

特集 「ゲーム産業における人工知能」

リアルタイムサッカーシミュレーションゲームのAIシステムの一手法について

A Technique to Construct AI System for a Video Game on Real Time Football Simulation

安藤 毅
Takeshi Ando

株式会社セガゲームス
SEGA Games Co., Ltd.
Takeshi.Ando@sega.com, <http://sega.jp/>

Keywords: video game, football simulation, real time, AI.

1. はじめに

本稿は、ゲーム開発者向け技術交流会 CEDEC*1において著者が発表を行った講演『「サカつく」のサッカー試合 AI システム』の内容を再構成し、文章化したものである。

家庭向けゲーム機というスペックの制約されたハードウェアで、「試合シミュレータとしての納得感」、「映像としての面白さ」、さらには「処理負荷の軽さ」、「リアルタイム操作性」といったテーマに取り組んだ。その結果、サッカーの試合局面を離散的な状態遷移で表現する AI システムを開発するに至った。

本稿ではこの AI システムの設計コンセプト、実装内容、そして課題に対する効果の考察を行う。

2. 「サカつく」におけるサッカー試合

「サカつく」*2とは、株式会社セガゲームスより発売されている『プロサッカークラブをつくろう!』シリーズの略称である。

ユーザはプロサッカークラブの代表兼監督として、クラブ運営や試合采配などを行いながら世界最強チームを目指す。

ゲームの性質上、ユーザは選手やコーチの補強、チーム戦術の設定などという間接的な形でチームの強さに寄与することはできる一方、実際の試合においては監督として見守る立場となり、選手個々のプレイを操作したり、直接関与することはできない。選手達の動き、プレイ判断はすべて AI が行う。

ユーザが試合中に行う操作の技量が結果を左右する

「アクションゲーム」ではなく、基本的に結果が出るのを眺めるだけの「リアルタイムサッカー試合シミュレータ」として実装されているのが特徴である。

2-1 求められる要件

§1 試合シミュレータとしての「納得感」

「サカつく」におけるサッカー試合のシーンは、ユーザがいかにかの確にクラブマネジメントをしたか、その成否を確認する場である。したがって、選手やコーチの能力が試合の結果に反映されるよう、「強いチームの勝率が高い」ことをきっちり担保するシステムでなければならない。

その一方で、「サカつく」ではユーザに“まるで本物のサッカーの試合を観戦しているような気持ちで”試合シーンを見てもらえることを目指している。サッカーは 2-1、3-2 といった比較的ロスコアかつ僅差で決着がつくことの多いスポーツである。強いチームが必ず大差をつけて勝つようなシステムではなく、このような「サッカーらしいスコア」に結果が落ち着くような調整も重要だと考えている。

§2 映像としての面白さ

上述のとおり、サッカーの試合を実際に観戦しているようなシミュレータを目指すにあたって、フィールド上で発生する局面も可能な限り本物に近い形で再現することを試みている。人間の形をした選手達が AI によるプレイ判断を行い、アニメーション動作を行いながら一つのプレイを行う様子を CG で再現し、本物のサッカーをほうふつとさせるような臨場感を表現している。

§3 処理負荷の軽さ

観戦して面白い試合シーンを制作する一方で、手早く試合結果のみを知りたいという要望に応える必要がある。「サカつく」というゲームの目的が長期的視野でクラブチームを強化していくことにあり、重要でない試合はスキップしてゲームを先に進めたいユーザが多く存在

*1 <http://cedec.cesa.or.jp/>

*2 <http://www.sakatsuku.com/>

するのは自然なことである。

したがって「高速で試合結果を計算するエンジン」が必要となる。実際の試合観戦とは別の単純化したシステムを利用する方法も考えられるが、二つのシステムで勝敗結果に有利不利が出ないように調整するコストを考慮すると、むしろ観戦可能な試合システムの高速化を図るほうが現実的である。

なお、「サカつく」は家庭用ゲーム機という固有のハードウェア上で動作するゲームなので、ハードウェアのスペック増強による高速化という手法を使えないことも付記しておきたい。

2.2 シリーズにおけるシステムの変遷

要件を満たすサッカー試合システムの実現に向け、「サカつく」シリーズではいくつかの方式が採用されてきた。

§1 完全オーサリング方式

シリーズ1作目となる「J.LEAGUE プロサッカークラブをつくろう！」(1996)^{*3}から「J.LEAGUE プロサッカークラブをつくろう! 04」(2004)^{*4}までの各タイトルで採用された方式である。

複数のユーザやボールの動き、カメラワークを含めて3DCG制作ツール上でオーサリングした試合シーンを多数用意した「シーンデータベース」を構築し、プレイ結果に応じて選択再生する。

図1に完全オーサリング方式のシステム概念図を示す。試合AIは時間進行とともに試合局面を更新する。そして局面ごとに3DCGで再生すべきシーン(例えばゴールなど)かどうかを判断する。再生すべきと判断した場合、そのときの局面を表すパラメータを使ってシーンデータベースから最も状況に近いシーンを選択、再生する。シーンを表す局面パラメータは(1)場面のタイプ(シュート場面、ファウル場面など)、(2)結果タイプ(シュートが決まる、外れるなど)、(3)登場人数などがある。

