

ゲーム学会 学会誌
Vol. 2, No. 1 別刷
2008. 3. 1 発行

不完全情報ゲームでの可能局面数え上げに基づく着手決定

—戦艦ゲームを題材として

Decision Making based on the Generation of Possible Positions in an Imperfect-Information Game

— A case study using the Battleship Game

作田誠*

*福岡工業大学

Makoto Sakuta*

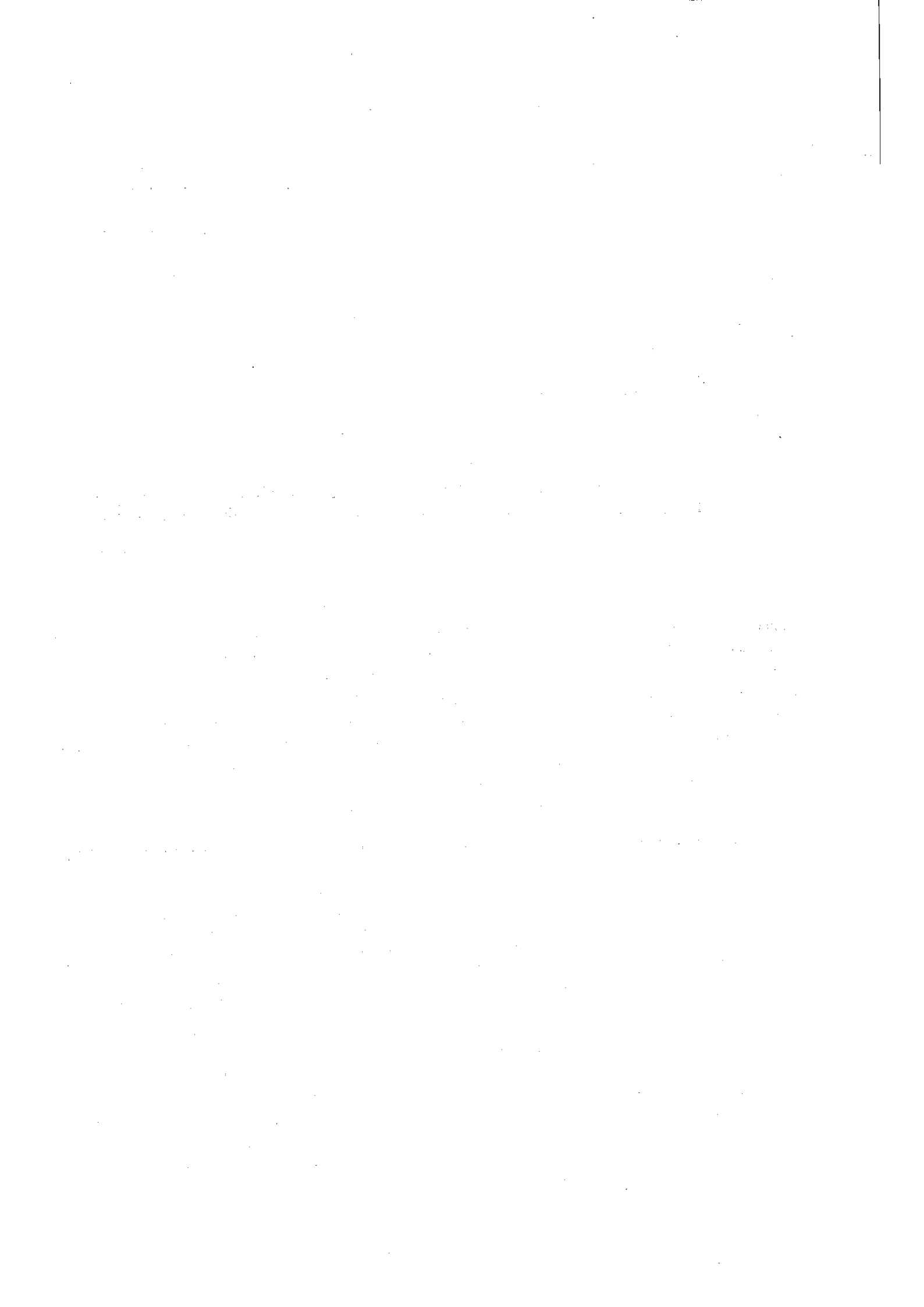
*Fukuoka Institute of Technology

飯田弘之**

** 北陸先端科学技術大学院大学

Hiroyuki Iida**

** Japan Advanced Institute of Science and
Technology



原著論文

不完全情報ゲームでの可能局面数え上げに基づく着手決定 — 戦艦ゲームを題材として

作田 誠* 飯田 弘之**

* 福岡工業大学

** 北陸先端科学技術大学院大学

抄録: ポピュラーな不完全情報ボードゲームの代表として, サイズ2~5の船5隻と10×10ボードを使う戦艦ゲームを選び, 配置をランダム戦略に固定して効率的な攻撃戦略について調べた. ボード全体を均等的にカバーする均等攻撃については市松模様攻撃が優秀であった. 完全情報問題の探索手法の応用といえる可能局面数え上げに基づく攻撃戦略を提案し評価したところ単なる均等攻撃に比べ優秀性を示した. 中でも, 市松模様攻撃を基本として攻撃がはずれになった場合の可能局面数を最小にする枡を攻撃する, すなわち, 当たる確率が最大の枡を攻撃する戦略が最もコストパフォーマンスに優れていた. より早く数え上げを始めるほど, より確度の高い攻撃を示した.

キーワード: 戦艦ゲーム, 不完全情報ゲーム, 攻撃戦略, 局面数え上げ

Decision Making based on the Generation of Possible Positions in an Imperfect-Information Game — A case study using the Battleship Game

Makoto Sakuta* Hiroyuki Iida**

*Fukuoka Institute of Technology

** Japan Advanced Institute of Science and Technology

Abstract: A battleship game that uses 10x10 boards and five ships of the sizes from 2 to 5 has been chosen as a target domain for a representative testbed of the popular board games with imperfect information. The effective attacking strategies have been investigated with the random placement strategy of ship positions and directions. As to the uniform attacking strategies for a whole board, the checkerboard strategy has shown the high performance. The strategies based on the generation of possible positions, which are derived from the methods developed