

エージェント・アーキテクチャに基づくキャラクターA Iの実装 — クロムハウন্ズのキャラクターA I を例として —

三宅 陽一郎[†] 横山 貴規[†] 北崎 雄之[†]

[†]株式会社 フロム・ソフトウェア 〒151-0073 東京都渋谷区笹塚 2-26-2

E-mail: [†]y_miyake@fromsoftware.co.jp

あらまし キャラクターA Iのエージェント・アーキテクチャによる実装をクロムハウন্ズの実例に沿って説明する。
キーワード エージェント・アーキテクチャ, マルチ・エージェント, キャラクターAI, デザインパターン, ゴール指向,
階層型プランニング, ロバスト

Developing Character AI Based on Agent Architecture in Digital Games — Character AI in Chromehounds —

Youichiro Miyake[†] Takanori Yokoyama[†] Takeyuki Kitazaki[†]

[†]FromSoftware, Inc. 2-26-2 Sasazuka, Shibuya-ku, Tokyo, 151-0073 Japan

E-mail: [†]y_miyake@fromsoftware.co.jp

Abstract A process of developing Character AI based on agent architecture is explained using Chromehounds as an example.

Keyword agent architecture, multi-agent, character AI, design pattern, goal-oriented, hierarchical planning, robust

1. 概論

1.1. キャラクターAIとは何か？

デジタルゲームにおけるA Iは、主にキャラクターのブレインとして実装されるキャラクターA Iとゲームシステムに埋め込まれて知的にゲームをコントロールするシステムA Iに分類される[1].

エンターテインメントにおけるキャラクターA Iはゲームの進化に伴い、ますます高度に知的な機能が要求され、次世代機開発(PlayStation3, Will, Xbox360...)からは学術において人工知能分野が培って来た技術が、いよいよ本格的に導入される機会を得ようとしている。本論文は、クロムハウন্ズ(Xbox360, ©SEGA

Corporation / FromNetworks, Inc. / FromSoftware, Inc., 2006年)を例に、エンターテインメントにおけるキャラクターA Iにおける人工知能技術の技術とその導入の方法を解説するものである。

実際のゲーム開発の中で育まれて来たキャラクターA Iの技術は、ゲーム開発を通じた成果がGDC(Game Developers Conference)を始めとするゲームカンファレンスやゲーム技術書[2][3]などで公開され蓄積されて来たが、学問として明確に体系化されたことはない。それは、ゲームA Iがゲームデザインに依存する部分が多く、統一したフレームを構築することが困難であることに起因している。

ゲームA Iには二つの側面があり、「技術としての

構造的な側面」と「ゲーム内のデザインパターンとしての機能的側面」である[4]。前者は純粋な技術であり、後者は、長い年月をかけてゲーム開発者が見出して来たゲームA Iの遵守すべき原則である。

プレイヤーはアルゴリズムではなく(たとえそれが演出だとわかっている)ゲーム世界における「キャラクター」と戦いたいという欲求を持つ。NPC(ノン・プレイヤー・キャラクター)をビジュアル、サウンド、ストーリー、演出を通して「知性を持つ生命」としてプレイヤーに見せることが、エンターテインメントにおけるキャラクターA Iの目標である。

1.2. キャラクターAIのデザインパターン

デジタルゲームにおいてキャラクターA Iがゲーム内で守るべきパターンとして、以下の5つを抽出することが出来る。

- ① NPCはゲーム世界の情報をプレイヤーと共有せねばならない(情報の共有)。
- ② NPCはプレイヤーを認識し、認識していることをアピールせねばならない(プレイヤーの認識)。
- ③ NPCはプレイヤーに対してあたかも意志を持っているように振舞わねばならない(意志の表現)。
- ④ NPCは自分自身を認識し、その知的、身体的特性を活かした行動を取らねばならない(自己認識)。
- ⑤ NPCが持つ思考は、プレイヤーに適用したときに、プレイヤーの行動を十分に予測できるものでな

ければならない（人の思考への近似）。

ゲームA Iの製作は、技術者、企画者、設計者の共同作業であり、デザインパターンは分野を超えた共通認識と方向性の統一に必要かつ効果的である。クロムハウন্ズのA I製作においても、幾つかは明確に意識して製作し、幾つかは結果的に遵守する形となった。