この方式のメリットは、選手のアニメーションやカメ

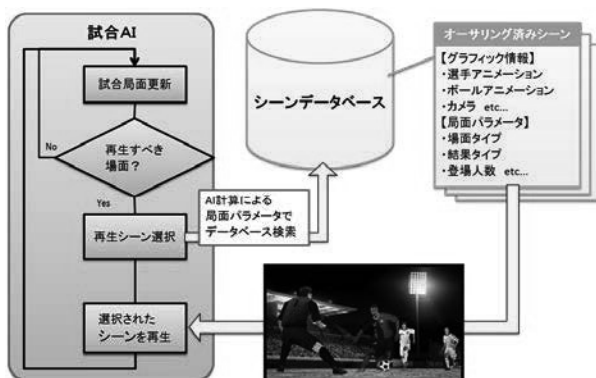


図1 完全オーサリング方式による試合AIシステム

ラワークをグラフィックデザイナーが適切に選択して配置できるため、一つ一つの試合シーンを高いクオリティで制作できることである。また、試合AIシステムとグラフィックシステムが独立したモジュールなので、それぞれ最適な設計を行いやすい。

ただ、長期間にわたってゲームを遊び、何百試合と観戦していると、ユーザは「いつも同じシーンばかりが再生される」、「何度も見たシーンなのでその後の展開が読めてしまう」という不満を抱くようになる。シリーズを重ねるにつれ、シーンデータベースのバリエーション増加、シーン選択アルゴリズムの改良で対応していたが、制作コストの肥大化が課題となってきた。

§2 完全リアルタイム方式

「プロサッカークラブをつくろう! ヨーロッパチャンピオンシップ」(2006)^{*5}などの作品で採用された方式である。

サッカー選手一人の動作を要素分解して構築した「選手アニメーションデータベース」を用意し、各選手が独立したエージェントとして状況に応じた自由な動作をできるようにした。いわゆる「アクションサッカーゲーム」と同様のアプローチである。

図2に完全リアルタイム方式のシステム概念図を示す。試合AIは、両チーム合わせて22名の選手やボールといった各エンティティの状態パラメータを取得して状況判断を行い、22名の選手に動作を指令する。

メリットは、マルチエージェントであるがゆえの自由度の高さである。フィールド上に現れる局面が同じになる確率は限りなく小さく、先が読めない本物のサッカーと同様の醍醐味を演出することができた。

その一方で、システムの調整が極めて困難なものとなったことがデメリットである。

このシステムでは、まず選手やチームの能力を、例えば移動スピードや蹴りだされるボールのスピードといった物理的パラメータに反映させた。そしてその結果の積み重ねで勝敗をコントロールしている。間接的なパラ

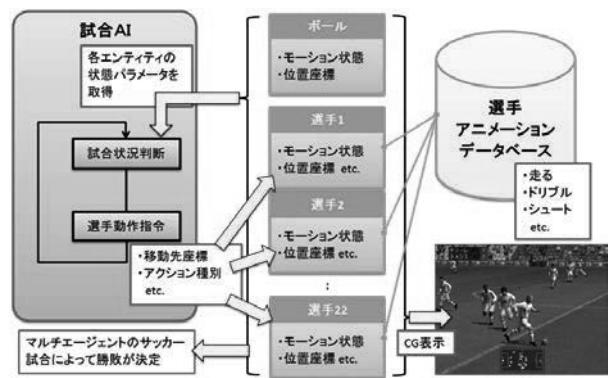


図2 完全リアルタイム方式による試合AIシステム

*3 <http://www.sakatsuku.com/history/01/>

*4 <http://www.sakatsuku.com/4/>

*5 <http://www.sakatsuku.com/sakaEU/>

メータ調整による結果コントロールなので「特定の能力パラメータの影響力が高い」, 「わずかな能力差が大差となって結果に反映される」などの問題が顕著となった。これら問題を克服するバランス調整に多大なコストがかかり, またシステムを知り尽くした特定の制作スタッフしか調整できなかった。

もう一つのデメリットとして, 処理速度の向上の限界があげられる。実際にCG再生するのに近いアニメーション計算などを行わないと局面計算が行えないため, 「試合シーンをスキップして結果だけ知りたい」ユーザへの要望に十分に答えられなかった。

§3 プレイセット方式

「サカつく DS タッチ and ダイレクト」(2008)^{*6}, 「サカつく DS ワールドチャレンジ 2010」(2010)^{*7}の二つのタイトルで採用した。本稿では, 次章でこのプレイセット方式の処理フローについて詳説する。

ここで『プレイセット』という新たな概念を導入しているが, その概念が生まれた背景から説明する。

サッカーの試合において, ある瞬間にボールのあるエリアでプレイに関与する選手は一人ないし多くても数人である。ボールに関与する選手の動作はパス, ドリブル, シュートなど多岐にわたり, ディフェンス側選手と絡んだ組合せ動作となることも多い。一方それ以外の周辺選手は, 将来自分のところにボールが来たときに備えて, 移動や立止まりといった基本動作をしていることがほとんどである。

この選手達の動作バリエーションの違いに本システムは着目した。ボール周辺でプレイに関与する数人の選手の動作を, フィールド上の小さなエリアの“局面シーン”として多数制作し, ここにサッカー独自のさまざまなアニメーションを盛り込んだ。他方, 周辺選手については走る, 立止まるといった選手1名ごとの基本動作アニメーションのみを用意し, 明確に区別している。