in the perfect-information games, have been proposed and examined. They have turned out superior to the simple checkerboard strategy. Among of all, a strategy, which is based on the checkerboard strategy and selects a square such that the number of possible positions is minimized when the attack is assumed to be missed, has shown the highest cost-performance. The accuracy of the attacks has turned out to get better as the generation of possible positions is performed earlier.

Keywords: battleship game, imperfect-information game, attacking strategy, position generation

1. はじめに

完全情報零和二人ゲームはミニマックス方式の探索により性能があがることが知られ, 現在では探索の効率化・高精度化についての研究が中心となっている. 一方, 多くの不完全情報ゲームには性能を上げる確固とした指針はなく, 今後の大きな課題となっている. カードゲームであるブリッジ [12] やポーカー [2, 3] の研究は進んでおり, また最も素朴で単純なじゃんけんを対象として世界じゃんけんプログラミング選手権 [1] が行なわれている. しかし不完全情報ボードゲームに関する研究は数少ない. 筆者らはプレーヤからある種の情報を見えなくすることにより成立している不完全情報のボードゲームに興味を

抱いてきた. 本研究では, 中でも最も素朴かつ有名で多くプレイされたと思われるクラシックな戦艦ゲームを題材とする. 今回, この不完全情報ボードゲームの効率的な攻撃戦略を調べた. 特に, 完全情報問題の解決手法の応用として, 可能局面数え上げあるいは探索に基づく着手決定手法を提案し, その有効性を調べた.

不完全情報ゲームに対する完全情報問題の解決手法の適用としては, ブリッジなどカードゲームにおいてモンテカルロ法を用いて繰り返し完全情報ゲームを解くことにより近似的な解を得る手法 [8, 11] が有効であることが知られており, 最近逆に完全情報の基への応用 [4] も報告されている. 本研究で提案する可能局面数え上げあるいは探索に基づく着手決定手法は, モンテカルロシミュ

レーションで完全情報ゲームを解くのではなく、全可能局面を数え上げた結果を利用するものである。この提案手法はボードゲームだけでなく宝探し問題など一般的な問題領域へも適用できるのではないかと期待される。

2. 戦艦ゲームとは

戦艦ゲームは潜水艦ゲームとも言われる二人不完全情報ゲームで、矩形ボード 2枚 2組とそれに挿す数種の船ピース及び攻撃用に挿すマークを使ってプレイする。船ピースは細長い形で、通常、種類別に 2 枡から 5 枡までのサイズがある。ボードは同じサイズの自分用ボードと相手用ボードがセットになっている。各プレーヤは攻撃に先立って自分の全船ピースを自分用ボード内に自由に挿す。その後は交代で攻撃する。手番プレーヤは相手用ボードの空いている枡目のどれか一つを選んで攻撃マークを挿し、その枡目の座標を相手に伝える。攻撃を受けたプレーヤは攻撃された自分用ボードの枡目に船がなければ「はずれ」、あれば「命中」を伝える。ただし、一つの船の全枡目が命中になったら、その船は沈没となる。最初に相手船を全滅させた方の勝ちとなる。