1.3. エージェント・アーキテクチャ

クロムハウন্ズのキャラクターA Iは、フレームとしてエージェント・アーキテクチャを利用する。エージェント・アーキテクチャは1980年代から人工知能の研究分野として発展し、2000年頃からデジタルゲームにおけるキャラクターA Iのフレームとして主に欧米のFPSのキャラクターA Iに応用されて来た（[5][6][7]）。ゲームA Iはこれまで「想定する全てのゲーム状態に対して行動を指定しておく”力任せによる作り込み”」の手法が主流であったが、エージェント・アーキテクチャは実装される知的なメカニズムによって複雑な情報を解析し自動的に行動を決定（生成）する仕組みとして注目されている。前者の手法は、全ての状態を押さえることが可能であるスケールのゲームで最も強力な方法であるが、次世代機開発のスケールでは状態数が膨大な数となり、序々に後者の手法が強力になりつつある。

2. クロムハウন্ズのキャラクターA I

2.1. クロムハウন্ズ

クロムハウন্ズは、オンラインの3D空間内で、プレイヤー・チーム（最大6体）同士が戦うロボット・アクションゲームである。ロボットは地表を歩き、装備された武器（大砲、銃、爆弾など多数）によって相手や建造物を破壊することが出来る（図1）。

ルールは、15分以内に相手基地を爆破するか、相手チームを全滅させることが出来れば、勝利となる。敵基地は候補として3～4つの地点が指定されているが、ゲーム開始時にどれが本物かわからないため、探索して仲間と情報を共有する必要がある（図2）。

ステージは操縦するロボット機体のスケールを20mとして4～8kmであり、通信塔がマップ全体に渡って配置されている、占拠した通信塔を中心とする一定半径距離内と特殊な所持レーダーをつないだ領域内でのみ味方同士の通信が可能となる。そのため「敵が占拠した通信塔を再占拠する」「キーとなる通信塔を占拠して通信領域を分断する」など様々な戦略が可能である。またゲーム開始時には敵チームの位置や武装の情報も皆無なため、ゲーム進行に応じた通信による情報戦が展開される。

基本はプレイヤー同士の対戦であるが、CPU戦と呼ばれるA Iチームと対戦するモードが以下の2つの目的で用意されている。

- (i) プレイヤー・チームの対戦相手がいない場合に対戦相手になる。
- (ii) 新人チームが研鑽を踏むための練習相手となる。

機体はその特性によって「斥候」「攻撃」「重攻撃」「防御」「指令官」「スナイパー」タイプに明確に分類され、役割分担と協調によってチームを勝利に導くことが、このゲームの最大の特徴である。



図1 クロムハウন্ズのゲーム画面 操作機体と通信塔

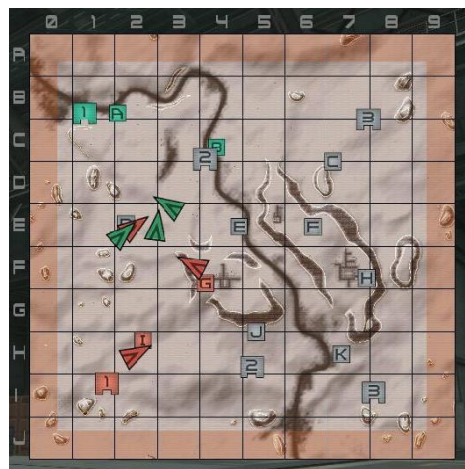


図2 クロムハウন্ズのマップ俯瞰図 両チームの基地候補地（番号付き四角）（デバッグ用に緑と赤に色分けし、NPCの位置を三角で表示）

2.2. クロムハウন্ズAIのデザインパターン

クロムハウন্ズでは上記の2つの目的のために、人間のチームに可能な限り近いA Iチームを構築することを目指した。デザインパターンと対応させ、クロムハウন্ズに必要な機能を抽出すると以下ようになる。

- ① 地形情報と戦況を認識して行動する。
- ② プレイヤーの位置を把握し、見失っても記憶をたよりに予測を行う。
- ③ プレイヤーに対し短期的には攻撃の意志を、長期的には勝利への意志を持っているように振舞う。

