

## МЕТОДИКА ФОРМИРОВАНИЯ СОСТАВА ОТКАЗОУСТОЙЧИВОГО ПРОГРАМНОГО ОБЕСПЕЧЕНИЯ НА БАЗЕ МУРАВЬИНОГО АЛГОРИТМА

Е.В.Соловьев

*Институт космических и информационных технологий СФУ*

Задача создания надежного и отказоустойчивого программного обеспечения всегда будет стоять перед его разработчиками. Одним из вариантов позволяющим существенно повысить надежность является использование методологии мультиверсионного программирования. Основной идеей данной методологии является использование в рамках одного модуля нескольких версий. Но тогда перед разработчиком встает выбор оптимального состава данных версий. Для решения данной задачи существуют различные алгоритмы. В данной статье будет рассмотрен муравьиный алгоритм.

Идея муравьиного алгоритма – моделирование поведения муравьев, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи и адаптироваться к изменяющимся условиям, находя новый кратчайший путь. При своём движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути. Это элементарное правило поведения и определяет способность муравьев находить новый путь, если старый оказывается недоступным

Основные ограничения, которые стоят перед разработчиком, это стоимость и надежность программного обеспечения. Поэтому классический муравьиный алгоритм изменяется для решения задачи с учетом данных ограничений.

Начнем с изменения вероятностно-пропорционального правила таким образом, чтобы оно отражало вероятность выбора версии  $j$ . Так же надо учесть как стоимость, так и надежность версий. Тогда формула примет следующий вид:

$$p_j(t) = \frac{[\tau_j(t)]^\alpha \left[ \frac{1}{C_j} \right]^\beta [R_j]^\varphi}{\sum_{j \in \text{allowed nodes}} [\tau_j(t)]^\alpha \left[ \frac{1}{C_j} \right]^\beta [R_j]^\varphi} \quad (1)$$

где  $p_{ij}$  - вероятность выбора  $j$ -ого модуля;

$C_j$  - стоимость  $j$ -ого модуля;

$R_j$  - стоимость  $j$ -ого модуля;

$\alpha$  — параметр, задающий вес следа феромона;

$\beta$  и  $\varphi$  параметры варьируя, которые можно сдвигать наше решение либо в сторону надежность, либо в сторону стоимости.

Формула обновления феромона:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k \in \text{Colony that used edge } (i,j)} \left( \frac{R_{ij}}{\text{Max}(R_{ij}, j=1, m_i)} \right) \left( \frac{\text{Min}(C_{ij}, j=1, m_i)}{C_{ij}} \right) \quad (2)$$

$\rho \in [0, 1]$  - коэффициент испарения феромона.

Рассмотрим работу алгоритма на примере выбора версии в следующем модуле программного обеспечения:

Название версии	Стоймость версии	Надежность версии
Version 1	57	0.83
Version 2	38	0.7
Version 3	71	0.96
Version 4	63	0.87

В результате работы алгоритма получены следующие данные:

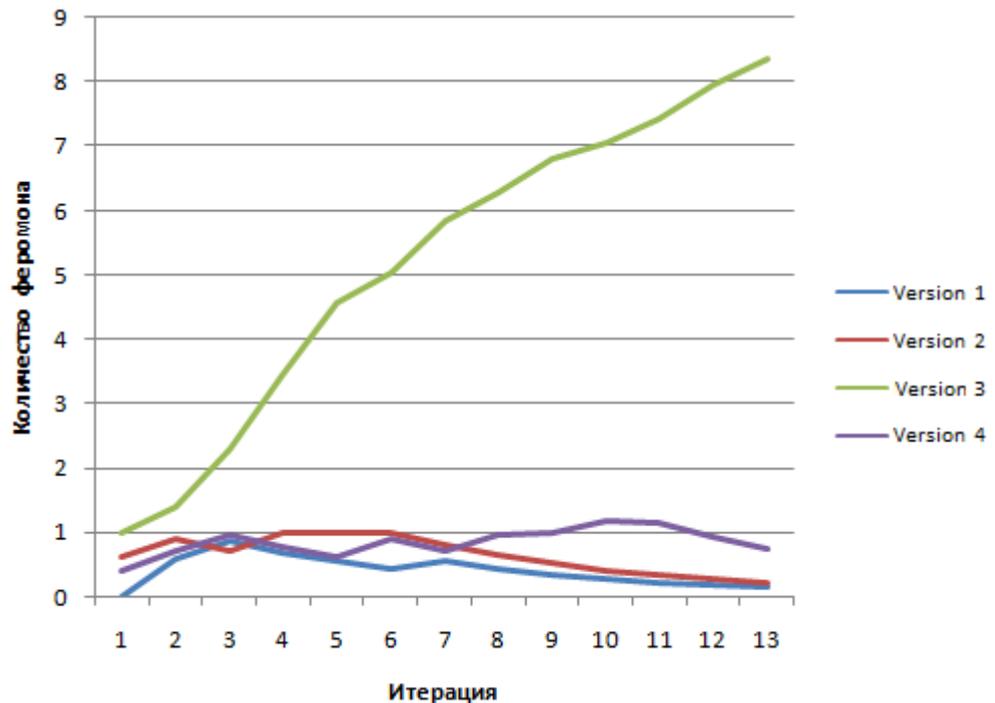


Рисунок 1 – График изменения количества феромона при  $\alpha = 2$ ,  $\beta = 4$ ,  $\gamma = 3$

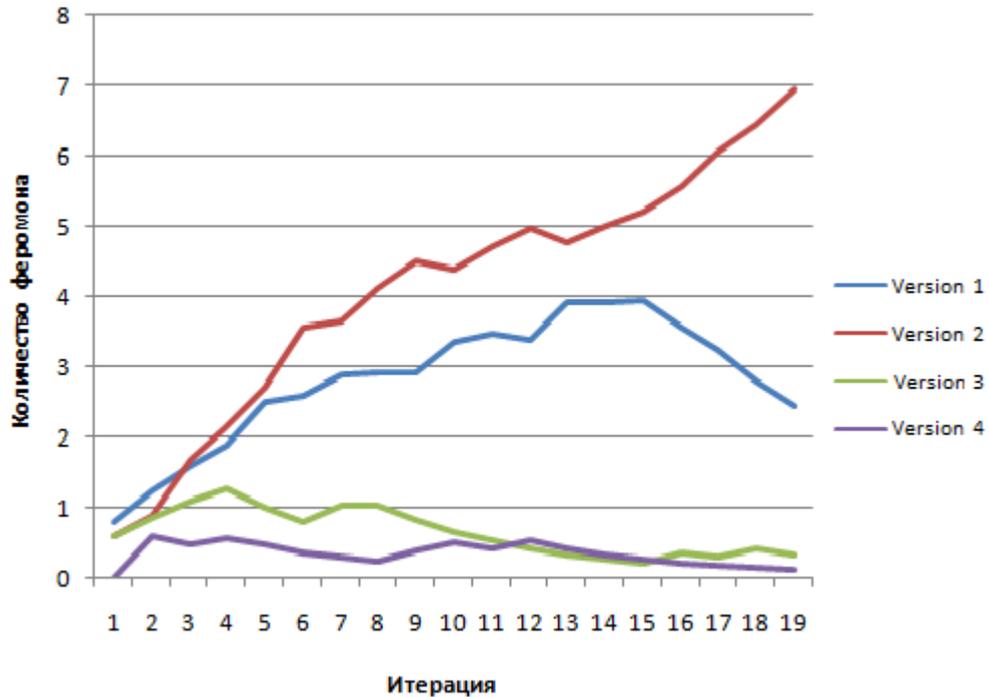


Рисунок 2 – График изменения количества феромона при  $\alpha = 2$ ,  $\beta = 0.4$ ,  $\gamma = 2$

Как видно из представленных выше данных муравьиный алгоритм можно использовать для нахождения оптимального состава мультиверсионного программного обеспечения. Он не только решает поставленную задачу, но делает это в разумный промежуток времени. Основной задачей пользователя является подбор таких коэффициентов в формуле вероятностно-пропорционального правила(1) таким образом, чтобы получить решение, которое будет удовлетворять ограничениям надежности и стоимости, которые он накладывает.

#### Список литературы:

- 1 Ковалёв, И. В. Мультиверсионный метод обеспечения отказоустойчивости программных архитектур систем управления критическими по надёжности техническими объектами / И.В. Ковалёв, Р.В. Юнусов // Науч. конф. «Научно-инновационное сотрудничество». Москва, 2002. – 172 с.
- 2 Штовба, С.Д. Муравьиные алгоритмы / С.Д. Штовба // EXPONENTA Pro. Математика в приложениях, – 2003. №4. – С.70-75