元々 Web サイト [10] の商品で紹介されているようなクラシックなボードゲームで、10×10 ボード・各軍 5 隻で構成される。サイト [7] では Web でゲームがプレイできる。なお、さまざまな変種が多くあり、例えば 16×16 ボードを使っているサイト [5] やユーザーインターフェースに凝った面白いページ [6] もある。

今回の実験では最も一般的で普及している表 1 に示す詳細ルールを採用した。なお船の沈没時に相手に与えられる情報は船の種類だけなので、その時点では沈没船の配置が確定しない場合もあることに注意してほしい。ちなみに、変種では、船の接続配置を禁じているもの、命中すると続けて攻撃できるもの、船の種類を教えずただ沈没したとだけ伝えるもの、あるいは、沈没した船の種類だけでなくその配置も伝えるものなどもある。

図 1 にゲームの一局を示す。左側が自軍側のボードで、右側が相手軍側である。自軍側の船の配置はすべて分っているが、相手軍側の配置は攻撃をした箇所しか分からない。◎は攻撃してはずれだった枡を、爆発印は当たりだった枡を表す。また、全枡が当りになって沈没し配置が確定した船を黒く塗りつぶして表示している。

3. 攻撃戦略の説明

本研究では、船配置の相手モデル化などを行わず、純粋に効率的に攻撃をすることを考え、以下の攻撃戦略を提案し評価する。なお、未確定枡とは当りになるかはずれになるか分からない枡で、攻撃をしていなくても船がないことが自明な枡は未確定枡には含まない。図 2 に船

表 1: 戦艦ゲームの詳細ルール

<ol style="list-style-type: none"> 1. ボードサイズは 10×10、各軍はサイズ 5 の航空母艦 1、サイズ 4 の戦艦 1、サイズ 3 の輸送船 1、サイズ 3 の潜水艦 1、サイズ 2 の駆逐艦 1、全 5 隻で構成される。 2. 船の配置は自由に行なってよい。 3. 船に命中したときも相手側の攻撃に替わる。 4. 一つの船の全枡目が命中になったら、「戦艦沈没」のように、ある船の種類が沈没したことだけが相手に情報として与えられる。

がないことが自明な枡の例を示す。図中、斜線の枡は残り船の最小サイズが 3 のとき船がないことが自明な枡となる。

3.1 攻撃戦略

1. ランダム (R)
未確定枡をランダムに狙う。
2. ランダム+近傍攻撃 (RNB)
通常はランダム攻撃だが、当りがあったとき沈没させるまでその近傍を攻撃する (近傍攻撃)。図 3 で命中した枡の上下左右の枡が近傍攻撃の対象となる。

戦艦ゲームでは近傍攻撃は確実かつ絶対的に効率を上げる戦略であり、以下の戦略はすべて近傍攻撃を備えている。

まず、ボード全体を均等的にカバーする攻撃である均等攻撃について基本的な 2 パターンの戦略を示す。

3. 市松模様攻撃 (CB)
最初、大きい市松模様を打ち、次にその間を埋めて小さい市松模様になるように打っていく。図 4 に 8×4 サイズで示したが、まず A の枡を攻撃し、続いて B, C, D の枡を攻撃する。ただし毎回同じ順序にならない様、縦横方向に 0 から 3 のランダムのオフセット値を加えている。
4. ナイト市松模様攻撃 (KCB)
ナイト移動 (縦横に (1,2) または (2,1) の移動) を基本に、最初大きい市松模様を打ち、次にその間を埋めて小さい市松模様になるように打っていく。図 5 に 10×7 サイズで示したが、まず A の枡を攻撃し、続いて B, C, D の枡を攻撃する。ただし毎回同じ順序にならない様、縦横方向に 0 から 4 のランダムのオフセット値を加えている。ボード全体にナイト市松模様を作り上げた後はランダム戦略に移行する。

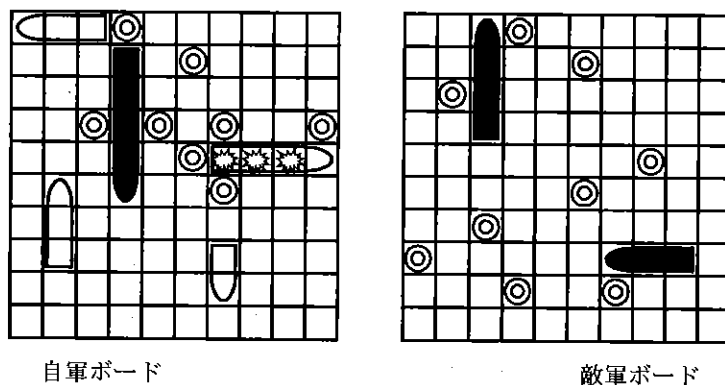


図 1: 戦艦ゲームの一局面

以下は、今回提案する「可能局面の数え上げあるいは探索に基づく戦略」で、上記の均等攻撃戦略と組み合わせて使用する。均等攻撃戦略としては、CBとKCBの比較実験の結果(後述、節4.1参照)をふまえ、優秀な結果の出たCBとの組合せを中心に実験を行った。ただし、参考のためKCBとの組合せも一つ入れてある。