プレイセットとは, ボール周辺の同一局面, 同一プレイ選択から分岐し得る“局面シーン”をグループ化し, これにおおのこの分岐条件を記述した“結果判定式”を結び付けたデータである。プレイセットデータベースでは, (1) 局面のタイプ(ドリブル選手が1対1で守備選手と向き合う, こぼれ球に対して両チームの選手が競り合う, など), (2) プレイ選択(パス, ドリブル, シュートなど)という局面パラメータをキーとして, 対応するプレイセットを検索できるよう情報が格納されている。

図3にプレイセット方式のシステム概念図を示す。試合AIは現在の局面パラメータをキーとしてプレイセットデータベースを検索し, 選手が実行したいプレイ(=プレイセット)を選択する。プレイセットには結果判定式が結び付けられており, 選手の能力パラメータを入力

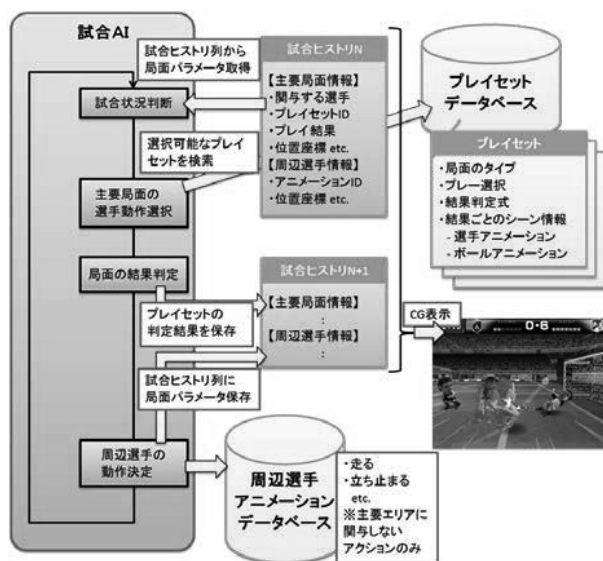


図3 プレイセット方式による試合AIシステム

として計算を行い, プレイ結果を確定させる。同時に, 主要局面に関与しない周辺選手については, 周辺選手アニメーションデータベースに基づいて目標移動地点などの簡易なAI計算を行う。

AI計算によって得られたこれらすべての情報は, 試合ヒストリという構造体に格納され, 時系列に「試合ヒストリバッファ」に保存される。試合AIシステムは, 保存された試合ヒストリ構造体を入力パラメータとして次の局面計算を行い, 得られた結果を再び試合ヒストリバッファに保存する。こうした一連の処理の結果, 1試合全体で起こるすべての出来事は, 試合ヒストリの配列として表現されることとなる。また, バッファに保存された試合ヒストリは, CG表示のために参照される情報としても機能する。試合ヒストリバッファのもつ機能を図4に示す。



図4 試合ヒストリバッファの機能

プレイセット方式は, 完全オーサリング方式と完全リアルタイム方式のメリット・デメリットを分析し, それぞれの“良いところ取り”を試みた折衷方式である。完全オーサリング方式のもつシーンのクオリティをボール周辺に限定して担保する一方で, 周辺選手はシンプルなアニメーションによるAI動作とすることでオーサリングの工数を抑えた。同時に, 細かいシーンをつなぎ合わせて試合が展開するため, ユーザにとって先が読みやすいということは起こりにくい。

*6 <http://www.sakatsuku.com/ds/index.html>
 *7 <http://www.sakatsuku.com/ds2010/index.html>

§4 ハイブリッド方式

「JLEAGUE プロサッカークラブをつくろう! 6 Pride of J」(2009)^{*8}などのタイトルで採用したシステムである。プレイセット方式による試合 AI システムが試合結果のコントロール、実行速度ともに実用的であったため、これを完全オーサリング方式の試合 AI 部分に当てはめたものである。

AI が生成する試合履歴情報は CG 画面生成には直接利用されないため、プレイセットデータベースには AI 計算に必要な結果判定式などの情報だけを残し、グラフィックデータをそぎ落としている。

また、試合履歴情報から決定的なシーンかどうかの判定、および結果と類似したオーサリング済みシーンの選択を行う「グラフィック再生モジュール」が存在する。

実際の CG 画面においては、通常時は 2D グラフィックスによる簡易画面で進行し、シーン再生すべきと判定されたときに、選択されたオーサリング済み 3D シーンの再生へと切り替わる。図 5 にハイブリッド方式のシステム概念図を示す。図中の「試合 AI」モジュールは、図 3 で示したプレイセット方式の「試合 AI」と同じ仕組みである。

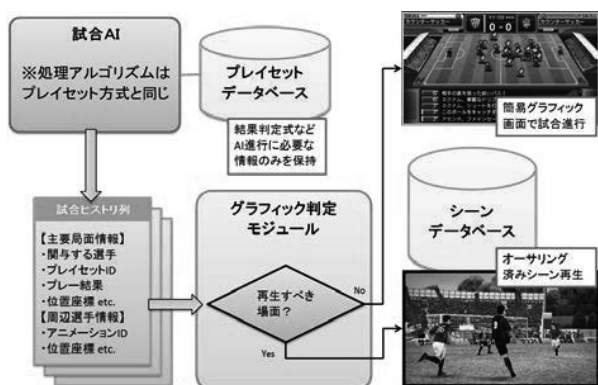


図5 ハイブリッド方式による試合 AI システム

3. プレイセット方式の試合 AI

本章では、前章で取り上げた四つのシステムのうち、プレイセット方式の試合 AI システムについてさらに詳細を解説する。

まず、本システムを開発するきっかけとなったゲームタイトル「サカつく DS」シリーズ固有の課題について触れ、それからプレイセットの概念について掘り下げて解説、試合 AI の処理シーケンスおよび CG シーン生成処理を解説する。

