

## RoboCup サッカーにおけるコーナーキック時の最適な守備配置 Optimal Defensive Positioning for Corner Kicks in RoboCup Soccer

○日紫喜 暁維<sup>†</sup>      楠木 祥文<sup>†</sup>      中島 智晴<sup>†</sup>      秋山 英久<sup>‡</sup>  
○Akei Hishiki   Yoshifumi Kusunoki   Tomoharu Nakashima   Hidehisa Akiyama

### 1 はじめに

ロボット工学と人工知能の領域横断型研究プロジェクトとして、RoboCup [1] が知られている。RoboCup には様々なリーグが存在しており、各リーグにおいて開発が活発に行われている。その背景から、RoboCup サッカーでは人間のサッカー以上に、堅実かつ安定した勝利が求められている。ランダムや必要以上に人為的に作られた戦略を使って勝利するよりも、試合のデータ分析や機械学習によるプレイヤー行動のモデル化を活用し、安定した勝利につながる戦略を構成することが RoboCup の理念と合致する。そのため RoboCup サッカーは競技としてだけでなく、研究としても様々な観点から進められており、パスやドリブルの行動を洗練する、相手の行動を予測して意思決定を行う、相手チームに合わせた戦略を取るなどの研究が行われている。

本研究では、RoboCup サッカーの 1 つであるサッカーシミュレーション 2D リーグを扱う。サッカーシミュレーション 2D リーグでは、プレイヤーの行動はコマンド単位で行われ、各プレイヤーは個人で意思決定をしながら競技に参加する。チームの勝利のためには、プレイヤー個人の意思決定だけでなく、チーム全体で戦略を取ることが求められる。フォーメーション設定などのプレイヤー配置はチーム戦略の 1 つであり、現実のサッカーと同様に、状況に応じてプレイヤーは柔軟に移動する必要がある。しかし現在、我々が開発しているチームでは多様な状況に対して網羅的にプレイヤー配置が考えられているのではなく、ボールの位置を考慮したモデルによるプレイヤー配置が採用されている。このような基本的なモデルでは試合のあらゆる状況において相手チームに対し優位に立つのが難しいことが問題である。そのため、状況に応じたプレイヤー配置を個々に最適化することが重要であると考え、そのための最適化手法の有効性を検証する。

本論文では、限定的な状況としてコーナーキック時の守備的プレイヤー配置について検討する。相手プレイヤーの位置を基にした守備配置を設定し、守備結果の変化を調査する。また、プレイヤー配置は進化計算を用いて最適化する。最後に、守備結果を 5 つに分類し、従来の手法と比較することによって有効性の検証を行う。

### 2 RoboCup

RoboCup は、「西暦 2050 年までに、サッカーの世界チャンピオンチームに勝てる、自律移動のヒューマノイドロボットのチームを作る」という目標を掲げ、人工知能やロボット工学の研究を推進し様々な分野の基礎技術として波及させることを目的としたプロジェクトである。

### 2.1 RoboCup サッカー

RoboCup サッカーは、RoboCup の中の 1 つの分野であり、ロボットの仕様によって、小型や中型、ヒューマノイド、標準プラットフォーム、シミュレーションと、いくつかのリーグに分かれている。このうちシミュレーションリーグは、他のリーグとは異なり、物理的なロボットが存在せず、コンピュータ上に生成された仮想フィールド上でサッカーを行う。RoboCup サッカーシミュレーションには、2次元平面のフィールド上で試合を行う 2D リーグと、3次元空間のフィールド上で試合を行う 3D リーグに分けられる。本論文では、RoboCup サッカーシミュレーション 2D リーグ (図 1) を対象とする。