- ④ 機体の性能に合った戦術を選択する。
 ⑤ プレイヤーの行動パターンを抽出して実装する、
 これらの要件を満たす基礎技術として

(I) エージェント・アーキテクチャ 周囲の状況を認識して自律的に意思決定を行う能力

(II) リアルタイムパス検索システム マップ上の任意の2点間のパスを決定して移動する能力

を実装する。ここで、ゲーム開発の技術導入で最も大切なことは、

(A)導入技術をゲームデザイナーがチューニングできる環境を完備すること。

(B)大量製作のためのコンテンツ・パイプラインを構築すること。

である。特にゲーム製作においてコンテンツに関与する技術は、ゲームデザイナーが調整できる形で提供して始めてタイトルの質の向上に貢献できる。また、ゲーム・プロトタイプ完成後は、同様のフォーマット・データをコンテンツ・パイプラインによって大量生産を行う体制となるため、ツールやライブラリの形で技術をパイプラインに組み込む作業が必要である。クロムハウন্ズは80を超えるステージに対して、全てのマップに対応できる一つのAI思考を、各マップ用のパス検索用データを自作ツールによって製作した。

3. エージェント・アーキテクチャ

キャラクターAIにおけるエージェント・アーキテクチャは、センサーをキャラクターの五感による情報の取得、エフェクタをゲーム世界へのキャラクター・アクションと見なして構成する。デジタルゲームは、デジタル空間内の仮想空間に内包されているため、キャラクターAIとゲーム世界の間に情報の循環が発生し、この流れがAIを駆動する仕組みとなる。AIは流れる情報から、一定のフォーマットの記憶をスタックし、その過去の記憶から抽出した情報を意思決定に利用することが出来る[5][6]。

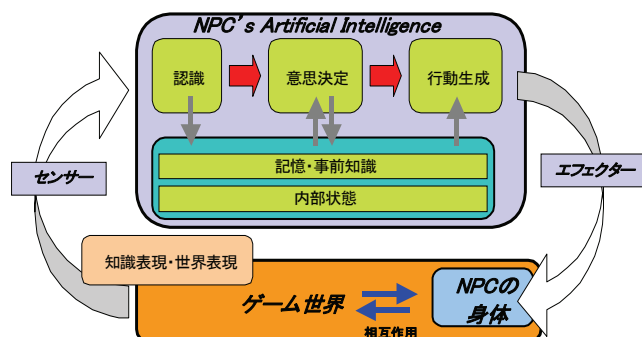


図3 ゲームにおけるNPCのエージェント・アーキテクチャ

3.1. 全体の仕組み

プレイヤーの行動の観察（通常、ゲームAIはゲー

ムの形成と共に製作しなければならないが、この開発ではゲーム性が確立してからAIを組み立てることが出来た）から「通信塔を取る」「基地を確認する」「敵を攻撃する」など目的を一つ一つ実行していることが確認できたため、AIに対しても同様に大きな時間スケールで行動を形成できる階層型ゴール指向プランニングの技術を採用した。