3.1 「サカつく DS」固有の要件

「サカつく DS」シリーズ固有の要件として、「ユーザ

がほぼリアルタイムに試合に介入できる」ゲームデザインであることがあげられる。具体的には、(1) 選手のフォーメーション変更、(2) 戦術指示、(3) 監督スキル(一定時間だけ選手の能力パラメータを上げるなど、ゲーム固有の要素)などである。

これらのユーザ介入を、できるだけ素早く試合結果にフィードバックすることが求められる。ただ、アクションゲームのように 0.1 秒未満で動作に反映するほどのレスポンスの速さは求められていない。

CG 画面では、図 6 のように選手は 2.5 頭身のデフォルメされたキャラクターとして表示される。「ニンテンドー DS」^{*9}というグラフィック表示機能に限界のあるゲーム機向けにソフトウェア開発していたため、フォトリアリスティックな CG というよりはコミカルなアニメーションで楽しませるコンセプトである。

こうした要件定義から、ボールに関与する選手の一連の動作が最低限自然に見える単位、すなわちパス・ドリブルなど一つのプレイが完了するまでを“局面シーン”としてオーサリングした。これら“局面シーン”をつなぎ合わせてフィールド上で連続再生することで、見栄えの良さとレスポンスの適度な速さの両立を目指すアプローチをとった。



図6 「サカつく DS」シリーズのサッカー試合画面

3.2 プレイセットについて

前述したとおり、プレイセットとは、同一局面、同一プレイ選択から分岐し得る“局面シーン”のグループと、分岐条件の“結果判定式”から構成されるデータである。

プレイセットのシーングループの例を図 7 に示す。これは左側の攻撃側選手が、右側の守備側選手をドリブルで抜き去ろうとしているシーンである。①、②までは共通のアニメーションで進行するが、後半の③、④、⑤のアニメーションが 4 種類に分岐している。

(1) 大成功：攻撃側選手はドリブルで抜き去ることに成功し、守備側選手は地面に倒れてしまう。

*8 <http://www.sakatsuku.com/6/>

*9 <https://www.nintendo.co.jp/ds/>

- (2) 成功：攻撃側選手はドリブルで抜き去ることに成功し、守備側選手は立った状態で終了する。
- (3) 失敗：守備側選手にボールを奪われ、攻撃側選手は立った状態で終了する。
- (4) 大失敗：守備側選手にボールを奪われ、攻撃側選手は地面に倒れてしまう。

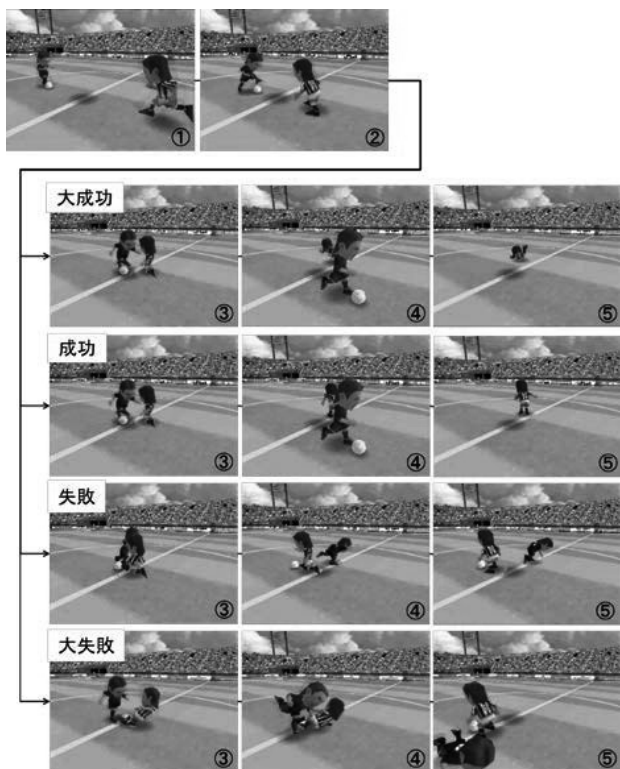


図7 プレイセットに含まれるシーングループの例

これらのシーンのうち結果としてどれを再生するかを決めるのが結果判定式である。結果判定式は、シーンに登場する選手の能力パラメータを入力とし、各シーンが選択再生される確率を出力とする。

プレイセット P に含まれる n 個のシーン集合を $S_P = \{S_{P1}, S_{P2}, \dots, S_{Pn}\}$ 、攻撃側・守備側の m 種類ある選手能力パラメータのセットをそれぞれ $A_O = \{A_{O1}, A_{O2}, \dots, A_{Om}\}$ 、 $A_D = \{A_{D1}, A_{D2}, \dots, A_{Dm}\}$ と表現する。結果判定式は、