5. 市松模様攻撃+可能配置数最小化 (MP)

最初はCBだが、残り船が4隻以下で、かつ、未確定桁数がある値以下になったら、可能な配置をすべて数え上げ、はずれのときの可能配置数が最小になるような桁を攻撃する。数え上げを始める未確定桁数がxのとき、MP_xと表す。

残り船が5隻だと可能配置数が未確定桁数の5乗で利いてくるが4隻だと4乗で利いてくるだけなので、計算時間を抑えるため「残り船が4隻以下」という条件を入れている。3.2節参照。

6. 市松模様攻撃+数手先の可能配置数最小化 (LA)

残り船が4隻以下で、かつ、未確定桁数がある値以下になったら、決まった回数(指定回数)の攻撃がはずれになるとして、その回数先まで先読みして可能配置数が最小になるような桁を攻撃する。すなわち、指定回数先までの攻撃のうち少なくとも一回が命中する確率を最大とする戦略である。探索のコストを抑えるため前向き枝刈りを使用する。先読み手数がx、枝刈り数がy、数え上げを始める未確定桁数がzのとき、LAP_{x,y,z}と表す。

7. ナイト市松模様攻撃+可能配置数最小化 (KMP)

最初はCBの代りにナイト市松模様攻撃(KCB)を行なう。後はMPと同様である。数え上げを始める未確定桁数がxのとき、KMP_xと表す。

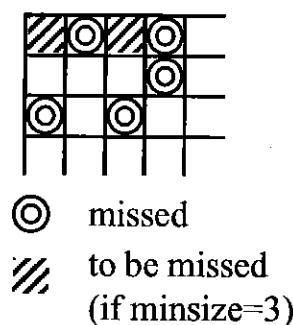


図 2: 空であることが自明な桁

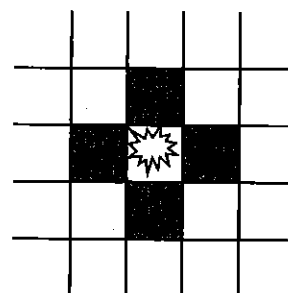


図 3: 近傍攻撃

3.2 可能配置に関する知見

10×10のボード上のx個の未確定桁に対して、サイズ5,4,3,3,2の各船を配置する場合の可能配置数の粗い上界は以下で与えられる。

$$(x-40) \cdot 2 \times (x-30) \cdot 2 \times (x-20) \cdot 2 \times (x-20) \cdot 2 \times (x-5-4-3-3) \cdot 2 < 2^5 x^5$$

初期状態ではx=100なので上界は7.3×10¹⁰となる。また、残り船が4隻の時は粗い上界は2⁴x⁴となる。

4節に示す本研究の実験環境で2週間程度プログラムを稼働させ、初期状態での可能配置について数え上げを行った。その結果得られた各桁における可能配置数を図6に示す。結果は上下左右面対称及び点対称となるので、10×10ボードの内左上の5×5の上三角部分を示してい

	A		C		A		C
D		B		D		B	
	C		A		C		A
B		D		B		D	

図 4: 市松模様攻撃戦略

	A				D		
			C				B
D				A			
		B				C	
				D			A
	C				B		
			A				D

図 5: ナイト市松模様攻撃戦略

るが、実際にはこれが対称性を持って全盤面に展開される。全可能配置数は 1.5×10^{10} で、予想通り真中(A)が最も可能配置数が多く、外側に行くに従い少なくなり、角(O)が最も少ない。ただし真中と角で2.7倍の開きしか見られない。なお、計算時間がかかりすぎるため、初期状態からの数え上げを実際の対戦プログラムで利用するのは現実的でないことを補足しておく。

3.3 戦略MP, LAの詳しい説明

MP 戦略を図 7 を例に説明する。左図が問題局面とする。ただし @ は攻撃したがはずれた枡を、黒い船影は沈没した船を表している。この局面でどの枡を攻撃するのが最も命中確率が高いだろうか？

MP 戦略では残りの船の全可能配置を数え上げ、各空き枡について船が存在している配置の数を求める。配置数が最も大きい枡が最も命中確率が高いので選択される。例では、残るサイズ3とサイズ2の船二隻の可能配置をすべて生成し、各枡への配置数をカウントする。例えば、b4, b5, b6 にサイズ3の船、e1, f1 にサイズ2の船を置く配置が見つかり、b4, b5, b6, e1, f1 各枡のカウントを +1 する。全部で 429 通りの配置についてカウントを足し合わせた結果を右図に示す。f3 の枡がカウント値が 117 と最も大きい(つまり、はずれた場合の可能配置数が最も小さい)ので選択される。

