

# ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ДЛЯ ОПТИМИЗАЦИИ ПОДБОРА ПАРАМЕТРОВ ТОРГОВЫХ БОТОВ

Хлопцев А. А., Кузьма Ю. В.

Факультет компьютерных сетей и систем, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: RedLeonFire@yandex.by, dzotcrew@gmail.com

*Задачами данной публикации являются: краткое описание автоматизированной торговли, а также демонстрация того, каким образом можно применить генетический алгоритм к оптимизации подбора параметров торговых ботов. В завершении статьи представлено несколько тестов для сравнения времени работы двух видов генетического алгоритма и алгоритма полного перебора*

## ВВЕДЕНИЕ

Торговый бот - это программа, реализующая специальный алгоритм, который работает на бирже в режиме реального времени и принимает решения без участия человека. Торговля, при которой совершение сделок происходит с использованием программ-советников (ботов), называется автоматизированной торговлей [1].

Преимущества автоматизированной торговли:

- компенсирование человеческого фактора (влияние эмоций, усталость, и т.д.);
- высокая скорость операций и моментальное реагирование на происходящие события;
- круглосуточное функционирование;
- одновременная работа с большим количеством информации.

Перед тем, как доверить боту деньги и запустить его на настоящем торговом сервере, необходимо провести качественное тестирование программы, чтобы:

- исключить наличие ошибок в боте;
- проверить алгоритм на работоспособность;
- ускорить получение результатов работы алгоритма.

Для тестирования торговых ботов применяются специальные тестировочные модули, которые встроены в торговые терминалы. В торговом терминале MetaTrader компании MetaQuotes Software Corp. данный модуль называется Тестировщик (Tester), в терминале AlgoTerminal компании Soft-FX – Бектестер (Backtester). Названные модули позволяют запустить торговый бот на историческом промежутке данных, на котором эмулируется его работа, что позволяет сравнить результаты торговли при различных наборах входных параметров и выбрать самый оптимальный (прибыльный) из них. Обычно для генерации наборов входных параметров ботов используется алгоритм полного перебора, но с увеличением количества параметров работа данного алгоритма начинает занимать очень много времени, поэтому требуется найти более эффективное решение. Одним из вариантов решения дан-

ной задачи является применение генетического алгоритма.

## I. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

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. Является разновидностью эволюционных вычислений, с помощью которых решаются оптимизационные задачи с использованием методов естественной эволюции, таких как наследование, мутации и отбор. Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе [2].

Поскольку алгоритм самообучающийся, то спектр применения крайне широк:

- задачи на графы;
- задачи компоновки;
- составление расписаний;
- создание «Искусственного интеллекта» [3].

Задача формализуется таким образом, чтобы её решение могло быть закодировано в виде вектора генов, где каждый ген может быть числом или другим объектом.

Некоторым, обычно случайным, образом создаётся начальный набор генов. Он оценивается с использованием «функции приспособленности», в результате чего с каждым набором ассоциируется определённое значение («приспособленность»), которое определяет насколько хорошо данный набор решает поставленную задачу.

Из полученного множества с учётом значения «приспособленности» выбираются наборы (обычно лучшие особи имеют большую вероятность быть выбранными), к которым применяются «скрещивание» и «мутация», результатом чего является получение новых наборов. Для них также вычисляется значение приспособленности,

и затем производится отбор («селекция») лучших наборов в следующее поколение.

Эти действия повторяются многократно, так моделируется «эволюционный процесс», продолжающийся несколько жизненных циклов (поколений), пока не будет выполнен критерий остановки алгоритма.

Таким образом, можно выделить следующие этапы генетического алгоритма:

- задать целевую функцию (приспособленности) для особей популяции;
- создать начальную популяцию;
- начало эволюционного цикла:
  1. размножение (скрещивание);
  2. мутирование;
  3. вычислить значение целевой функции для всех особей;
  4. формирование нового поколения (селекция);
  5. если выполняются условия остановки, то конец эволюционного цикла [2].

## II. ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ НАСТРОЙКИ ТОРГОВОГО БОТА

Для работы боту необходим набор входных параметров, которые применяются в торговом алгоритме. Чтобы ограничить область поиска оптимального набора, для каждого параметра должны выполняться следующие условия:

1. параметр не должен быть больше некоторого значения  $Max$ ;
2. параметр не должен быть меньше некоторого значения  $Min$ ;
3. разница между параметром и его минимальным значением должна быть кратна некоторому значению  $Step$ .

Таким образом, каждый параметр должен удовлетворять системе ограничений (1).

$$\begin{cases} value \leq Max \\ value \geq Min \\ value = Min + i \cdot Step, i \in \mathbb{Z} \end{cases} \quad (1)$$

Теперь представим задачу подбора параметров так, чтобы её можно было решить с помощью генетического алгоритма:

- параметр является единичным геном;
- набор входных параметров представляет собой вектор генов;
- функция приспособленности – результат тестирования бота с данным набором параметров (чистая прибыль);
- в качестве мутации выберем функцию изменения текущего значения параметра на

некоторое число, удовлетворяющее системе (1).

Примерное время работы генетического алгоритма в рамках поставленной задачи вычисляется по формуле (2):

$$T = t \cdot countH \cdot (n \cdot \log(n) + n + mut), \quad (2)$$

где

- $t$  – время работы бота на отрезке;
- $countH$  – количество поколений;
- $n$  – количество элементов в поколении;
- $mut$  – количество мутаций в поколении.

## III. ТЕСТИРОВАНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Для измерения времени подбора параметров были выбраны 2 разновидности генетического алгоритма и стандартный алгоритм полного перебора. Результаты тестирования представлены в таблице 1.

Таблица 1 – Сравнение времени работы различных алгоритмов

Количество наборов параметров	Полный перебор, мс	Генетический алгоритм с равномерной селекцией, мс	Генетический алгоритм с рулеточной селекцией, мс
5	624	1041	1101
20	1494	1703	1802
100	4163	2117	2351
500	19002	4160	4181
1000	78889	5498	5701
7500	249158	5501	5766
15000	500983	7023	7478

## ЗАКЛЮЧЕНИЕ

Из таблицы 1 видно, что если тестируемых наборов параметров не много (меньше 20), то для генерации наборов входных параметров лучше использовать алгоритм полного перебора. Иначе наблюдается сильный рост времени работы данного алгоритма, из чего можно сделать вывод, что при больших наборах параметров выгоднее использовать генетический алгоритм.

1. Технический анализ фьючерсных рынков: Теория и практика / Д. Д. Мэрфи – «Альпина Диджитал», 2011. – 479 с.
2. Алгоритмы эволюционной оптимизации / пер. с англ. А. В. Логунова. – М.: ДМК Пресс, 2020. – 940 с.
3. Алгоритмы. Руководство по разработке. – 2-е изд.: Пер. с англ. – СПб.: БХВ-Петербург, 2016. – 720 с.