各シーン S_{P_i} ($i = 1, 2, \dots, n$) の発生確率 $p(S_{P_i})$ を以下の式で表現するような関数 f_{P_i} ($i = 1, 2, \dots, n$) の集合である。

$$p(S_{P_i}) = f_{P_i}(A_O, A_D) \quad (i = 1, 2, \dots, n)$$

$$0 \leq p(S_{P_i}) \leq 1, \sum_{i=1}^n p(S_{P_i}) = 1$$

通常、関数 f_{P_i} は選手の能力パラメータ A_O, A_D の中から、プレイセットの表現するプレイに関連の高い能力パラメータが結果に影響するよう設定する。例えば図7のような「ドリブルで抜き去る」シーンであれば、攻撃側選手の「ドリブルのうまさ」と守備側選手の「ボール奪取のうまさ」という能力パラメータが該当する。一般に、攻撃側選手の能力パラメータが高いほど「成功」、「大成功」の発生確率が高く、逆に守備側選手の能力パラメータが高いほど「失敗」、「大失敗」の発生確率が高くなるように関数を調整する。

「サクつくDS」シリーズのシステムでは問題を単純化するために、一つの局面シーンに関与する選手は最大2人、分岐のパターンは「大成功」、「成功」、「失敗」、「大失敗」の4パターンに限定して設計を行った。

3.3 試合AIの処理シーケンス

図3で示したとおり、試合AIの処理は四つのフェーズで構成されている。

§1 試合状況判断

まず、直前のフィールド上の状態パラメータから試合状況を決定する。過去のフィールド上の状態は試合ヒストリ構造体として時系列に保存されており、ここに格納された状態パラメータから主要局面のタイプを決定する。

局面のタイプは主に二つの要素で決定される。

- (1) 攻撃側選手とボールの関係 (ボールを保持してドリブルしているのか、パスを受けようとしているところかなど)。直前の局面の終了状態から自動的に決定される。
- (2) 攻撃側選手と守備側選手の関係 (守備側選手が攻撃側選手に正対しているか、横から対峙しているか

表1 主要局面のタイプ (抜粋)

局面タイプ	フリー	1対1	プレス	背負い	GK1対1
模式図					
説明	ボールホルダーがドリブルしている。プレイに関与するディフェンス選手はいない。	ボールホルダーがドリブルしている。ディフェンス選手がボールホルダーの正面から対峙している。	ボールホルダーがドリブルしている。ディフェンス選手がボールホルダーの横からディフェンスしに行く。	ボールホルダーがボールをキープして停止。ディフェンス選手がボールホルダーの背後にはりつく。	ボールホルダーがドリブルしている。ゴールキーパーがボールホルダーの正面から対峙している。
関与する選手	ボールホルダー	ボールホルダー ディフェンス選手	ボールホルダー ディフェンス選手	ボールホルダー ディフェンス選手	ボールホルダー ゴールキーパー

など)。これは主に守備側のチーム思考によって決定される。フォーメーション設定や、守備の戦術（前線から激しくボールを奪いに行くか、自陣に引いて守るかなど）によって、どの選手が主要局面に関与する守備側選手になるかを決定する。そしてプレイセットデータベースに用意されている局面タイプの中から、攻撃側選手との位置関係によって局面タイプを選択する。

本システムで設定した局面タイプの一部を抜粋したものを表 1 に示す。

§ 2 主要局面の選手動作選択

局面タイプが決定したら、プレイセットデータベースに登録されているプレイセットから、攻撃側選手がとり得るプレイの選択肢をリスト化する。

プレイの選択肢は、(1) パス・ドリブル・シュートなどのプレイ種別と、(2) プレイ種別ごとの固有パラメータによって記述される。固有パラメータはパスであればパス先選手、ドリブルであれば進行方向などである。行いたいプレイに対応するプレイセットデータが存在するものだけが選択肢としてリストアップされる。

プレイの選択肢ごとに、成功した場合のメリットや成功確率などを考慮して「評価点」を付与する。そしてこの評価点を重み付けとしたルーレット選択アルゴリズムにより、プレイが選択される。

§ 3 局面の結果判定

試合の局面タイプ、および攻撃側選手が行いたいプレイを決定したことで、今回使用するプレイセットが決定した。次はプレイセットに登録されているシーングループの中から再生するシーンを決定する。

各シーンの選択再生確率が、関与する選手の能力パラメータの関数として表現されることはすでに述べた。「サカつく DS」シリーズでは、「確率パラメータ」 \mathcal{P}_p という一つのパラメータを計算し、この値に応じたテーブル参照で各シーン S_{P_i} の発生確率 $p(S_{P_i})$ を決定するアプローチを採っている。

\mathcal{P}_p は攻撃側選手と守備側選手の能力値をパラメータ入力とし、スカラ値を出力する関数である。攻撃側選手の能力パラメータが高く、プレイの成功確率が高いときほど大きな値となるように設定する。