MP 戦略が一手先読みによる命中確率最大の枡を求める戦略であるのに対し、LA 戦略は d 手先読み探索による命中確率最大の枡を求める戦略である。図 8 に LA の探索木(探索深さ $d=2$)を示す。ルート(問題局面)で可能配置数が N とする。ここで着手 m_1 を選択しはずれた場合の可能配置数を N_1 、さらに着手 m_2 を選択しはずれた場合の可能配置数を N_2 とする。このとき、 d 着手全部はずれる確率 P_{allmiss} および d 着手のうち少なくとも一回命中する確率 P_{hit} は以下で与えられる。

$$P_{\text{allmiss}} = (N_1/N) \times (N_2/N_1) = N_2/N$$

$$P_{\text{hit}} = 1 - P_{\text{allmiss}} = (N - N_2)/N$$

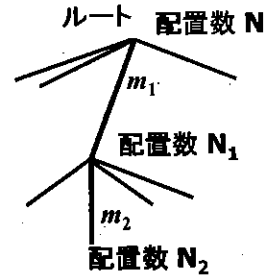


図 8: LA 戦略の探索木

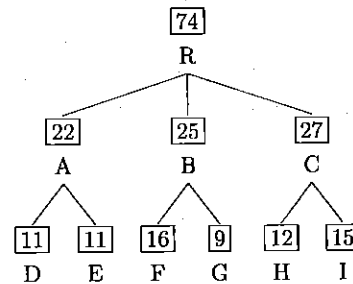


図 9: LA 戦略による MP 戦略への反駁

P_{hit} を最大にするには d 手先の可能配置数 N_2 を最小にできる着手を探索し選択すればよい。

一手先に最もよい着手が数手先にもよい場合、MP 戦略と LA 戦略では同一の枡が選ばれる。しかし、そうでないケース(反駁という)も考えられる。反駁の例を図 9 に示す。問題局面 R(可能配置数 74)において、有力な 3 着手があり、それぞれがはずれると可能配置数 22, 25, 27 を持つ局面 A, B, C に至るとき、MP 戦略では最も命中確率の高い A への着手が選択される。一方、2 手先読みする LA 戦略では 2 着手ともはずれるとき最も可能配置数の小さい局面 G に至る着手、すなわち局面 B への着手が選択される。

4. 攻撃戦略の性能評価

本研究では、船の配置については表 2 に示すランダム配置戦略に固定し、同じ攻撃戦略同士は 10 万回、違う攻撃戦略同士は先後を交代させそれぞれ 10 万回対戦させ、

	a	b	c	d	e
1	O	N	L	K	I
2		M	J	H	F
3			G	E	D
4				C	B
5					A

Total number of possible placements: 15046987768 $\approx 1.50 \times 10^{10}$

Number on each square from A (center) to O (corner): $\times 10^9$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3.21	3.14	3.06	3.00	2.92	2.77	2.77	2.67	2.51	2.49	2.39	2.16	2.15	1.73	1.20

図 6: 全配置数え上げ結果

	a	b	c	d	e	f	g	h	i	j
1		@		@						
2	@		@							
3		@		@						
4	@		@				@			
5			@		@			@		
6	@			@		@				
7		@				@				@
8	@		@		@		@		@	
9		@		@		@		@		@
10	@		@		@		@		@	

問題局面

	a	b	c	d	e	f	g	h	i	j
1		@		@	43	86	86	43		43
2	@		@							84
3		@		@	54	117	81	55		111
4	@	43	@		27	91	@	14		110
5	14	71	@		@	93	13	@		82
6	@	57	28		@	41	@	81	67	82
7		@	57	71	43	@	41	107	54	@
8	@		@	14	@		@	41	@	
9		@		@		@		@		@
10	@		@		@		@		@	

MP 戦略による解決

図 7: MP 戦略の例

表 2: ランダム配置戦略

船のサイズの大きいものから順にすべての船について以下を繰り返す。

1. まず 0 か 1 の擬似乱数を得て船の向きを水平 (右方向) または垂直 (下方向) に決定
2. 次に横方向の位置・縦方向の位置を擬似乱数によって決定
3. 得られた方向・位置にその船が配置できれば配置して次の船の配置へ。できなければ 1 に戻ってやり直す

