

ボードゲームの戦略プログラミングを題材とした Java 演習の大会運営サーバの効率化

Revision of a Contest Management Server for Java Programming Exercise with Board-Game Strategy

山田 航平† 富永 浩之‡
Kohei Yamada Hiroyuki Tominaga

1. はじめに

大学情報系では、C 言語などの入門的なプログラミング演習が必修とされる。続いて、応用的な演習では、C++/Java 言語などオブジェクト指向の導入、ソフトウェア開発手法の実践などが中心となる。また、データベースやネットワークなど、多様な情報系技術との融合も扱われる。さらに、特定分野の課題に対する問題解決手法としてのプログラミングが重視される。例えば、数理シミュレーションや、顧客管理などの具体的な情報システムが題材とされる。

しかし、提示した題材が学生の興味と程遠いと、具体的なイメージが湧きにくく、プログラミングの到達目標を描きにくい。また、成果物としてのソフトウェアへの愛着が得られない。そこで、知識情報処理の分野では、ゲーム戦略を題材とする演習が試みられている。これには、競争型学習の要素も盛り込まれ、動機付けへの効果が期待される。

2. 盤面ゲーム戦略のプログラミング演習

以上の背景により、先行研究[1]では、盤面ゲーム五五の戦略を題材とした Java プログラミング演習を提案している。また、演習支援のための支援環境 WinG を開発した。演習の問題設定は、ゲーム戦略を Java で実装することとし、総当り戦による順位付けとレポートによる評価を行う。二抜き連珠をベースに五五ゲームのルールを整備し、ゲーム実行のための基本ライブラリを実装した。戦略作成の基本方針を提示し、評価関数による戦略実装のためのサンプルコードを配布する。演習の形態として中間大会としての予備戦期間と最終大会を設ける。本研究では、2005 年度からの授業実践の成果を踏まえ[2]、2012 年度の教育実践に向け、演習方法の改善や支援システムの改良に取り組む[3]。

3. ローカル支援パッケージ WinG-LA

学生の躓きを減らし、全体的な戦略のレベルアップを図るため、戦略作成のローカル支援パッケージ WinG-LA(図 1)を提供する。WinG-LA は、4 つのモジュールから構成される。WinG-LA の各モジュールとゲームの実行ライブラリは、大会運営サーバから事前にダウンロードしておき、戦略検討の各種のサンプルと合わせて利用する。プログラミング自体は、既存のテキストエディタや、Eclipse などのプログラミング統合環境で行う。WinG-LA は、試行錯誤による戦略の検討や修正を支援する。対戦

実行モジュールは、実装した戦略の確認のため、人間または他の戦略プログラムとの対戦を行う。局面生成モジュールは、着手のデバッグのため、任意の初期局面を生成する。戦譜再現モジュールは、参考とすべき戦譜からの対戦を鑑賞する。

これら既存のモジュールに加え、新たに追加した着手確認モジュールは、指定された戦略と局面に対して、次の一手のみ実行する。実行結果について、適切な着手かどうか判定する。また、デバッグを効率よく行えるよう、複数の局面に対して一括して実行する。個々の結果を総合的に評価する。機能は、試行する戦略を指定する。複数の初期局面を一括して読み込む。各局面に対して、着手を期待する/しない枱を指定する。各局面に対して、次の一手のみを連続して実行する。期待する/しない着手と実際の着手を照合して結果を表示する。

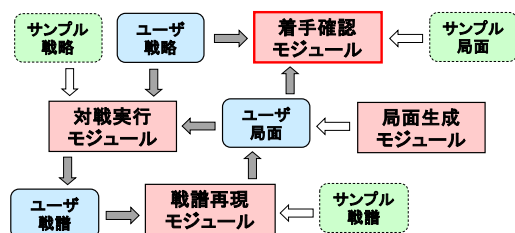


図 1 ローカル支援パッケージ WinG-LA

4. 大会運営サーバ WinG-CS

作成中の戦略にフィードバックをかけて、持続的に演習に取り組みさせるため、最終大会の締切までを予備戦期間とする(図 2)。予備戦期間中に提出された戦略は、サーバ上で他の戦略と対戦し、定期的の結果が更新され、順位が公開される。順位の推移を見て、自分の戦略を再検討し、状況に応じて戦略を修正していく。予備戦後に、提出した戦略の強さを総合的に判断し、最終大会の戦略を選択する。これらの戦略同士で総当り戦を行う。この結果から最終順位を決定し、成績に反映させる。このように、自分の戦略を常に評価する機会を設けることで、試行錯誤の繰返しを動機付ける。以上を支援するため、大会運営サーバ WinG-CS を運用する(図 3)。

2011 年度の大会実施に向けて、WinG-CS を全面的にリビルドし、新サーバに移行した。システムを、ユーザ管理部、戦略提出部、戦略管理部、全体結果部、予備対戦部、最終対戦部の各モジュールに整理した。Java での対戦実行は JDK1.7、内部処理は Ruby1.9.2 で実装し、DB は XML を用いる。

戦略提出ページでは、アップロードする戦略に、名前やコメントを付けられる。戦略が実行可能かどうかを学生に通知する。順位表示ページでは、全戦略の総当り戦の結果を勝率順に表示する。最強戦略による個人毎の順位も表示する。戦略管理ページでは、自分が提出した戦略について、戦績などを集約して表示する。対戦履歴ページでは、対戦ごとに、勝因や手数などを確認できる。指名対戦ページでは、任意の戦略を選んで対戦し、戦譜を再現できる。ただし、他人のソースコードを閲覧することはできない。最終結果ページでは、各自が選択した戦略で総当り戦を実施し、勝点で順位表示する。