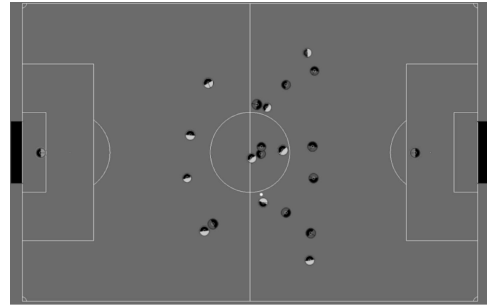


図 1: RoboCup サッカーシミュレーション 2D リーグ

### 2.2 サッカーシミュレーション 2D リーグ

RoboCup サッカーシミュレーション 2D リーグでは、実機を使用せずに、コンピュータ内に用意された 2次元平面を仮想的なサッカーフィールド上で競技する。また、円形のエージェントであるプレイヤーやボールの位置と速度は全て 2次元のベクトルとして表される。プレイヤーや試合中プレイヤーに指示を送るコーチはそれぞれ独立したエージェントとしてプログラムされている。各プレイヤーには実際の人間のような視野が設定されており、視野内で認識できた情報に基づいて、視覚情報が形成される。また他のプレイヤーやコーチがメッセージとして発している情報を、聴覚情報として使用できる。これらの視覚情報や聴覚情報からフィールド情報を形成し、意思決定を行う。しかし、視覚情報にはノイズが含まれ、正確な情報を得ることができない。聴覚情報についても確実に受信できる訳ではなく、メッセージは到達までに遅延が発生する。そのため、プレイヤーはフィールド上にある全てのオブジェクトに関する正確な情報を獲得することができない。このようなノイズやランダムな要素が多くある環境において試合が行われるため、同一の対戦相手であっても、試合毎に結果や内容が異なる。

サッカーシミュレーション 2D リーグでは、試合は 6,000 サイクルからなり、1 サイクルの時間間隔は 0.1 秒である。試合を開始すると各プレイヤーは状況に合わせてあらかじめ用意された戦略をもとに行動を実行し、

<sup>†</sup>大阪公立大学, Osaka Metropolitan University

<sup>‡</sup>岡山理科大学, Okayama University of Science

6,000 サイクルの試合終了まで自動で試合が行われる。チーム開発者は、試合の状態を正確に分析し、最適な戦略を開発することが求められる。そのために、試合における様々な状況に対する戦略を個々に開発できるような独自の環境構築が必要である。

## 2.3 プレイヤ配置戦略

### 2.3.1 ボールの位置に基づくプレイヤ配置

基本的なプレイヤの配置は、ボールの座標やポジションなどのフィールド状態を入力とし、移動すべきプレイヤの座標を出力する写像関数により決められる。有効な配置手法として、Akiyama と Noda が提案した Delaunay 三角分割により空間を分割し、それに線形補間を加えることでコート全体に対応した写像関数近似モデルがある [2]。この手法によるプレイヤ配置は、以下のように行われる。まずフィールド上に代表点を与え、与えられた点群から平面領域を三角分割する。これらの代表点は三角形の頂点となる。次に、三角形の頂点をボールの位置としたとき、それに対応したプレイヤ座標のデータを用意する。つまり各頂点には、ボールの座標に対するプレイヤの移動位置が出力値として割り当てられる。最後に、ボールの座標が三角形内部にある場合、三角形の頂点を用いた内挿法によりプレイヤの移動位置を算出する。各頂点におけるプレイヤの配置は、GUI ツールである FormationEditor [3] を使って細かく調整することができる (図 2)。

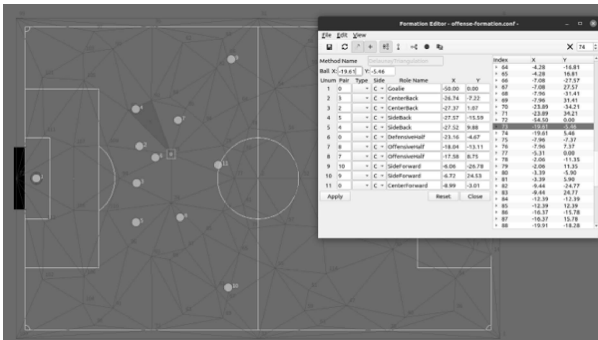


図 2: FormationEditor

Delaunay 三角分割を用いた関数近似モデルは高い柔軟性や再現性、汎用性があり、多くの利点を持つ。しかし、配置の変更が手作業であること、また入力があるボールの位置座標という単一の状態のみでその他のフィールド情報を考慮していないという欠点を持つ。また、特定の相手チームに対する戦略強化の有望性も乏しい。そのため、別の情報も参照した戦略を考える必要がある。