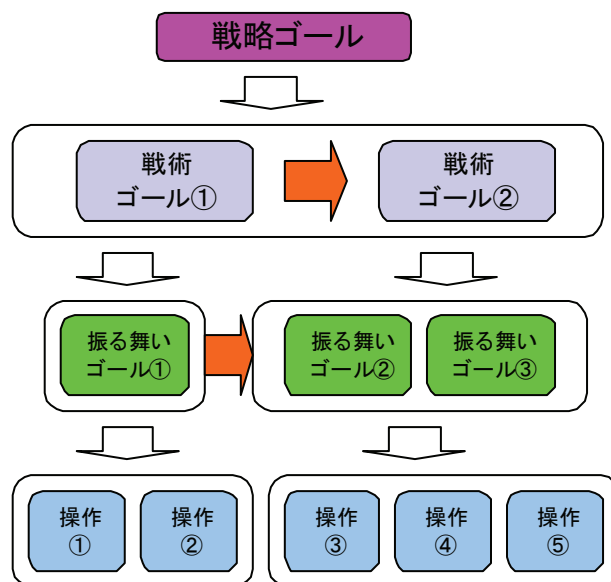


図4 階層的にゴールが分解されて行く模式図

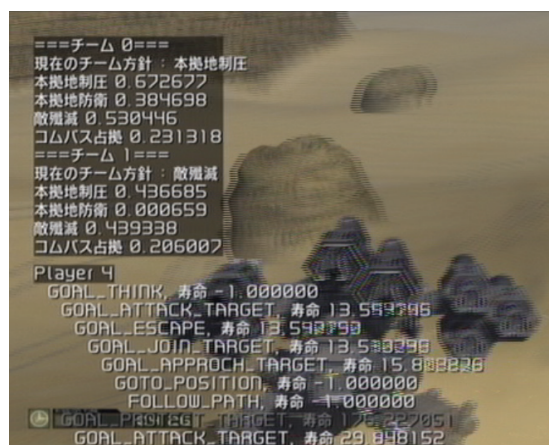


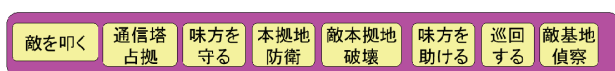
図5 プランニングにより生成された階層型ゴール

階層型ゴール指向プランニングシステムとは、まず達成すべき大目標（戦略ゴール）から出発して、その達成に必要なより小さなゴール（中間ゴール）に分解し、さらに、その中間ゴールを、最終的にはキャラクターの身体（一般的には、AIが作用するゲーム内の要素：ロボットゲームであればロボット、ボードゲームであれば駒の集合）に対する単純なコマンドのシーケンスまで分解して行動を生成する方法である[8]。ある程度想定される基本となるゴール（「移動」「戦闘」など）を実装した後は、プレイヤー・チーム

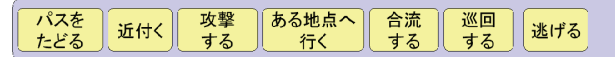
と対戦を100回ほど継続し、対戦ごとに開発チーム内で討論を重ねながら漸的にゴールを拡張した。最終的に、「戦略」「戦術」「振る舞い」「操作」4階層からなる階層型ゴールを構築した。各ゴールはクラスとして実装し、スクリプトによって「ゲーム状態と内部状態」に応じた下位のゴールへの分解の仕方を定義する。各ゴールには達成時間を設定し、時間内に達成できなかった場合は「失敗」と見なす。戦略ゴールレベルの達成時間は1～3分であり、下位に行くほど、達成時間は短くなる。

最終的なゴール総合図

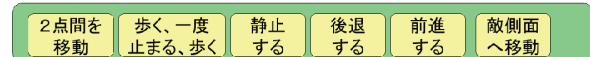
戦略層



戦術層



振る舞い層



操作層

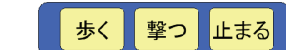


図6 クロムハウنزのゴール階層図（一部）

3.2. センサー

機体のセンサーは、前方に対する約70度の視野角の中に入った敵に対して、レイキャスト・チェックをパスした機体を「認識した機体」として登録する。下半身と独立に旋回する上半身によって、より広いフィールドを索敵することができる。一度認識した敵は一定時間記憶される。これは「一端視野角から外れた敵に対しても攻撃を可能にしておく」ためであり、また、遠くから視認した敵を追尾するために利用する。

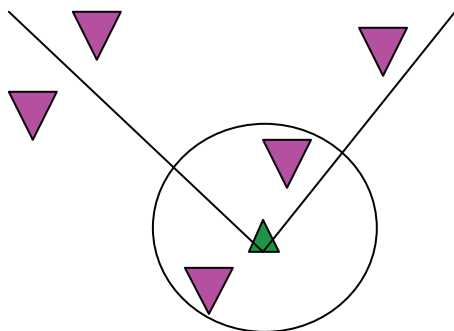


図7 クロムハウنزAIのセンサー領域

記憶した敵の位置情報は、見失った場合、敵の位置を中心に時間経過と共に拡大する半径の円の中のランダムな一点に対して定期的に追尾する。これによ

て相手の位置がだんだんと「ぼやけて行く」効果を取り入れている。

またNPC同士が相互通信可能領域内に入った場合、最新の敵情報が相互に交換される。このように個体のみならず、通信によってチーム全体の認識能力を拡大している。

また、「斥候」タイプ的高速な機体は、敵基地の真偽を探索する役割を持ち、情報を味方の通信領域内に持ち帰って共有する。チーム全体として、簡単なナレッジ・マネジメント機能（「4つのうち3つが偽者なら残りが本物である」など）を持つ黒板モデルの共有メモリを持ち、収集した情報から推論して敵の真の本拠地を決定する。

3.3. 意思決定機構

3.3.1. 戦略意思決定

ゴール指向の最上層である戦略ゴールには、それぞれ自己評価関数が設定されている。評価関数は、ゲーム状態とAIの内部状態（機体の攻撃力、防御力、残り体力）を変数とする関数であり、初等関数の組み合わせで一つ一つ作り込まれている。ゲームテストをくり返し、各戦略が戦況に応じて適切に競合するように関数を微調整した。また機体のタイプによっては選択できない戦略を指定するフィルターをかけた（例えば「ディフェンダー」は「敵基地偵察」を行うことはない）。意思決定は、最大評価値を持つ戦略ゴールを選択することで行う。意思決定のタイミングは、