確率テーブルには \mathcal{P}_p の値レンジごとに「大成功」、「成功」、「失敗」、「大失敗」の選択再生確率 $p(S_{P_i})$ を設定

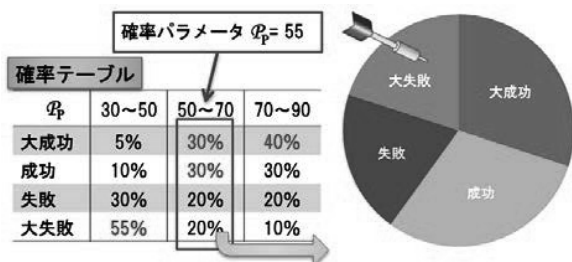


図 8 確率パラメータとテーブルによるプレイセット結果決定

してある。この確率を重み付けとしたルーレット選択アルゴリズムにより、再生するシーンが決定される。処理の流れを模式化したものを図 8 に示す。

§ 4 周辺選手の動作決定

主要局面で再生するシーンが決定されると、次にこのシーンを再生するタイミング調整を行う。シーン再生開始時点の選手・ボール位置座標に移動するまでの所要時間を計算し、その最大値の分だけシーン再生開始までの待ち時間を挿入する。待ち時間の間、選手は周辺選手と同様のアニメーションで移動を行う。

シーン再生までの待ち時間とシーンのアニメーション長から、直前の局面から今回の局面までの所要時間が決定する。そして、この間シーンに関与しない選手の動きを決定する。各選手には、チームのフォーメーションや役割に応じてフィールド上の目標地点が与えられる。移動限界速度、使用可能なアニメーションの制約のもとに、目標移動地点、使用するアニメーションを時系列に規定したコマンド列を各選手に対して生成する。周辺選手のコマンド列の例を図 9 に示す。



図 9 周辺選手のコマンド列の例

選手だけでなく、ボールにもコマンドが生成される。シーンのアニメーションにボールが含まれない時間は、前のシーンでのアニメーション終了地点から、次のアニメーションシーンの開始地点をつなぐようにボールの軌道を生成するコマンド列が与えられる。

3.4 CG シーン生成処理

試合 AI の計算結果は、試合ヒストリ構造体に格納される。これが次の試合局面の計算のための状態パラメータとして利用されると同時に、CG グラフィックのシーン生成のパラメータとしても利用される。

前節でも述べたとおり、ボールが主要局面のシーンアニメーションに含まれない時間の軌道は、次のアニメーションシーンが決定してはじめて生成できる。そこで本システムではまず試合ヒストリを試合 AI により先行計算し、CG 画面生成に必要な試合ヒストリが試合ヒストリバッファに蓄積された時点でアニメーション再生を行う。結果的に、ユーザがゲーム画面で目にしているのは、実際には試合 AI の計算結果を少し遅れて追いかけるように CG 表示したものとなる。このことを模式的に表現したものが図 10 である。

これは「サカつく DS」シリーズがアクションゲーム並みのレスポンスを求められていないために採用し得る

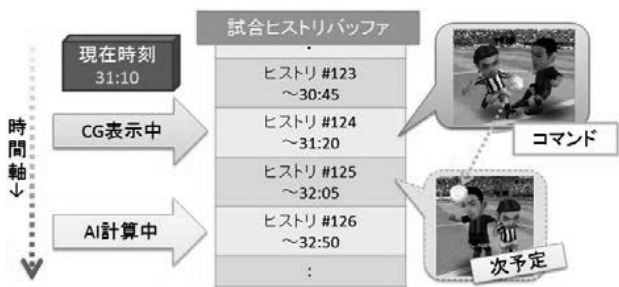


図 10 AI 計算と CG 表示のタイミング

アプローチである。ユーザがチームに指示を与えると、その瞬間以降の試合 AI 計算にユーザ介入の効果を反映させる。CG 画面上でユーザがその効果を目にするのは少し遅れてのことになるが、それで問題ないゲームデザインとしている。

主要局面のシーンで動作する選手アニメーションは必ずしも同じ長さではない。また次シーン再生までの間に待ち時間が発生することもある。CG シーン生成処理では、各時刻ごとに選手やボールといった各エンティティが (1) 主要局面シーンに所属しているか否か、(2) どの試合履歴情報に基づいて動作しているか、という情報を管理している。主要局面以外の選手の動作は、試合履歴上に生成されたコマンド列によって規定される。

エンティティごとの状態管理が時系列で進行していく様子を模式的に示したものが図 11 である。試合履歴構造体には、試合開始から通し番号が振られている。この図では左から右に時刻 t が進行していくにつれ、各エンティティの動作を規定する試合履歴情報が番号 #123 から #125 にかけて遷移しながら CG 画面が生成されていくことを表している。

ある時刻において各エンティティの動作を規定する試合履歴が同時に切り替わるとは限らない。例えば図 11 において、周辺選手が試合履歴 #123 から #124 に基づく動作に切り替わった瞬間でも、選手 B は引き続き試合履歴 #123 の主要局面アニメーション再生を続けていることが見て取れる。このため CG 表示エンジンは、エンティティごとに独立に状態を管理しながら、アニメーション再生やコマンド列に基づいた動作を行う。

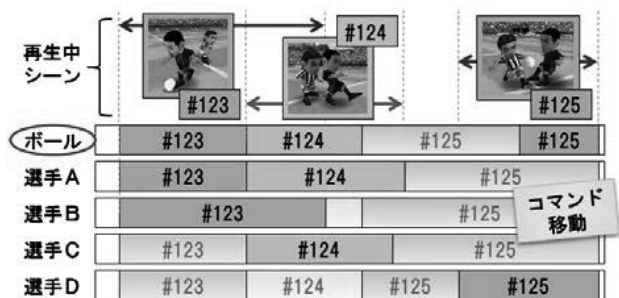


図 11 CG 表示エンジンにおけるエンティティの状態管理

4. 課題に対する効果の考察

前章で、プレイセット方式の試合 AI システムについて解説した。このシステムは過去作品で採用した完全オーサリング方式、完全リアルタイム方式のそれぞれのメリットを取り入れ、デメリットを克服するために考案された方式である。

