1)$$

В зависимости от здания и времени суток можно наблюдать различные потоки движения. Вместо конкретных шаблонов рассматривается общий случайный трафик, пренебрегая другими ограничениями.

II. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Генетический алгоритм (ГА) - это алгоритм оптимизации, основанный на естественном процессе эволюции. В ГА популяция состоит из индивидуумов, каждый из которых представляет собой решение для максимизации (минимизации) функции приспособленности. Индивидуумы (или хромосомы) представляют битовые строки. Для процесса эволюции поколений ГА имитирует естественный отбор, мутацию и скрещивание.

При диспетчеризации лифтов, скрещивание применяется как общая замена генов у двух хромосом, гены которых представляют собой номера лифтов. Когда скрещивание применяется к паре, после случайного выбора номера лифта для каждого гена, кабины назначаются на вызовы на этажах. При мутации, для одной хромосомы случайно выбранный ген, который представляет номер лифта, заменяется другим номером лифта. Функция приспособленности для ГА определяется как:

$$f = 1/T_{av} \quad (2)$$

$$T_{av} = \left(\sum_{i=1}^K WT_i \right) / K \quad (3)$$

где K , WT_i и T_{av} представляют количество вызовов, расчетное время ожидания i -го вызова и среднее время ожидания соответственно. Входные параметры системы, которые будут использоваться при вычислении приспособленности, приведены в таблице 1.

Таблица 1 – Параметры системы

Обозначение	Значение
NF	Количество этажей
NC	Количество кабин
PT	Пассивное время
IT	Время межэтажного перекрытия
HC	Этажные вызовы
CF	Этажи расположения кабин

Сумма времени открытия/закрытия двери и времени погрузки пассажиров обозначено как «пассивное время» (PT).

III. СТРУКТУРА ХРОМОСОМЫ

В работе [1], следуя той же схеме, представленной в [2] для кодирования кабин в ГА, Тартан определил хромосому как конкатенированные массивы, где каждый массив представляет состояние присвоения номера кабины вызовам на этажах. Битовая длина хромосом $2M \times [NF - 1]$, где M - количество кабин, делает её громоздкой для вычисления. В [3] хромосома была адаптирована, где каждый ген соответствует этажу, по которым зарегистрирован вызов, отбрасывая

этажи без вызова. Каждое значение бита представляет собой идентификатор кабины, назначенный этому этажу. В этом подходе длина хромосом, но она более практична, чем двоичное кодирование с включением всех этажей. Этот подход уменьшает вычислительные затраты в ГА за счет уменьшения матрицы популяции от $2P \times M \times [NF - 1]$ до $P \times HC$, где P - количество хромосом.

IV. РАСЧЕТ ПРОГНОЗИРУЕМОГО ВРЕМЕНИ ОЖИДАНИЯ

В [4] были продемонстрированы возможные маршруты кабины. Маршруты кабины к этажу, с конечным пунктом назначения выше, чем этаж с вызовом, представлены на рис. 1.

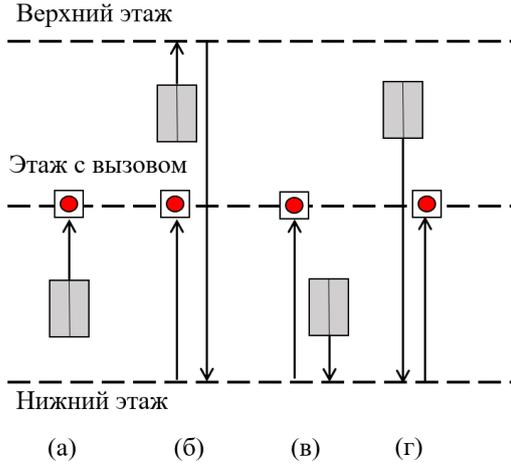


Рис. 1 – Маршруты кабины

Если кабина не движется или поднимается, расчетное время ожидания для i -го этажа с вызовом WT_i рассчитывается в (4), иначе в (5). Здесь n - номер кабины, присвоенный вызову на этаж. Как видно из (4) положение кабины CF_n , относительно позиции этажа с вызовом HC_i и направление движения определяет формулу для расчета. NS_i - минимальное количество остановок между этажом вызова и этажом с кабиной, назначенной на этот вызов. Вызовы регистрируются, и в условиях предварительной нагрузки известны номера кабин. Для выбора решения, ГА

