

**Седухин Д.В.** — рецензент *Финогенов А.Д.*  
УНК “ИПСА” НТУУ “КПИ”

## Параллельный “муравьиный” алгоритм для решения задачи коммивояжёра

Одной из важнейших задач комбинаторной оптимизации является задача коммивояжёра. Одним из методов её решения является “муравьиный” алгоритм [1]. При его инициализации каждый “муравей” располагается в случайной вершине графа. Затем каждый из них начинает движение по графу, двигаясь при этом от узла  $i$  к узлу  $j$  с вероятностью (1), где  $\tau_{i,j}$  – это количество феромона на ребре  $i, j$ ;  $\alpha$  – параметр влияния на  $\tau_{i,j}$ ;  $\eta_{i,j}$  – величина, обратная длине ребра  $i, j$ ;  $\beta$  – параметр влияния на  $\eta_{i,j}$ .

$$P_{i,j} = \frac{(\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}{\sum (\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}, \tau_{i,j} = (1 - \alpha)\tau_{i,j} + \Delta\tau_{i,j}. \quad (1)$$

Затем “муравей” перемещается по ребру с наибольшей вероятностью и обновляет на нём значение феромона по принципу (2), где  $\tau_{i,j}$  – количество феромона на ребре  $i, j$ ,  $\rho$  – коэффициент испарения феромона, а  $\Delta\tau_{i,j}$  – количество дополнительного феромона, равное:

$$\Delta\tau_{i,j}^k = \begin{cases} 1/L_k, & \text{если } k\text{-й “муравей” прошёл по ребру } i, j, \\ 0, & \text{в противном случае.} \end{cases}$$

где  $L_k$  – длина пути  $k$ -го “муравья”. Затем выбирается минимальный путь, и, если он не удовлетворяет конечным условиям, поиск продолжается новым проходом “муравьёв” по графу. В отличие от последовательной реализации (рис. 1.а), в параллельном алгоритме каждый следующий “муравей” уже в начале пути может рассчитывать вероятность перехода по ребру на основе значения феромона, обновлённого предыдущим “муравьём” на этом же проходе (рис. 2.б). Пусть  $n$  – число городов, а  $m$  – число “муравьёв”. Таким образом, вероятность прохода по ребру, принадлежащему кратчайшему пути, будет возрастать при каждом проходе всех “муравьёв” в  $m$  раз быстрее. А скорость одного прохода алгоритма сократиться с  $O(mn)$  до  $O(n)$ .

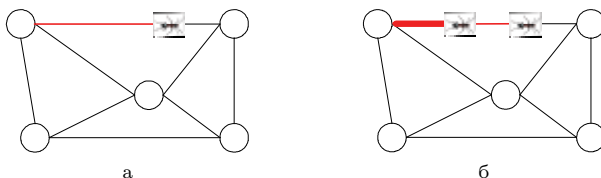


Рис. 1. Движение “муравьёв” в последовательном (а) и параллельном (б) алгоритме. Красным обозначен след феромона.

**Выводы.** “Муравьиный” алгоритм используется для решения задач диспетчеризации, транспортной задачи, задачи целераспределения, задачи сетевой маршрутизации [2], и т. п. Параллельная версия данного алгоритма позволяет существенно уменьшить временные затраты при поиске оптимального решения.

### Литература

1. Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem. TR/IRIDIA/1996-5, University Libre de Bruxelles, Belgium, 1996. – 24с.
2. [http://www.codeproject.com/KB/recipes/Ant\\_Colony\\_Optimisation.aspx](http://www.codeproject.com/KB/recipes/Ant_Colony_Optimisation.aspx) Lawrence Botley. Artificial intelligence network load balancing using Ant Colony Optimisation, 4 May 2006.