各攻撃戦略の評価を行なった。実験はすべて Pentium 4 3.0GHz, メモリ 1GB マシンの Windows 2000 OS 上で行った。

なおランダム性の実現には注意が必要であるが、実験では長周期・高次元均等分布を持つ擬似乱数 Mersenne Twister [9] を用いた。

4.1 ランダムおよび均等攻撃戦略の評価

まず、ランダムおよび均等攻撃戦略の対戦結果を表 3 に示す。firstp 及び secondp は先手・後手プレイヤーの攻撃戦略を表す。R, RNB, CB, KCB はそれぞれランダム攻撃, ランダム攻撃+近傍攻撃, 市松攻撃, ナイト市松攻撃を表す。なお、R 戦略は他の戦略に比べ弱すぎるため RNB との対戦結果のみ示している。winrf 及び winrs は先手または後手のパーセントで表した勝率である。ランダム抽出による標本比率の正規分布を仮定して括弧内に 95%信頼区間の増減値を 2 桁で示している。例えば、52.31(31) は 52.00~52.62 が 95%信頼区間であることを表す。hitrf 及び hitrs は先手または後手のパーセントで表した命中率であり、同様に 95%信頼区間を示している。seconds は秒単位の全実行時間である。

対戦結果から、R, RNB, KCB, CB の順に強さが増していることが分かる。CB の優秀性は、均等的攻撃であり、かつ、終了時には必ず全隻に命中させることができる性質のためと考えられる。均等攻撃には CB や KCB の他に、縦横に (1,3) の移動を基本とする大桂馬均等攻撃や、

縦横に (2,3) の移動を基本とする均等攻撃も考えられるが, (1,3) 移動の均等攻撃は CB または KCB に帰着するし, (2,3) 移動の均等攻撃は 7,8 個で全盤面をカバーしてしまうため隙間が大きすぎ適していない。結局, 均等攻撃で最も優秀なのは CB であると言える。

R 攻撃戦略の命中率は R 対 R において先手 20.8%, 後手 20.6% で, 全くランダムな場合の理論値 $100/100 \times (5+4+3+3+2) = 17\%$ に比べ少し大きい。これは R 戦略においても船がないことが自明な枡への攻撃をしなためと考えられる。

4.2 可能局面数え上げに基づく攻撃戦略の評価

戦略 MP, KMP, LA では数え上げを開始する未確定枡数が大きければ大きいほど攻撃確度が上がると考えられる。MP において数え上げ開始の未確定枡数を 10 から 80 まで 10 刻みで変化させたときの CB 戦略に対する勝率の変化を図 10 に示す。first は対 CB で MP を先手プレーヤとした場合, second は後手プレーヤとした場合である。予想通り, 可能局面数え上げに基づく攻撃戦略の優秀性が示され, 数え上げを早く開始すればするほど勝率が高くなった。ただ, 10 万局対戦で MP70 では 600 秒程度で終わるのに対し, MP80 では 13 時間程度を要しており実験時間がかかりすぎている。ただし MP80 での 13 時間も, 10 万局対戦で命中確率 38.5% 程度で全 17 枡の船を沈没させるので, 一着手当りの決定時間については

$$13 \times 3600 \div (100000 \times 17 \div 0.385) \approx 0.01 \text{ 秒}$$

となり, 人間の感覚的には無視できる時間に過ぎないことを補足しておく。

実験時間を考慮し, 可能配置数え上げに基づく攻撃戦略として MP70, KMP70, LA2.8.70 を選択し, 比較のため最も優秀だった均等攻撃戦略 CB も取り上げ, 相互に対戦させた。結果を表 4 に示す。表中の項目は表 3 と同じである。

MP70, KMP70, LA2.8.70 すべて, CB 戦略よりも優秀である。3 つの中では KMP70 が弱い。これは KMP70 のベースである KCB 均等攻撃 (図 5 参照) を行った後も縦横最大 4 枡の空きが連続しており, CB 均等攻撃 (図 4 参照) に比べ可能配置が少なくできないためと思われる。ただし KMP と MP の強さの差は表 3 にある KCB と CB の差ほど大きくない。これは可能配置数え上げ戦略の効果である。

また LA2.8.70 は, 実行時間から分かるように計算量がかなり増大するにもかかわらず, 勝率では MP70 と有意差が見られない。MP70 は LA1.x.70 に相当するが, さらに 1 手先読みを深くして LA2.8.70 にしてもあまり反駁が起らず, その数少ない反駁の効果も勝率に影響を与

えるほどは大きくないことを意味する。実際, LA2.8.70 と MP70 との対戦において LA2.8.70 で着手選択の際反駁の起こる比率を調べたところ 1.6% であった。

5. まとめと今後の展開

サイズ 2~5 の船 5 隻と 10×10 ボードを使う戦艦ゲームで, 配置をランダム戦略に固定して効率的な攻撃戦略について調べた。ボード全体を均等的にカバーする均等攻撃については市松模様攻撃が優秀であった。