### 2.3.2 相手プレイヤの位置に基づくプレイヤ配置

サッカーのような 1 つのフィールド上に敵のプレイヤが存在するようなスポーツでは、それらの位置情報が重要である。相手プレイヤの位置情報は、攻撃の場合はドリブルやパスなどの自チームの攻撃行動に、守備の場合はインターセプトやマークによる相手チームの攻撃行動の制限などに必要である。本論文では、守備の場合のマークに着目し、相手プレイヤの位置情報を用いた有効なプレイヤ配置を考える。セットプレイにおけるプレイヤ配置は、マーク割当てと相対位置決定という 2 つのタスクからなる。マーク割当てでは各プレイヤに対してマーク

すべき敵プレイヤを決定し、相対位置決定では、割り当てられた敵プレイヤに対してどの位置につくべきかを考える。前者は最小費用流問題として定式化して計算する方法が提案されている [4] が、後者はこれまで検討されていない。そのため、各プレイヤに割り当てられたマーク対象に対する相対位置を最適化する手法を考える必要がある。

## 3 問題設定

### 3.1 コーナーキックシミュレーション

本研究では、限定的な状況としてコーナーキック時の守備を採用した。この状況におけるプレイヤ配置問題を、目的関数をコーナーキック守備時の失点率とし、これが最小となるプレイヤ配置を求めるとみなすことにする。そのため、コーナーキックを繰り返し行う環境を作成した。ここで、1 エピソード (コーナーキック開始から終了までのサイクル) の終了条件は次の 5 通りである。

- Myball: 攻撃側チームからボールを奪い取った場合 (ボールを奪って 10 サイクル保有し続けたとき)
- Myballout: 守備側チームのセットプレイとなった場合
- Backball: ボールが自陣のコートの半分よりも相手側に移動した場合
- Ballout: 攻撃側チームのセットプレイとなった場合
- Goal: 相手にゴールを決められた場合

これらの終了条件は、守備結果の比較のための指標としても用いる。チームとして、守備側は大阪公立大学で開発している omuHam、攻撃側は 2021 年に RoboCup 世界大会で優勝した CYRUS2021 [5] を使用する。

### 3.2 CMA-ES の適用

RoboCup サッカーシミュレーション 2D リーグでは、オブジェクトの位置は実数値ベクトルで表される。そこで、各プレイヤのマーク対象との相対位置  $(x, y)$  座標を全て結合したものを多次元実数値と表し、最適化する手法を考える必要がある。また最適化により得られた解は、プレイヤ配置に直接変換可能である。つまり、マーク対象の相対位置最適化問題を多次元実数値最適化問題として考えることができる。ただ、最適化される評価関数はシミュレーションを通して計算されるブラックボックス関数である。本研究では、この問題に良好な性能があるとされる CMA-ES [6] を用いる。

CMA-ES は、実数値連続探索空間のブラックボックス関数最適化に使用される進化戦略手法である。各世代における個体の生成は、平均ベクトル、ステップサイズ、共分散行列によって制御される多変量正規分布のサンプリングを用い、解を逐次的に導出する。ある世代  $g$  ( $g = 0, 1, 2, \dots$ ) における新たな個体  $x_k^{(g+1)}$  のサンプリングは以下の式に基づいて行われる。

$$x_k^{(g+1)} = m^{(g)} + \sigma^{(g)} N(0, C^{(g)}), \quad k = 1, \dots, \lambda$$

