

УДК 004.415

студ. О.В. Луцевич

Науч. рук. доц. А.И. Бракович

(кафедра информационных систем и технологий, БГТУ)

ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ РЕАЛИЗАЦИИ АЛГОРИТМОВ ПОИСКА НА ПРИМЕРЕ ШАХМАТНОЙ ИГРЫ

Машина играет настолько умно, насколько ее научит человек. Есть простые игры, типа "Крестиков-ноликов", с ограниченным количеством ходов. В таких играх машину ничему не нужно учить, она легко просчитывает ситуацию до конца и играет идеально. В шахматах количество ходов намного больше, и дерево перебора огромно. До конца просчитать ситуацию, кроме некоторых редких случаев, просто невысказимо. Как же машина играет и выбирает лучший ход?

1951 году, математик Клод Шеннон пишет свою первую статью о программировании шахмат. Он писал: «Хотя, возможно, это и не имеет никакого практического значения, сам вопрос является, теоретически, интересным, и будем надеяться, что решение этой проблемы послужит толчком для решения других проблем аналогичной природы». Шеннон отмечает теоретическое существование лучшего хода в шахматах и практическую невозможность найти его. Он описывает две стратегии поиска лучшего хода, обе основываются на эвристической функции оценки конечных точек:

- перебор всех возможных ходов на фиксированную глубину, с вызовом в конце оценочной функции (т. к. невозможно осуществить перебор до конца);
- выполняет только выборочное расширение определенных строк, используя накопленные шахматные знания, чтобы подрезать неинтересные ветви.

Теперь разберём сами алгоритмы. Для примера возьмём несколько алгоритмов: AlphaBeta, который основан на переборе всех возможных позиций, и Giraffe, обучением которого занималась нейросеть.

AlphaBeta. Как уже было упомянуто выше, AlphaBeta основан на полном переборе всех позиций, правда с механизмом осечения ветвей. Кажется на первый взгляд, что нельзя досрочно прекратить поиск в узле без ухудшения качества поиска. Но это не так. Дерево поиска,

формируемое рекурсивно, можно представить как своеобразную нейронную сеть. Лучшая оценка увеличивается, пока не достигнет некоторого предела. Какого? В этом весь вопрос. Представьте себе некоторый узел поиска в дереве перебора. Оценка в нем только увеличивается. Достигнутый максимум передается далее. Оппонент занижает оценку. Если он ее уже занизил до нашего максимума, то дальше искать нет смысла. Это позволяет существенно снизить поиск, отбрасывая ненужные ветви. Хоть алгоритм и кажется простым, тем не менее, он очень эффективен. До сих пор алгоритмы на его основе успешно учувствуют в международных соревнованиях.

Giraffe. Студент Имперского колледжа Лондона Мэтью Лай (Matthew Lai) разработал компьютерную программу Giraffe, которая трое суток играла в шахматы сама с собой — и смогла извлечь все предметно-ориентированные знания, необходимые для игры на уровне международного мастера ФИДЕ.

Для этого ей достаточно вычислительных ресурсов нормального компьютера. Обучение нейросети происходило 72 часа в 20 тредов на машине с двумя 10-ядерными процессорами Xeon E5-2660. В научной работе автор пишет, что после тренировки в течение 72 часов программа выбирала наилучший возможный ход в 46% случаев, а один из трёх наилучших ходов — в 70% случаев. Это очень неплохой результат даже для обычных шахматных программ. Но этот алгоритм достиг своего предела, и дальнейшее его обучение потеряло смысл (рисунок 1).

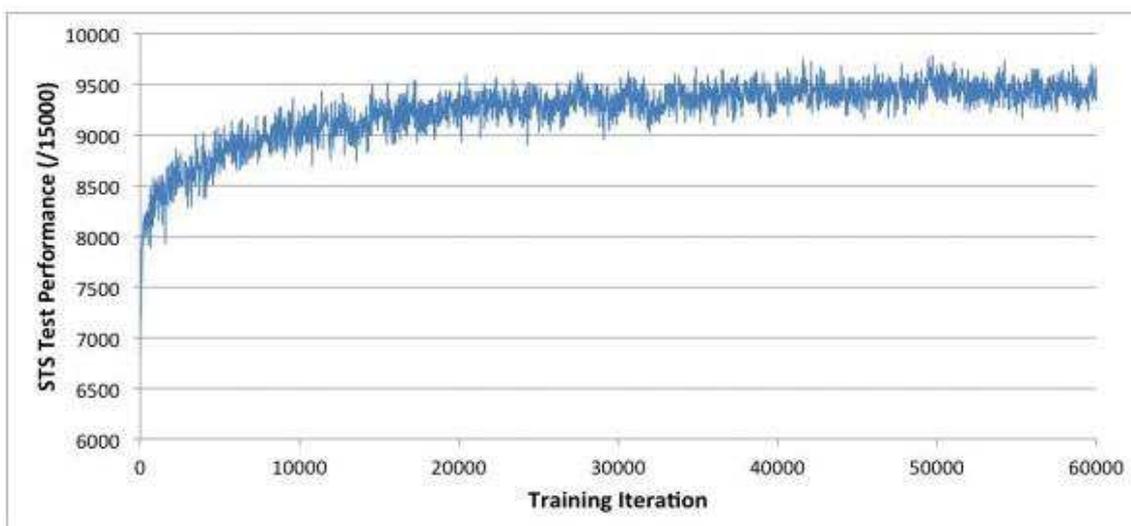


Рисунок 1 – Скорость обучения Giraffe

Муравьиный алгоритм. В основе алгоритма лежит поведение муравьиной колонии — маркировка более удачных путей большим количеством феромона.

Работа начинается с размещения муравьёв в на шахматной доске, затем начинается движение муравьёв — направление определяется вероятностным методом, на основании формулы (1):

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

Решение не является точным и даже может быть одним из худших, однако, в силу вероятности решения, повторение алгоритма может выдавать (достаточно) точный результат. Эффективность муравьиных алгоритмов сравнима с эффективностью общих мета эвристических методов, а в ряде случаев — и с проблемно-ориентированными методами.

Наилучшие результаты муравьиные алгоритмы показывают для задач с большими размерностями областей поиска. Муравьиные алгоритмы хорошо подходят для применения вместе с процедурами локального поиска, позволяя быстро находить начальные точки для них.

Идея применения заключается в том, чтобы объединить AlphaBeta и Муравьиный алгоритм, тем самым добавив в него человечности, и возможность обучения.

Результатом выполнения данной работы является клиент-серверное приложение, с помощью которого можно наблюдать за эффективностью различных алгоритмов натравливая, их друг на друга, либо самим попробовать выиграть их в честном бою.

ЛИТЕРАТУРА

1. Евгений Корнилов «Программирование шахмат и других логических игр», 2005. – 272 с.