完全情報問題の探索手法を応用し, 可能局面数え上げに基づく攻撃戦略を提案し評価したところ, 単なる均等攻撃に比べ優秀性を示した。この手法は, 不完全情報が状態が不確定だが進行に伴い状態が徐々に確定していった終了・解決するような問題 (多くのカードゲームなどの不完全情報ゲームや宝探し問題など) には有効だと考えられる。

実験においては, 市松模様攻撃を基本として攻撃がはずれになった場合の可能配置数を最小にする枡を攻撃する, すなわち, 当たる確率が最大の枡を攻撃する MP 戦略が最もコストパフォーマンスが優れていた。ただ, 未確定枡数が多いときは計算に時間がかかりすぎるのが難点である。これは可能配置の数え上げをインクリメンタルに計算すること等で効率化が期待でき, 今後の課題である。さらに, 数え上げをするのではなく, モンテカルロシミュレーションによって一定回数の配置試行を行い, 最も配置されやすい枡を求めて攻撃する戦略も考えられる。数手先まで探索し最も高い命中確率を与える枡を攻撃する LA 戦略も実装し 2 手先探索を評価したが, 上記の MP 戦略に対して優位性を確認できなかった。こちらでも探索の効率化を図れば, より深い広い探索をコストパフォーマンスが許す範囲で実現できる可能性がある。

今後, 数種の典型的な配置戦略と攻撃戦略との関係を調べる一方, 自動学習により近似的最適混合戦略を求め, 総合的に優れる攻撃戦略及び配置戦略を探っていく。

参考文献

- [1] D. Billings. The first international RoShamBo programming competition. *ICGA Journal*, 23(1):42-50, 2000.
- [2] D. Billings, N. Burch, A. Davidson, R. Holte, J. Schaeffer, T. Schauenberg, and D. Szafron. Approximating game-theoretic optimal strategies for full-scale poker. In *Proceedings of the 2003 International Joint Conference on Artificial Intelligence (IJCAI-03)*, pp. 661-668, 2003.

表 3: ランダムおよび均等攻撃戦略対戦結果 (100,000 ゲーム)

firstp	secondp	winrf	winrs	hitrf	hitrs	seconds
R	R	52.31(31)	47.69(31)	20.796(29)	20.619(29)	86
RNB	RNB	51.02(31)	48.98(31)	31.405(41)	31.489(42)	28
RNB	R	94.69(14)	5.31(14)	30.917(39)	18.162(33)	44
R	RNB	6.20(15)	93.80(15)	18.212(32)	30.949(39)	45
CB	CB	51.54(31)	48.46(31)	36.687(46)	36.782(46)	27
CB	RNB	70.71(28)	29.29(28)	36.403(45)	31.949(44)	27
RNB	CB	31.47(29)	68.53(29)	31.897(43)	36.463(45)	27
KCB	KCB	51.20(31)	48.80(31)	34.145(44)	34.238(44)	27
KCB	CB	39.88(30)	60.12(30)	34.655(45)	36.674(46)	27
KCB	RNB	57.57(31)	42.43(31)	33.669(43)	31.649(43)	28
CB	KCB	62.82(30)	37.18(30)	36.640(46)	34.840(45)	26
RNB	KCB	44.43(31)	55.57(31)	31.586(42)	33.817(43)	28

