

Сравнение алгоритмов маршрутизации сети передачи данных

Чернышов А.И.¹

¹ Московский физико-технический институт (государственный университет)

В докладе рассматривается задача управления сетью передачи данных. Процесс, обеспечивающий передачу пакетов данных от источника к получателю в сети, называется маршрутизацией. Маршрутизация в больших сетях является сложной, поскольку пакет проходит много промежуточных узлов пока он достигнет узла назначения. Таким образом, алгоритм маршрутизации, обеспечивающий минимальное время или расстояние прохождения пакета данных, должен учитывать информацию о состоянии сети. Результат работы алгоритма маршрутизации – возможные маршруты между узлами и выбранный путь передачи трафика. Требованием к алгоритму является его высокая производительность, так как время на расчет пути ограничено, а общая производительность сети определяется маршрутизацией, в сочетании с контролем перегрузки сети и контролем доступа.

Под алгоритмом маршрутизации понимается правило, в соответствии с которым в каждом узле сети передачи данных осуществляется выбор линии связи для передачи блока данных (сообщения или пакета).

В работе предлагается рассмотреть следующие три алгоритма расчета оптимального пути передачи данных: алгоритм Беллмана – Формана; алгоритм Дейкстры; эвристические методы, например генетический алгоритм; на сетях малой, средней и большой размерности.

Алгоритм Беллмана - Формана (маршрутизация RIP) был разработан в 1969 году и предназначен для небольших и относительно однородных сетей. Маршрут характеризуется вектором расстояния до места назначения. Предполагается, что каждый маршрутизатор является отправной точкой нескольких маршрутов до сетей, с которыми он связан. Описания этих маршрутов хранится в специальной таблице, называемой маршрутной. Таблица маршрутизации содержит по записи на каждую обслуживаемую машину (на каждый маршрут).

Алгоритм Дейкстры (протокол OSPF) заключается в поиске наикратчайшего пути из

одной изначально заданной вершины графа до всех остальных. Данный алгоритм пошагово перебирает все вершины графа и назначает им метки, которые являются известным минимальным расстоянием от вершины источника до конкретной вершины. Этот алгоритм считается наиболее эффективным методом вычисления кратчайшего пути в IP сетях. Однако он становится неэффективным в больших сетях, так как возникает необходимость повторения расчетов. Также он не может быть решен за ограниченное время и не работает на графах, дуги которых имеют отрицательный вес [1].

Генетический алгоритм — это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, аналогичных естественному отбору в природе. [3] Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе. Алгоритм делится на три этапа: скрещивание, селекция (отбор), формирования нового поколения [2]. В существующих алгоритмах маршрутизации данный метод поиска пути не используется, поэтому в данной работе рассматривается как альтернатива существующим методам.

Рассмотрим следующую модель сети передачи данных (СПД). СПД состоит из N узлов коммутации и M линий связи. Маршрутизация может быть представлена в следующем виде. Пусть дан направленный взвешенный граф $G=(V,E)$, в котором каждый узел из множества V представляет собой устройство, обрабатывающее и передающее данные, а каждое ребро из множества E является линией связи. Основной задачей алгоритмов маршрутизации является передача данных из узла источника в узел приемник по оптимальному пути. Критерием оптимизации является скорость передачи данных или количество пройденных узлов.

Литература

1. *Black U.* IP Routing Protocols, RIP, OSPF, BGP, PNNI & Cisco routing protocols, 2000.
2. *Laurence Davis* Hand book of Genetic Algorithms, 1991.
3. *Sivanandam S.N. & Deepa S.N.* Introduction to Genetic Algorithms, 2008.