ここで、 $m^{(g)}$  は探索分布の中心 (平均ベクトル)、 $\sigma^{(g)}$  は世代  $g$  の子個体全体の標準偏差 (ステップサイズ)、 $C^{(g)}$  は共分散行列、 $\lambda$  は子個体数、 $k$  は子個体の番号を表す。世代の進化は、 $m^{(g)}$ 、 $\sigma^{(g)}$ 、 $C^{(g)}$  の更新によって

行われる。

CMA-ES には、各更新式における学習率や調整値など多くのハイパーパラメータが存在する。本研究では推奨値 [6] を用いて実験を行った。ただし、初期の探索分布中心  $m^{(0)}$  とステップサイズ  $\sigma^{(0)}$ 、探索終了世代数に対する推奨値は存在しないため、対象問題に合わせて自分で調整を行う必要がある。

### 3.2.1 CMA-ES の構成

本研究へ CMA-ES を適用するために、個体の遺伝子型とその評価方法を決める必要がある。

個体の遺伝子型は各プレイヤーのマーク対象との相対位置  $(x, y)$  座標からなる。これは、各プレイヤーの相対位置を直列で結合したものとなる。本論文では、1 個体を  $2(x, y)$  座標  $\times 6$  (プレイヤー) = 12 (次元) で表現することにする。マークを行うプレイヤーの数は、攻撃側チームのコーナーキックに関するプレイヤーの数がコーナーキックキッカーを除いて 6 人であるため、上記のように設定した。それ以外の守備側プレイヤーは、2.3.1 節の手法で決定した位置に移動する。また、個体の評価値は失点率とする。

標準的な CMA-ES [6] では前世代の個体を次世代へ残さずすべて入れ替えるが、今回は前世代の最良個体を 1 つ残すエリート保存 [7] を行う。これは、2.2 節で述べたように試合毎に結果が多少異なる場合があり、評価値が偶然良くなった可能性があるためである。

### 3.2.2 パラメータ設定

CMA-ES [6] では、実装する際に事前に初期の正規分布中心とステップサイズ、探索終了世代数、生成子個体数を設定する必要がある。本研究の数値実験では、初期の正規分布中心を原点  $(0, 0)$  と設定した。初期のステップサイズ、探索終了世代数を決定するために事前実験を行った。事前実験では個体群収束度を調べた。個体群収束度は 1 世代における個体群の標準偏差であり、標準偏差が小さいほど、個体がある解に収束していると考えられる。事前実験より初期ステップサイズを 5.0、探索終了世代数を 300 世代と設定した。生成子個体数の推奨値は文献 [6] によると  $4 + 3 \times \log n$  ( $n$ : 遺伝子型の次元数) であり、本実験の次元数は 3.2.1 節より 12 次元であるため推奨値は 7 となる。そこで、7 個の子個体を CMA-ES により生成する。エリート保存による前世代の最良個体の 1 個を考慮して個体群サイズは 8 個体となる。

## 4 性能評価

### 4.1 性能比較

CMA-ES により配置を最適化した結果得られた最終世代の最良個体を使って、コーナーキックシミュレーションを 1,000 エピソード行い、その守備性能を確かめた。比較対象はマークなしの従来手法、つまり 2.3.1 節で説明したボールのみを参照する守備配置 (図 6a) とする。表 1 に 3.1 節で説明した各指標を割合で示す。これより CMA-ES で最適化した相手の位置を参照した守備配置のほうが従来手法の配置より失点率が少ないことがわかる。ただ、失点率が大幅に減少したとはいえない。これは、評価方法に原因があると考え、別の評価方法で最適化を行った。

用いた評価値は、図 3 の濃い色で示されたエリア (以



図 3: 危険エリア (濃い色の四角)