- ① 現在実行中の戦略ゴールを達成したとき
 - ② 現在実行中の戦略ゴールを失敗したとき
 - ③ チームAIから意思決定を促されたとき
 - ④ 戦況が変わったとき
- である。

3.3.2. 短期意思決定

上記の戦略意思決定は、1～5分に渡る長い時間スケールに対応した意思決定であるが、例えば「曲がり角で敵と遭遇する」場合など、短期的に戦闘状態に入ることが多い。この場合、戦略ゴールを撤回して「敵を叩く」戦略を選択しては、実行中の戦略は頻繁にキャンセルされてしまう。そこで、戦略実行中でも「敵を叩く」ゴールを一時的に挿入し、一定時間（15秒）の後クリアし、もとの戦略に沿った行動に復帰するようにした（図5）。この仕組みによって、プレイヤーの攻撃に対してロバスタな意思決定システムを実現した。さらに「射程距離に敵を捉えた場合は撃つ」「パイプ・砲台は射程距離に捉え次第撃つ」など、さらに短期的で反射的なアルゴリズムを実装した。このように、時間スケールに合わせた多段階の意思決定によって多様な時間スケールに対応するAIを構築した。

3.3.3. 武器選択決定

機体は複数の武器を装備し、同時に使うことが可能である。複数の武器から「装填必要時間」「ダメージ力」「残り弾数」「距離」を変数とする評価関数によって武器を自動選択するシステムを実装した。

3.4. 行動生成

選択した戦力と戦況と内部状態に応じて、最終的に行動はプランニングによって操作シーケンスとして生成される（図4）。

ここでは、特別な局所戦闘の「敵と戦う」ゴールにおける行動ルールを説明する。「敵と戦う」ゴールの下には、「前進する」「左右に進む」「敵の横に移動」「敵の背後に回りこむ」など5～10秒単位のゴールが用意されており、これをランダムに選ぶことで局所戦闘を実現している。また、局所戦闘においては、射程距離の間合いを考慮して移動する必要があるが、「敵と射程距離ぎりぎりの間合いで打ち合う」というゴールを戦闘の中で頻繁に選択させることで、間合いを取りながら戦闘を行う能力を実現している。

4. リアルタイムパス検索システム

4.1. ナビゲーション・メッシュの生成と仕掛け

NPCの移動はナビゲーション・メッシュ上のA*アルゴリズムによるパス検索によって行う。ナビゲーション・メッシュデータは、当たりモデルから、建物の周囲など、細かい当たりモデルをメッシュのまま残したい部分にフィルターをかけ境界を保護しつつ、3dsMaxのポリゴンリダクション関数を用いてオリジナルのツール上で自動的に生成する。全80マップであり、各4～8万メッシュである。

各メッシュには生成の過程で、モデルデータから「地表状態」（砂、水、雪、コンクリート、土）、地形解析から「トポロジー状態」（崖の淵、谷の淵）の情報を取得して埋め込み、A*検索においてコストとして換算する。例えば「水」「雪」「砂」などは機体の移動速度を減少させるため、同様に崖の淵は機体が引っ掛かりやすいためメッシュ上の移動コストを高く設定する。これによってA*パス検索は最短距離検索ではなく最短時間検索を計算することになる。例えば、最短パスでは「足を取られながら河を横切って戦闘」をするが、最短時間移動では「自然にコンクリートの橋を渡って移動する」ようになり、プレイヤーに対するAIの知性のアピールが大きい。また建築物についてもメッシュ生成過程においてオブジェクトとメッシュの衝突計算を実行し、建築物の下にあるメッシュを建築物の形状に沿って細分化する形で新しくメッシュを生成し、NPCが通行不能であるというフラグ情報を埋め込む（図8）。ゲーム内では「建築が破壊されるとその下にあるメッシュのフラグが解除される仕組み」になって