表 2 に、「サカつく」シリーズの試合 AI における課題への効果の高さを、2・2 節であげた四つの方式ごとに示す。以下、それぞれの課題について各方式の効果を考察する。加えて、プレイセット方式の採用により生まれたメリットについても触れる。

表 2 各試合 AI 方式の課題への効果

方式		①	②	③	④
CG の見栄え	主要局面	◎	△	○	◎
	全体	◎	○	△	◎
展開の多様性		△	◎	○	○
シーン制作工数		×	○	○	×
結果コントロール		◎	×	○	○
AI 処理実行速度		◎	×	○	○
操作レスポンス		×	◎	△	△

① 完全オーサリング方式、② 完全リアルタイム方式、
③ プレイセット方式、④ ハイブリッド方式

4.1 CG 画面の見栄え

ここでいう“見栄え”の評価は、AI による自動生成で往々にして発生しがちな不自然な動きがなく、ユーザが試合を観戦する楽しさを阻害しないクオリティであることを基準としている。

完全オーサリング方式およびハイブリッド方式では、各シーンのエンティティ配置やアニメーションをすべて人力で制作するため、見栄えの良さは制作者自身が担保できる。

完全リアルタイム方式では、マルチエージェントであるがゆえに起こりがちな不自然な連携、時系列の反復動作など、サッカー試合として違和感をユーザに抱かせる場面が少なからず発生した。製品化に向けて何とか対処は行ったが、多大な労力を要した。

プレイセット方式では、ボール周辺の局面に関してのみ人手によるシーン再生として最低限の見栄えを担保している。その一方で周辺選手についてはユーザが目立たないものと割り切ったため、細部のグラフィックの質が高いものとはいえない。

4.2 展開の多様性

CG 画面上で同じ展開となることなく、ユーザが先の展開を読めないような演出ができていくかどうかの尺度である。これは各エンティティの位置や状態が独立してリアルタイム管理されている完全リアルタイム方式が最

も優れているといえるだろう。

プレイセット方式も、局面ごとのシーンは同じものが使われたとしても、フィールド上での座標・向きが同じとなることはまずなく、またシーン長も短いため先まで見通せることは少ない。

完全オーサリング方式は、同じ局面から開始して結果が異なるシーンを多数用意することでこの問題に対処しようとしたが、制作工数やシーン選択アルゴリズムの改良など、課題を残したままの運用であった。

ハイブリッド方式は、簡易画面がプレイセット方式、シーン画面が完全オーサリング方式と同じ仕組みなので、中間的な位置付けといえる。

4.3 シーン制作工数

シーン制作の工数はすでに述べたとおり完全オーサリング方式での負担が非常に大きいものであり、シリーズを重ねて得られたシーンデータベースの蓄積でまかなう比率が大きい。ハイブリッド方式も同様である。

完全リアルタイム方式では、一人の選手に関するアニメーションを一通り用意すればよいので、グラフィックデータの準備工数は最も少ないといえる。

プレイセット方式で用意した局面シーンの個数は、完全オーサリング方式の約 10 分の 1、完全リアルタイム方式の選手アニメーション数とほぼ同程度である。局面シーンが複数選手やボールを含むため、データ制作工数は完全リアルタイム方式よりやや大きい。要素アニメーションを使い回すことで工数を抑制している。

4.4 結果コントロール

「強いチームが勝つ」、「サッカーらしいスコアを実現する」という観点では、試合結果生成とグラフィック生成を全く独立のアルゴリズムで制作、調整できる完全オーサリング方式が最も調整しやすい。実際このメリットを生かして試合 AI を入れ替えたのがハイブリッド方式である。

これに対して、試合結果生成とグラフィック生成を同じアルゴリズムで実現したのがプレイセット方式である。サッカーというスポーツが究極的にはボール周辺で発生する局面の連続であり、おのおのの局面での選手同士の小さな勝負の積み重ねで試合結果が決まるというアプローチで設計を行った結果、ある程度の調整を行えばバランスが取れるシステムに仕上がったと考えている。

一方、完全リアルタイム方式では、選手やチームの能力を物理パラメータに反映させることで間接的に試合結果をコントロールする方式であったため、このシステムを知り尽くしたスタッフがパラメータ調整に膨大な時間を費やして辛うじてバランスをとることとなった。

4.5 AI 処理実行速度

試合 AI 処理の実行速度についても、完全オーサリン

グ方式であればいかようにも調整し得るため、最も対処がしやすい。

プレイセット方式と完全リアルタイム方式はどちらもグラフィック表示と結び付いた試合 AI であるため、局面ごとの計算速度を劇的に向上させることは難しい。ただ、これら 2 方式の決定的な違いとして、プレイセット方式が「試合の局面を離散的な状態遷移で表現していること」があげられる。

プレイセット方式では、試合局面は選手のプレイ単位で遷移するものと定義し、プレイに関与しない周辺選手の動きもコマンド列に単純化してしまうことで、局面の変化する瞬間の発生頻度を少なくした。完全リアルタイム方式がマルチエージェントを管理するシステムとして瞬間ごとに判断を行い、計算を省略できるタイミングが存在しないのと対照的である。