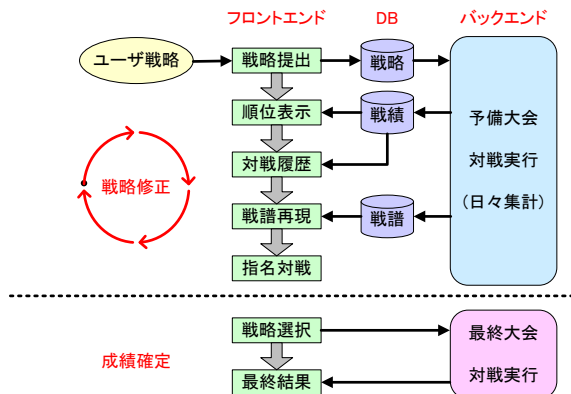


図 2 大会運営サーバ WinG-CS

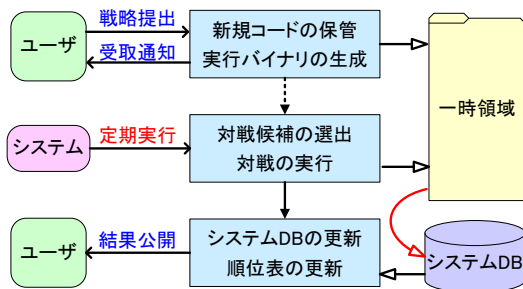


図 3 WinG-CS の内部処理

5. 2011 年度の演習実践と運営上の問題点

2011 年度は、情報環境コースの 3 年次の必修科目「情報環境実験 II」にて、開発したシステムを運用し、本演習を実践した。受講者は、35 名である。予備戦期間は約 5 週間で、強さの指標となる 3 つのダミー戦略を予め加えた。予備戦では、戦略の提出数が大幅に増加した。全体で 277 個の提出があり、対戦数は 4 万を超えた。1 人あたり平均 8 個であった。最大 46 個もの提出をする学生もいた。

現在のシステムでは、戦略のアップロード時には、受理のみとし、その場での対戦は行っていない。予備戦期間での対戦処理は、1 日ごとに新たに提出された戦略に対し、深夜に実施する。新しい戦略ごとに、既に提出されている全ての戦略と対戦させ、その時点での総当り戦を実現する。終盤では、対戦数が 200 以上となり、相当の時間を要した。また、対戦処理の間、システムをロックしているため、新規戦略の提出ができない時間が増えてしまった。

6. 対戦結果の即時通知のための処理の効率化

予備戦期間の教育効果を高めるには、戦略のアップロード時に対戦結果の即時通知が必要である。予備戦期間の順位は、暫定的なものであり、その都度更新されるものである。すなわち、戦略の強さのおおよその指標となればよい。その結果を踏まえ、すぐに戦略の修正に着手し、再アップロードまでのサイクルが短い方がよい。したがって、対戦処理の効率化を実現しなければならない。

現行のシステムの一斉対戦部では、新規提出リストにある戦略の数だけ、対戦実行を繰り返す。その間、新規提出と指名対戦の機能が停止する。戦略プログラムの中には、バグを含んで停止するものや、無限ループになるものもあり、トランザクショナルな一貫性が求められる。

しかし、完全な総当り戦の実行を求めるより、戦略評価の判断にほぼ十分な程度の実行を実現する方が望ましい。そこで、結果整合性(Eventual Consistency)に基づく分散処理を導入して、システムの可用性を高める。すなわち、対戦処理中も新たな戦略のアップロードを受け付け、常に結果の即時通知を目指す。具体的には、2 つのサーバで、フロントエンドとバックエンドの処理を分ける。同等の性能を有する 2 つのサーバを用意し、NFS でファイル共有を行う。再構成されたディレクトリに従って、アップロードされた戦略を管理するフロントエンドのデータと、総当り戦を行うバックエンドのデータを連携させる。

一方、少ない対戦数で戦略の妥当な評価を行える対戦方法も検討している。まず、試合数が異なっても比較できるように、勝点度を導入する。初期の弱い戦略による見かけ上の強さをなくすため、対戦相手の勝点度で重み付けする。勝点度でグレード分けし、各グレードから対戦相手を絞って、暫定的な強さを求める。

7. おわりに

盤面ゲーム戦略を題材とする応用的な Java 演習を提案し、支援環境 WinG を開発している。2011 年度の授業実践では、ダミー戦略を取り入れたため、提出数が大幅に増えた。しかし、対戦処理に時間がかかり、結果の迅速な反映が行えなかった。そこで、戦略の強さの暫定的な指標を即時通知できるように改良を目指す。そのために、結果整合性に基づく分散処理を検討する。また、勝点度を導入し、対戦数を間引く方法を検討する。

参考文献

- [1] 尾崎浩和, 他: "ボードゲーム戦略を題材とする問題解決型プログラミング演習支援 - 試行錯誤的な戦略作成の支援環境とサンプル提示 -", JSiSE 研究報告, Vol.22, No.4, pp.69-74, (2007).
- [2] 山田航平, 富永浩之: "ボードゲームの戦略プログラミングを題材とした Java 演習支援 - 演習実践と対戦結果の分析 -", 信学技報, Vol.111, No.141, pp.59-64, (2011).
- [3] 山田航平, 富永浩之: "ボードゲームの戦略プログラミングを題材とした Java 演習支援 - 着手確認モジュールの導入と大会支援サーバの GUI の改良 -", 信学技報, Vol.111, No.473, pp.19-24, (2012)