

# 囲碁ソフトとスーパーコンピュータ

加藤 英樹, 竹内 郁雄

東京大学大学院情報理工学系研究科創造情報学専攻

## 1. はじめに

コンピュータ・ゲームは人工知能の黎明期から常に挑戦的な課題であった。現在、チェスは家庭用のパソコンでも人間のチャンピオンより強く、将棋ではコンピュータに勝てる人間は世界に数十人しかおらず、名人に勝つのも時間の問題と言われるレベルに達している。にも関わらず、囲碁は例外的にアマチュアの級クラスの強さで長年停滞していた。

しかし、2006年にイタリアのトリノで開かれた第11回コンピュータ・ゲーム・オリンピック<sup>1</sup>で Crazy Stone というプログラムが9路碁部門で優勝してから状況が激変した。全く新しい、モンテカルロ探索 (Monte-Carlo tree search; MCTS) と呼ばれる手法の登場である。それから3年余り経った現在、MCTSにも改良が加えられ、トップレベルのプログラムは、19路で強いアマチュア三段とのお墨付きを貰い、9路ではプロ棋士と互先 (ハンディ無し) で戦えるレベルに達している。人間とコンピュータのこれまでの対局結果は “Human-Computer Go Challenges<sup>2</sup>” というサイトに良くまとめられている (英語)。

## 2. モンテカルロ探索

チェスや将棋等、これまでのゲーム・プログラムは全て静的評価関数と  $\alpha\beta$  探索を用いて人間を超えるレベルに到達してきた。しかし、この手法は囲碁にはあまり有効ではなく、囲碁プログラムは長い間段に届かないレベルで停滞していた。この理由について述べるにはスペースが足りないが、これを打破するには現在モンテカルロ探索と呼ばれている新しい手法が必要であった。

モンテカルロ探索は二つの要素から成る。一つは、ランダム性を許容する木探索アルゴリズムで、最適化分野で古典的な多腕バンディット問題の解法として発見された UCB (upper confidence bound) アルゴリズム<sup>3</sup> を木探索に適用した UCT (UCB applied to trees) アルゴリズム<sup>4</sup> が広く用いられている。

もう一つはランダム・シミュレーションで、ある局面から適当に (ランダムに) 手を選びながら終局まで打ち進め、結果 (勝/負) をその局面の評価とする。これを何度も繰り返して平均するとその局面の評価値が得られる。言い換えれば、勝ちに至る手順が多い局面が「良い」局面であるということになる。この様な単純な方法でも、シミュレーション回数を増やせば局面の評価値としてそれなりに使えるものが得られる。実際のプログラムでは、シミュレーション中の手を選ぶのに囲碁の知識を利用しており (アタリならツグ等)、これでプログラムの強さは

---

<sup>1</sup> <http://www.grappa.univ-lille3.fr/icga/event.php?id=1&lang=3>

<sup>2</sup> <http://www.computer-go.info/h-c/index.html>

<sup>3</sup> Peter Auer, et al. Finite-time Analysis of the Multiarmed Bandit Problem, Machine Learning, vol. 47, pp. 235-256, 2002.

<sup>4</sup> Levente Kocsis, et al. Bandit based Monte-Carlo planning, ECML06, pp. 282-293, 2006.

大きく変わるが、原理的には囲碁のルールさえ知っていればそれなりに機能するという事は、その分野の知識を詰め込んだ評価関数が必要なこれまでの探索手法と異なり、モンテカルロ探索が色々な分野に簡単に適用できることを意味している。

モンテカルロ・シミュレーションは、これまで物理学や量子力学分野の数値計算で広く用いられており、ゲーム分野でもポーカーの様な不確定性のあるものには使われてきたが、これが囲碁の様な不確定性の無い、完全情報ゲームにも有効であることは新しい発見である。

### 3. 並列モンテカルロ探索

MCTS には 3 種類の並列化が考えられている。一つ目はメモリ共有型マルチプロセッサに適す、複数のスレッドで探索木を共有してシミュレーションは並列に実行する（各スレッドは探索木にアクセスする時だけ排他制御される）探索木共有対称マルチスレッド並列である。これは実装も容易で近年廉価になったマルチコア・プロセッサとの相性も良く、ほぼ全てのモンテカルロ囲碁ソフトで採用されている。

二つ目は、ある局面のモンテカルロ・シミュレーションを複数個同時に実行するもので、末端並列と呼ばれており、必要な記憶容量が小さく通信遅延による性能低下が大きいので、Cell Broadband Engine<sup>5</sup> の様な通信時間が短い分散記憶型非対称マルチプロセッサに適している。

三つ目はルート並列と呼ばれ、探索木のルート・ノードだけ共有する。各プロセッサは個々に探索木を持って独立に探索を行いつつ、一定周期（例えば 0.1 秒）でルート・ノードの情報を交換して平均化する。通信頻度が低く、また通信遅延による性能低下も比較的小さいので、ユニフォームなプロセッサをネットワークで結合したコンピュータ・クラスタに適している。これら三つの手法は組み合わせて使うことができる。

今回我々が対象にした東京大学情報基盤センターの HA8000 クラスタシステム<sup>6</sup> は、4 コアの AMD Opeteron 4 個から成る共有記憶型マルチプロセッサをネットワークで接続しているので、一番目と三番目の並列化手法を併用している。

### 4. さいごに

筆者らは、昨年 5 月にスペインのパンプローナで開かれた第 14 回コンピュータ・ゲーム・オリンピック<sup>7</sup> で優勝した Zen を、作者の協力を得てルート並列化し<sup>8</sup>、世界最大規模のコンピュータ囲碁トーナメントである第 3 回 UEC 杯<sup>9</sup> に参加した。HA8000 の 32 ノード（512 コア）を用い、残念ながら優勝は逃したが、エキジビションで青葉かおり 4 段<sup>10</sup> と 6 子のハンディで対局する機会を得た。クラスタ版 Zen は終始優勢に局面を進めたが、最後に劫争いで逆転負けを喫した。この棋譜は下記の URL<sup>11</sup> から通常のウェブ・ブラウザで観戦することができるので、ご覧頂ければ現在のトップレベルの囲碁プログラムの強さをお分かり頂けるだろう。

---

<sup>5</sup> <http://cell.scei.co.jp/>

<sup>6</sup> <http://www.cc.u-tokyo.ac.jp/service/ha8000/>

<sup>7</sup> <http://www.grappa.univ-lille3.fr/icga/event.php?id=41&lang=3>

<sup>8</sup> 加藤英樹、竹内郁雄、「Zen のクラスタ並列化」、第 14 回 ゲーム・プログラミング ワークショップ 2009 論文集, pp. 22-26, 情報処理学会, 2009.

<sup>9</sup> <http://jsb.cs.uec.ac.jp/~igo/>

<sup>10</sup> <http://www.nihonkiin.or.jp/player/htm/ki000343.htm>

<sup>11</sup> [http://jsb.cs.uec.ac.jp/~igo/result\\_ex/Aoba\\_Zen.sgf.html](http://jsb.cs.uec.ac.jp/~igo/result_ex/Aoba_Zen.sgf.html)