「サカつく DS」シリーズは「ニンテンドー DS」という処理性能の限られたハードウェアにおいて、1 回の局面計算に要する時間は 1/30 秒弱と、画面のリフレッシュレート 1 回分とほぼ同等であった。このため、1~2 秒に 1 回程度、瞬間的な処理落ちが発生するが、それを人間の目で知覚することはまずない。

その一方、完全リアルタイム方式では、CG 画面生成処理の省略だけでは十分な速度向上ができず、試合画面をスキップして結果を見るモードと試合を見るモードとで、1 試合に要する時間がほとんど変わらない結果となった。

4.6 操作レスポンス

「サカつく」シリーズはユーザが監督として間接的に試合介入する形式のゲームなので、ユーザ操作に対するレスポンスの遅れがあまり問題にならない。

ユーザが選手の動作を操作するようなアクションゲームを制作するのならば完全リアルタイム方式以外に選択肢はないと思われるが、「サカつく DS」のようなゲームデザインであれば、プレイセット方式で実現可能なレスポンス速度で十分である。

完全オーサリング方式を採用した作品では、そもそもユーザ介入の方法がきわめて限定されたゲームデザインで、レスポンスそのものを考慮していない。

4.7 デバッグ・調整におけるプレイセット方式の利点

プレイセット方式の採用により、試合 AI のデバッグやパラメータ調整でも利点があったので言及したい。

プレイセット方式のシステムでは、試合局面の状態パラメータがすべて試合ヒストリ構造体上に集約されており、CG 生成にも直接パラメータとして使用される。このため、試合ヒストリを退避・復帰させることで特定の状況を繰り返し再現でき、バグの再現が容易となった。

試合 AI の実行速度が速いことで、何百試合、何千試合とテスト試合を自動実行させ、試合結果の各種統計を

取りやすくなったこともメリットである。ゲームバランスをとるパラメータのチューニングにおいて、この統計データが大いに役立った。

4.8 ま と め

4種類の方式について各種課題に対する効果を考察してきた。プレイセット方式は過去の2方式をすべての面において凌駕する完全なシステムではない。全体的な見方については完全オーサリング方式が優位であるし、ユーザ操作に対するレスポンスの速さを求めるなら完全リアルタイム方式以外は考えづらい。

プレイセット方式は、「サカつく」シリーズのゲーム特性を踏まえれば、飛び抜けた利点もないが欠点の少ない、いわば“総合的な最適解”のシステムといえるだろう。

5. お わ り に

本稿は、家庭用ゲームの制作現場において、実際に目の前で発生していた課題に対応、克服しようとした事例を論文形式にまとめたものである。過去の研究論文などを背景として生まれたものではないため、実質的に参考文献といえるものは一つもあげられないことはご容赦いただきたい。実際のところ、開発過程において他のゲームを遊んでみたり、講演会を聴講してヒントを探したことはあっても、学術論文などからの情報収集はほとんど行わなかった。唯一あげられるとすれば、ロボカップの書籍[秋山 06]だろうか。目を通した結果、「現実世界のシミュレーションとして環境センシングをも制約条件とする」ロボカップのアプローチと、「エンタテインメントを成立させるためならばすべてを把握・制御できる AI を導入しても構わない」コンピュータゲームではアプローチが異なるということを知り、応用を諦めたしだいである。

また、研究発表を前提とした業務ではなかったため、4章における効果の考察については客観的評価データのない、著者の主観に基づくものとなっている点についてもご容赦いただきたい。

近年ゲーム AI は高度化してきており、最先端の研究に触れながら製品開発にあたる機会も多くなってきた。ゲーム開発者としては、そうした諸先生方の研究成果を一方的に享受するばかりでなく、ゲーム分野において何らかの成果が得られた際にはフィードバックしていく姿勢も求められるだろう。

願わくば、本稿が少しでもお読みいただいた方々の刺激となり、今後の人工知能の発展に向けたヒントとなれば幸いです。

謝 辞

「サカつく DS」開発チームメンバの尽力なくしては本システム開発はあり得ませんでした。この場をお借りして感謝申し上げます。

また今回、本誌編集委員の三宅陽一郎氏から原稿執筆のお話をいただいたときには6年前の講演内容が何かの役に立つのだろうかかと不安も覚えました。改めてまとまった形で本論文を残す機会を与えていただいたことは、ただただ感謝のひとつです。

◇ 参 考 文 献 ◇

[秋山 06] 秋山英久:ロボカップサッカーシミュレーション 2Dリーグ必勝ガイド, 秀和システム (2006)

2017年1月20日 受理

著 者 紹 介



安藤 毅

1995年東京大学工学部機械情報工学科卒業, 1997年同大学院工学系研究科情報工学専攻修士課程修了。同年, 株式会社セガ・エンタープライゼス(現株式会社セガゲームス)入社。遊戯施設向けのアーケードゲーム制作などに携わり, 2012年に株式会社セガネットワークス(現株式会社セガゲームス セガネットワークス カンパニー)に転籍。主にスマートフォン向けのゲームアプリ開発基盤の整備に携わる。人工知能分野からは「サカつく」シリーズ開発以降離れていたが, 第三次 AI ブームに乗る形で, ディープラーニングなど最新技術研究を勉強中。