降、危険エリア) にボールが入った割合である。これを危険エリア侵入率と呼ぶ。また、危険エリアはペナルティエリアを  $\frac{1}{4}$  にした部分である。これは、敵にシュートを打たれる可能性が高いエリアであり、失点の原因となる事象を減らすためである。ここで、図 4 は、各世代における評価値の遷移を表した図である。丸は外れ値、端点は最大値、最小値を表す。また、図 5 は個体群収束度の遷移を表した図である。どちらも評価値を危険エリア侵入率とした場合である。評価値を危険エリア侵入率として得られた最良個体を使って、性能調査を行った結果を表 1 に示す。ここで、Hazard は危険エリア侵入率である。これより、評価値を失点率の場合より危険エリア侵入率とした場合の方が失点率が少ないことがわかる。これは、評価値が失点率という直接的なものではなく、失点に起因するような事象の最小化の方が失点を減らすことができるからだと考えられる。また、評価値が Hazard において Ballout の値が多いのは、最適化した配置により攻撃チームのボール保持者に対してパスなどの行動を制限することはできたが、コーナー近くでボールを奪い取ろうとしてライン際での守備が増えたことで、スローイン等の攻撃側のセットプレイとなる場合が増えたためと考えられる。

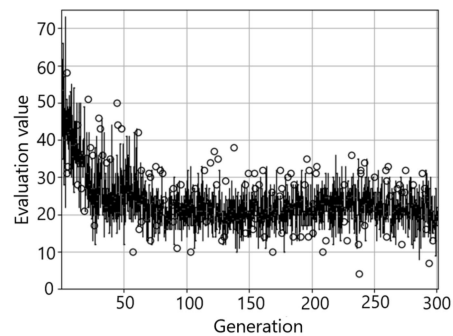


図 4: 世代ごとの評価値の遷移

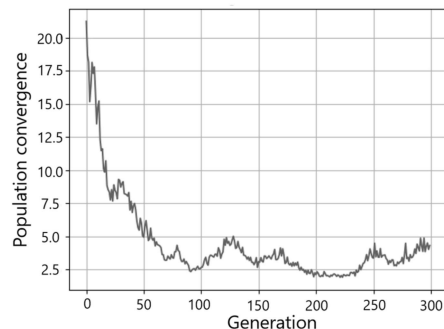
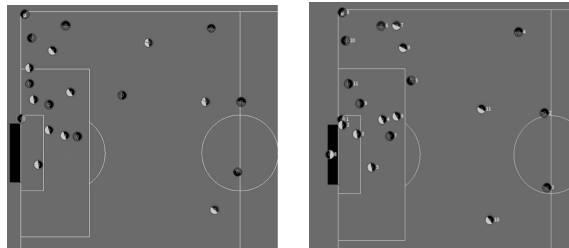


図 5: 個体群収束度の遷移

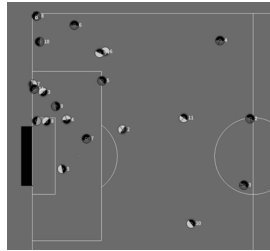
表 1: 1,000 エピソードの守備結果 (vs CYRUS2021)

守備方法 (マーク)	なし		あり	
	評価値	危険エリア	失点率	
Myball	20.7	14.2	33.6	
Myballout	22.8	23.5	24.6	
Backball	18.2	22.0	9.5	
Ballout	24.7	33.7	23.2	
Goal	13.6	6.6	9.1	
Hazard	29.9	18.2	35.7	



(a) マークなし配置

(b) 失点率



(c) 危険エリア侵入率

図 6: プレイヤ配置の比較 (vs CYRUS2021)

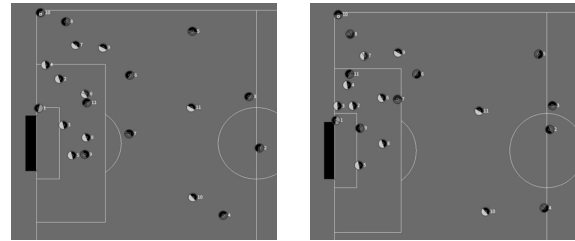
評価値ごとに最適化した最良のプレイヤ配置を図 6 に示す。ここで、薄い色の円形エージェントは守備側チームのプレイヤであり、濃い色のエージェントが攻撃側チームのプレイヤである。これより、評価値を失点率とした場合はプレイヤがゴール付近に多く配置され、危険エリア侵入率とした場合はコーナーに近いペナルティエリアにプレイヤが配置されている。つまり、評価値を危険エリア侵入率にすることでよりボールをゴール前に持ち込まれる可能性を減らす配置となった事がわかる。