おり、AIは動的なオブジェクト破壊に対応して新しくパスを見出す能力を持っている。

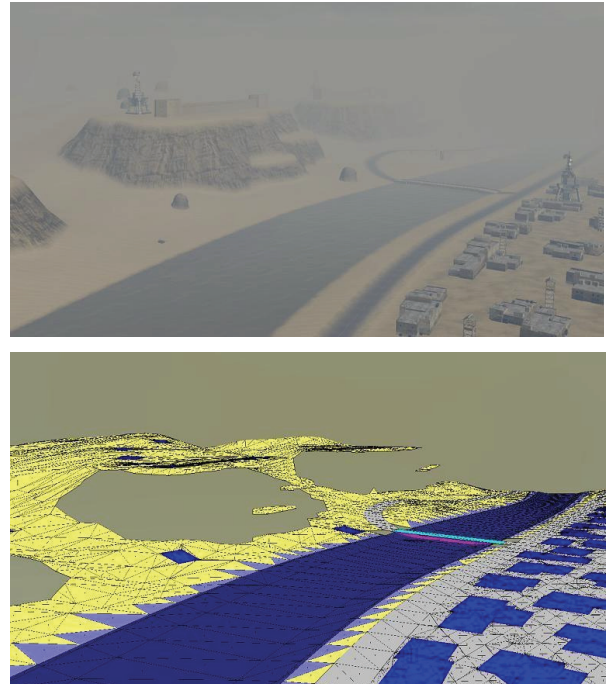


図8 描画マップ（上）と地表情報が埋め込まれたナビゲーション・メッシュ（下）（黄色は砂、紺色は水、青は建築、ピンクは橋の下、水色は橋）

4.2. パス検索の利用法

パス検索は単に移動能力を獲得するだけでなく、様々な活用法がある。例えば「味方（敵）と合流する」ゴールは其中で定期的にパス検索をくり返すことで、味方へ漸近的に近づくアルゴリズムを実現する。また、デバッグにおいてナビゲーション・メッシュの接続状態をチェックするために利用することが出来る。

4.3. デバッグと多重アルゴリズム

リアルタイムパス検索のデバッグは、使用するメッシュ全てのメッシュの組み合わせをテストする必要があるが、「4～8万メッシュ×80マップ」を全テストすることは不可能である。そのため、基地間の経路や、通信塔間の経路など、頻繁に使用される経路を集中的にデバッグをくり返し（半自動チェックで3000回以上）、それ以外の経路については、万が一、パスを辿れなかった場合の補償策を以下のアルゴリズムで用意した。

- ① オブジェクトに引っ掛かった場合には、少し方向を変えて（後ろ、左、右）進行し直す。
- ② 谷に落ちた場合は現在行っている戦略ゴールをリスタートとする。
- ③ 現在選択している戦略ゴールの達成時間をオーバーすると「ゴール失敗」となり新しい戦略ゴールが意思決定される。

ゲーム内ではプレイヤーにパスを阻害される（谷へ落

とされるなど）ことも頻繁であり，上記の補償策はそこでも大きく A I の移動をロバストなものにする効果を持つのである．

5. マルチエージェント・システム

5.1. ゴール指向によるマルチエージェント

対戦テストを通じて個としての A I の知能を高度化しても，プレイヤー・チームに勝利することは難しい．そこで，各 N P C の戦力を結集してプレイヤー・チームにぶつける必要があった．そこで連携した行動を取らせるために，N P C の戦略ゴールをセットにした「チーム戦略ゴール」を構成し，適切な N P C に戦略ゴールを割り当てるチーム A I の仕組みを導入した（図 9）．例えば，チーム戦略ゴール「敵基地攻撃」は，3 つの「敵基地攻撃」戦略ゴールからなり，最も適切なメンバー（ここでは，敵基地に近く戦力のある N P C たち）を選択して割り当てる．ゴールを割り当てられた N P C は，現在実行中のゴールを再評価した値と，割り当てられたゴールの評価値を比較して，評価値の大きな方のゴールを選択する（つまり命令への拒否権を持つ）．これはチームの大局的な判断と N P C の局所的な判断を競合させるためである．「チーム戦略ゴール」は複数用意し，個の戦略ゴール決定と同様に，各ゴールに設定された評価式に従って最大の値を取るチーム戦略ゴールを採択する．