рассматривает этажи с вызовами, которые назначены на эту кабину в NS_i . В обычных системах конечные пункты назначения вызова неизвестны, так как есть кнопки направления вверх и вниз. Значит промежуточный вызов может появиться до того, как кабина достигнет HC_i . Если кабина поднимается вверх, на самый высокий этаж для неё, то следующие вызовы будут назначены на кабину в обратном направлении. Поэтому проверяются первые условия в (4) и (5) и добавляются дополнительные условия. Если в кабина не имеет больше вызовов в своём направлении, то после выхода пассажира, который находился внутри, она может взять пассажира в обратном направлении. Тогда новые ограничения для первых условий в (4) и (5) становятся соответственно (6) и (7).

ЗАКЛЮЧЕНИЕ

Рассматриваемая модель сочетает некоторые свойства предыдущих исследований и улучшает метод, представленный в (Tartan, 2014). Данный метод:

1. Принимает во внимание начальные состояния кабины и условия предварительной нагрузки;
2. Использует более компактную десятичную схему кодирования вместо двоичной кодировки;
3. Улучшает функцию приспособленности для предполагаемого времени ожидания.

1. Tartan, E. O., Erdem, H. and Berkol, A. (2014). Optimization of waiting and journey time in group elevator system using genetic algorithm, Innovations in Intelligent Systems and Applications (INISTA) Proceedings, Italy.
2. Cortes, P., Larraneta, J. and Onieva, L. (2004). Genetic algorithms for controllers in elevator groups: Analysis and simulation during lunchpeak traffic, Appl. Soft. Comput., vol. 4, no. 2, pp. 159–174.
3. Bolat, B., Altun, O., Cortes, P. (2013). A particle swarm optimization algorithm for optimal car-call allocation in elevator group control systems, Appl. Soft Comput., 13 (5), pp. 2633–2642.
4. Jamaludin, J., Rahim, N. A. and Hew, W.P. (2009). Development of a self-tuning fuzzy logic controller for intelligent control of elevator systems. Eng Appl Artif Intel, 22(8): 1167–1178.

$$T_i = \begin{cases} [HC_i - CF_n] \times IT + NS_i \times PT; & \uparrow HC_i \geq CF_n \\ [(NF - CF_n) + (NF - HC_i)] \times IT + NS_i \times PT; & \downarrow HC_i \\ [(NF - CF_n) + (NF - 1) + (HC_i - 1)] \times IT + NS_i \times PT; & \uparrow HC_i < CF_n \end{cases} \quad (4)$$

$$T_i = \begin{cases} [CF_n - HC_i] \times IT + NS_i \times PT; & \uparrow HC_i \geq CF_n \\ [(CF_n - 1) + (HC_i - 1)] \times IT + NS_i \times PT; & \downarrow HC_i \\ [(CF_n - 1) + (NF - 1) + (NF - HC_i)] \times IT + NS_i \times PT; & \uparrow HC_i < CF_n \end{cases} \quad (5)$$

$$T_i = \begin{cases} [(\downarrow HC_{max} - CF_n) + (HC_{max} - HC_i)] \times IT + NS_i \times PT; & \downarrow HC_{max} = CDF_n(max) \\ \text{if all } \downarrow HC < CD_n; & \end{cases} \quad (6)$$

$$T_i = \begin{cases} [(CF_n - \uparrow HC_{min}) + (HC_i - \uparrow HC_{min})] \times IT + NS_i \times PT; & \downarrow HC_{min} = CDF_n(min) \\ \text{if all } \uparrow HC > CD_n; & \end{cases} \quad (7)$$