#### 4.2 追加検証

加えて、攻撃チームを変更して最適化を行った。相手チームとして、2021 年 RoboCup 世界大会で 2 位の HELIOS2021 [5] と 3 位の Yushan2021 [5] を採用する。最適化の際の評価値は、4.1 節の実験より性能が良いと思われる危険エリア侵入率とした。表 2 に実験結果を示す。HELIOS2021 は、コーナーキック開始時にプレイヤが守備プレイヤのいないスペースに動く戦略をとってくるチームである。実験結果よりそのような戦略をとるチームに対しても有効であることがわかる。ただ、Yushan2021 の場合も同様に失点率が大幅に減少しなかった原因として、プレイオン中の守備行動が洗練されていないことが挙げられる。対戦チームごとの最良個体の配置を図 7 に示す。どちらもコーナーに近いエリアにプレイヤが配置されていることがわかる。

表 2: 1,000 エピソードの守備結果

攻撃チーム	HELIOS2021		Yushan2021	
	守備方法 (マーク)	なし	あり	なし
Myball	18.4	15.7	16.4	14.3
Myballout	19.0	14.0	21.7	23.5
Backball	23.2	33.4	13.5	10.5
Ballout	24.4	25.7	42.0	40.7
Goal	15.0	11.0	6.4	4.3
Hazard	33.3	28.1	14.2	11.0



(a) vs HELIOS2021

(b) vs Yushan2021

図 7: 対戦相手ごとのプレイヤ配置

#### 5 おわりに

本論文では、コーナーキック守備時の初期位置において CMA-ES を用いて最適化したプレイヤ配置の有効性を調査した。実験の結果、相手プレイヤの位置情報に基づき最適化されたプレイヤ配置は、従来のボールのみの位置情報に基づいた守備配置よりも失点数が少なくなった。また、異なるチームや戦術に対しても有効であることを示した。

今後の課題として、評価値や味方選手の位置情報も参照する配置手法の検討などが挙げられる。さらに、コーナーキック以外のセットプレイ時の配置、またセットプレイ時の初期配置だけでなく、プレイオン時の最適配置や行動の調査などが必要である。

#### 参考文献

- [1] Hitoaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, Eiichi Osawa and Hitoshi Matsubara, "RoboCup: A Challenge Problem for AI", AI Magazine, Vol. 18, No. 1, pp. 73–85, 1997.
- [2] Hidehisa Akiyama, Itsuki Noda, "Multi-agent Positioning Mechanism in the Dynamic Environment", RoboCup 2007, LNAI 5001, pp. 377–384, 2008.
- [3] Hidehisa Akiyama, Tomoharu Nakashima, "HELIOS Base: An Open Source Package for the RoboCup Soccer 2D Simulation", S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 528–535, 2014.
- [4] 田中翔, 三船 哲, 中島 智晴, 秋山 智久, "RoboCup におけるセットプレイのマーク割当の性能調査", 人工知能学会第二種研究会資料 社会における AI 研究会 SIG-SAI-023-01.
- [5] archive.robocup.info, <https://archive.robocup.info/Soccer/Simulation/2D/binaries/RoboCup/2021/Day3/>, 2023 年 1 月 19 日
- [6] Nikolaus Hansen, "The CMA Evolution Strategy: A Tutorial", arXiv:1604.00772v1 [cs.LG] Apr 2016.
- [7] 小林 一郎, 人工知能の基礎, サイエンス社, 2008.