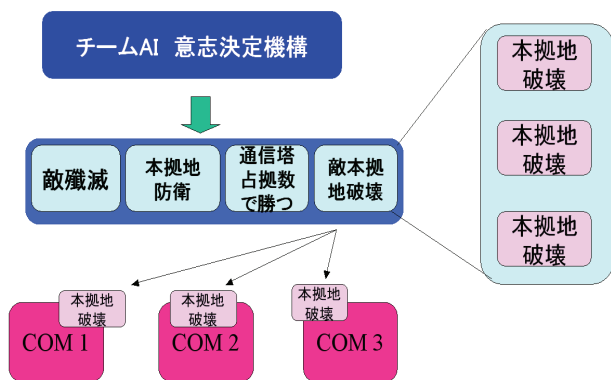


図 9 チーム A I による戦略ゴールの割り当て

5.2. マルチエージェントによるゲームメイキング

チーム A I はゲームの終盤で勝敗を決するタイミングで最も効果的であり，逆にゲーム序盤では各 N P C の自由な判断がゲームを面白くする．そこで，チーム A I が割り当てるゴールと N P C が実行中のゴールを比較する段階で，チーム側の評価値に変動ファクターをかけてチーム制御の強さを調整する．変動ファクターは序盤ではゼロ，中盤で序々に上昇，後半では 1.2 に固定し，チーム A I の力が次第に強まるように調整した．

この仕組みによって，これまで分散した戦力によって各個撃破されていた N P C は，終盤に戦力を結集す

ることで「勝利へ向かう意志をプレイヤーに表現する」ことができるようになった．

$$\begin{aligned} \text{比較のためのチーム A I ゴール評価値} &= \text{ゴール評価値} \times \text{comp_team} \\ \text{比較のための COM A I ゴール評価値} &= \text{ゴール評価値} \times \text{comp_idv} \end{aligned}$$

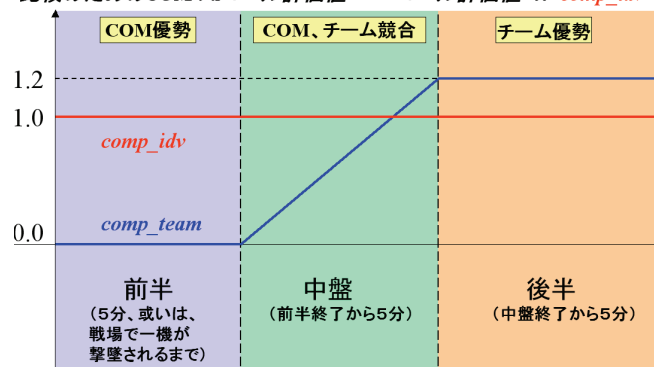


図 10 ゲーム進行とチーム A I の影響力の上昇

6. 総論と反省点

プレイヤー・チームに A I に近づけようとすればする程，人間と A I の間の埋め難い知性の決定的な差異を強烈に意識させられることになる．その差異とは以下の 4 点である．

- ① 人間は同時並行して複数の戦略を思考し，あらゆる瞬間に異なる戦略に切り替えることができる．
- ② 一つの行動を複数の戦略のために行う（「とりあえず通信塔を占拠しておく」という発想）．
- ③ 地形の細かな特徴を利用する（「武器部分だけを突き出して谷間から攻撃する」など）．
- ④ チーム内で言葉を介さずお互いの行動から意図を理解して連携する．

こういった「柔軟な思考」と「細かな粒度の認識」は，現在の決定論的な A I の仕組みの中で明確な壁となって勝敗を分ける原因となっている．こういった差異は，ゲーム A I の分野のみならず，人工知能研究全般に対して重要な課題を投げかけるものである．

文 献

- [1] 三宅陽一郎：「デジタルゲームにおける人工知能技術の応用」，人工知能学会誌,23,1,pp.44-51(Jan.2008)
- [2] Game Programming Gems , Vol.1-6, 株式会社ボーンデジタル
- [3] Steve Rabin (編)：「AI Game Programming Wisdom」, Charles River Media, Massachusetts(2002)
- [4] Christopher Alexander：「オレゴン大学の実験」, 鹿島出版会(1989)
- [5] D. Isla, R. Burke, M. Downie, B. Blumberg：「A Layered Brain Architecture for Synthetic Creatures」, Proceedings of IJCAI (2001)
- [6] Orkin, J：「Agent Architecture Considerations for Real-Time Planning in Games」, AIIDE 2005 (2005)
- [7] Isla,D：「Managing Complexity in the Halo2 AI」, Game Developer's Conference Proceedings. (2005)
- [8] Mat Buckland：「Programming Game AI by Example」, Wordware, Plano (2004)