

РОЗПАРАЛЕЛЮВАННЯ КОМБІНОВАНОГО МУРАШИНОГО АЛГОРИТМУ

Мурашиний алгоритм – один з ефективних поліноміальних алгоритмів для знаходження наближених розв'язків задачі комівояжера, а також розв'язку аналогічних завдань пошуку маршрутів на графах. Підхід запропонований бельгійським дослідником Марко Доріго (англ. Marco Dorigo). Суть підходу полягає в аналізі та використанні моделі поведінки мурах, що шукають дороги від колонії до їжі.

У основі алгоритму лежить поведінка мурашиної колонії — маркування вдалих доріг великою кількістю феромону. Робота починається з розміщення мурашок у вершинах графа (містах), потім починається рух мурашок — напрям визначається імовірнісним методом, на підставі формули(1):

$$P_i = \frac{l_i^q \cdot f_i^p}{\sum_{k=0}^N l_k^q \cdot f_k^p} \quad (1)$$

де:

- P_i — ймовірність переходу шляхом i ,
- l_i — довжина i -ого переходу,
- f_i — кількість феромонів на i -ому переході,
- q — величина, яка визначає «жадібність» алгоритму,
- p — величина, яка визначає «стадність» алгоритму і
- $q+p=1$

Результат не є точним і навіть може бути одним з гірших, проте, в силу імовірності рішення, повторення алгоритму може видавати (досить) точний результат.

В природі, мурахи (початково) ходять довільним шляхом, і по знаходженні їжі повертаються до колонії, залишаючи по собі феромоновий слід. Якщо інші мурахи знаходять такий шлях, вони, імовірніше за все, будуть слідувати позначеним шляхом, посилюючи його під час повернення у разі знайдення їжі.

Однак, з часом, феромонові шляхи випаровуються, тоді привабливість шляхів зменшується. Чим більше часу потрібно мураці для проходження шляху до цілі і повернення назад, тим більше часу мають феромони, щоб випаруватись. Натомість, короткий шлях проходиться частіше, отже щільність феромонів стає більшою на короткому шляху. Випаровування феромонів також надає перевагу уникнення локально найкращих шляхів. Якби феромони не випаровувалися б взагалі, то шлях, обраний першим мурахою, був би найбільш привабливим для наступних мурах. В цьому разі, розвідка можливих шляхів була б обмежена.

Таким чином, коли мураха знаходить вдалих (тобто короткий) шлях з колонії до джерела їжі, інші мурахи, швидше за все, слідуватимуть йому, і позитивний зворотний зв'язок зрештою призведе до обрання цього шляху всіма мурахами. Ідея мурашиного алгоритму полягає в наслідуванні поведінки з «симулятором мурахи», що прогулюється графом, який представляє проблему, що треба розв'язати.

Припустимо, що навколишнє середовище для мурах представляє повнозв'язний неорієнтований граф. Кожне ребро має вагу, яка позначається як відстань між двома вершинами, що ним з'єднується. Граф є двохскерованим, тому мураха може подорожувати по грані в будь-якому напрямку.

Пройдений мурахою шлях відображається, коли мураха відвідає всі вузли графа. Цикли заборонено, оскільки в алгоритм включено список табу. Після завершення довжина шляху може бути підрахована - вона дорівнює сумі довжин всіх ребер, якими подорожувала мураха. Рівняння (2) показує кількість феромону, який був залишений на кожному ребрі шляху для мурашки k . Змінна Q є константою.

$$\Delta \tau_{ij}^k(t) = \frac{Q}{l^k(t)} \quad (2)$$

Результат рівняння є засобом вимірювання шляху, - короткий шлях характеризується високою концентрацією феромонів, а більш довгий шлях - більш низькою. Далі, отриманий результат використовується в рівнянні (3), щоб збільшити кількість феромону вздовж кожного ребра пройденого мурахою шляху.

$$\tau_{ij}(t) = \Delta \tau_{ij}(t) + (\tau_{ij}^k \cdot \rho) \quad (3)$$

Важливо, що дане рівняння застосовується до всього шляху, при цьому кожне ребро позначається феромоном пропорційно до довжини шляху. Тому слід дочекатися, поки мураха закінчить подорож і лише потім оновити рівні феромону, в іншому випадку справжня довжина шляху залишиться невідомою. Константа ρ - значення між 0 і 1.

На початку шляху у кожного ребра є шанс бути обраним. Щоб поступово видалити ребра, які входять в гірші шляхи графа, до всіх ребер застосовується процедура випаровування феромону. Використовуючи константу ρ з рівняння (3), отримуємо рівняння (4):

$$\tau_{ij}(t) = \tau_{ij}(t) \cdot (1 - \rho) \quad (4)$$

Приклади дій для оптимізації.

Застосування розпаралелювання.

Такий підхід дозволить значно збільшити швидкість отримання кінцевого результату.

Існує декілька видів розпаралелювання мурашиного алгоритму:

- Глобальні однопопуляційні — існує одна головна популяція на основному ядрі, а обрахунок функції цілі розпаралелено на декілька процесорів.

- Однопопуляційні — знаходження шляху кожного мурахи відбувається на окремому ядрі, так само, як і обрахунок його довжини і оновлення феромону на ньому.

- Багатопопуляційні — на кожному ядрі існує декілька незалежних популяцій, індивіди кожної з яких шукають свій шлях. Після знаходження шляхів кожним мурахою з популяції відбувається оновлення феромону на шляхах цієї популяції.

Мурахи, що знаходяться на одному ядрі, зазвичай називаються колонією. Після завершення роботи кожної колонії можна вираховувати найкраще рішення з усіх колоній і пересилати інформацію про найкоротший шлях і кількість феромону на ребрах всім іншим колоніям. Таким чином всі колонії володітимуть найкращим рішенням на даному кроці. Окрім цього можна спробувати здійснювати обмін цією інформацією не на кожному кроці, а через певний проміжок часу.

В якості новизни в даному проєкті можна зазначити наступне:

Мурашиний алгоритм можна зкомбінувати з алгоритмом імітації відпалу для покращення певних ділянок знайденого шляху.

Принципи, які покладені в основу роботи алгоритмів, можна пояснити на наступній фізичній аналогії. На рис.1. зображено кульку в коробці, внутрішня поверхня якої відповідає формі функції цілі. При сильному струшуванні коробки в горизонтальному напрямку кулька може переміститися з будь-якої точки в будь-яку іншу. При поступовому зменшенні сили струшування буде досягнуто умови, коли сила струшування достатня для переміщення кульки з точки в А в точку В, але недостатня для того, щоб кулька змогла переміститися з точки В в А. При подальшому зменшенні сили струшування до нуля, кулька зупиниться в точці В — точці глобального мінімуму. В алгоритмах імітації відпалу аналогом сили струшування є ймовірність переходу в стан з більш високим значенням функції цілі. В процесі роботи алгоритму вірогідність переходу зменшується в залежності з вибраним законом

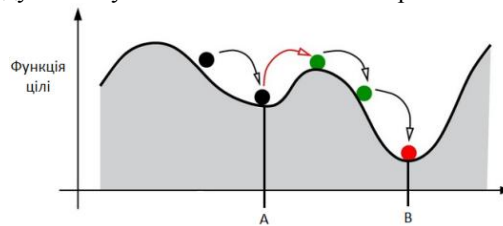


Рис. 1. Алгоритм імітації відпалу

Алгоритм імітації відпалу в процесі пошуку оптимального рішення з деякою ймовірністю допускають перехід в стан з гіршим(у випадку мінімізації — з вищим) значенням функції цілі. Ця властивість дозволяє йому виходити з локальних оптимумів.

Даний алгоритм не гарантує знаходження мінімуму функції, проте, як правило, відбувається покращення початкового приближення, що ми і можемо застосувати для покращення певних ділянок шляху, знайденого за допомогою мурашиного алгоритму.

Наступним нововведенням є те, що мурахи в межах кожної колонії можуть мати різну кількість феромону. Наприклад, в першій колонії кількість феромону у кожного мурахи становить 1, в другій — 2 і т.д. Метою роботи буде дослідження впливу такого експерименту на стан роботи алгоритму

Також, через певний проміжок ітерацій до попереднього рішення можна застосувати генетичні алгоритми і почати перемішувати індивідів між колоніями, що мають різну кількість феромону.