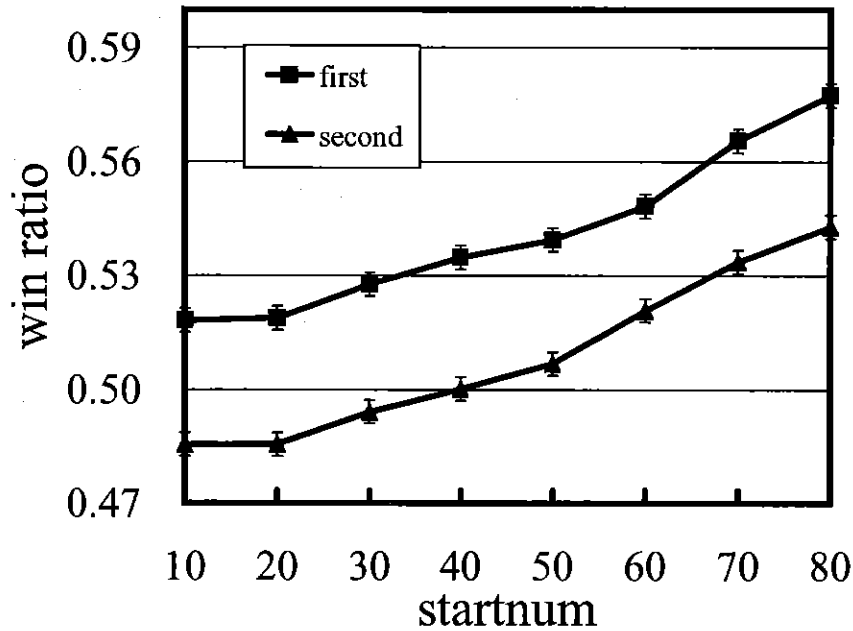


図 10: CB に対して MP で配置数え上げ開始桁数を変化させたときの勝率

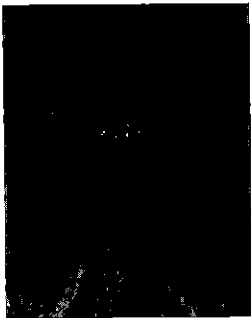
- [3] D. Billings, A. Davidson, J. Schaeffer, and D. Szafron. The challenge of poker. *Artificial Intelligence*, 134(1-2):201–240, 2002.
- [4] B. Bouzy. Associating domain-dependent knowledge and Monte Carlo approaches within a go program. *Information Sciences*, 175(4):247–257, 2005.
- [5] BURST! Media, LLC. Battleship Game. <http://www.artistic-designers.com/bkgds/puzzles/battleship/index.html>, 1999.
- [6] F-Active.com. Battleships – General Quarters II. <http://www.battleships.f-active.com/>, 1999–2005.
- [7] A. D. Fuegi. ARMADA. <http://scv.bu.edu/~aarondf/java/armada.html>, 1996.
- [8] M. L. Ginsberg. GIB: Steps toward an expert-level bridge-playing program. In *Proceedings of IJCAI-99*, pp. 584–589, 1999.
- [9] M. Matsumoto and T. Nishimura. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation*, 8(1):3–30, 1998.
- [10] R. Rasa. Battleship Info - Board Game Central. <http://boardgamecentral.com/games/battleship.html>, 2005.
- [11] 作田. 不完全情報ゲームの研究. *オペレーションズ・リサーチ*, 52(1):27–34, 2007.
- [12] 上原. コンピュータブリッジ. *人工知能学会誌*, 16(3):385–392, 2001.

表 4: 可能局面数え上げに基づく攻撃戦略対戦結果 (100,000 ゲーム)

firstp	secondp	winrf	winrs	hitrf	hitrs	seconds
MP70	MP70	51.65(31)	48.35(31)	38.156(47)	38.242(48)	1159
MP70	CB	56.71(31)	43.29(31)	38.016(47)	36.900(47)	614
CB	MP70	46.40(31)	53.60(31)	36.838(47)	38.130(47)	595
KMP70	KMP70	51.30(31)	48.70(31)	37.508(47)	37.630(47)	2670
KMP70	MP70	48.77(31)	51.23(31)	37.597(47)	38.144(47)	1919
KMP70	CB	53.64(31)	46.36(31)	37.457(47)	36.827(47)	1398
MP70	KMP70	54.36(31)	45.65(31)	38.099(47)	37.735(47)	1891
CB	KMP70	49.56(31)	50.44(31)	36.842(46)	37.598(47)	1309
LA2.8_70	LA2.8_70	51.76(31)	48.24(31)	38.152(47)	38.240(48)	8337
LA2.8_70	KMP70	54.26(31)	45.74(31)	38.090(47)	37.725(47)	5553
LA2.8_70	MP70	51.70(31)	48.30(31)	38.147(47)	38.245(48)	4740
LA2.8_70	CB	56.56(31)	43.44(31)	38.007(47)	36.923(47)	4234
KMP70	LA2.8_70	48.74(31)	51.26(31)	37.629(47)	38.195(47)	5425
MP70	LA2.8_70	51.54(31)	48.46(31)	38.120(47)	38.230(47)	4776
CB	LA2.8_70	46.09(31)	53.91(31)	36.811(47)	38.129(47)	4246

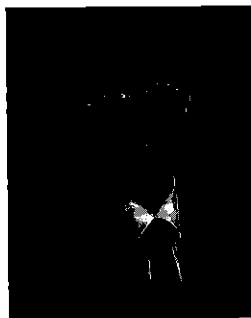
著者紹介

作田 誠



福岡工業大学情報工学部准教授。
博士(工学)。ゲーム・パズルを
題材とした探索・推論・学習な
どに興味を持つ。

飯田 弘之



日本将棋連盟プロ棋士六段。東京農工大学大学院博士後期課程
修了。博士(工学)。リンブルグ
大学コンピュータサイエンス学
科客員研究員、科学技術振興事
業団・博士研究員(電子技術総
合研究所勤務)、静岡大学情報
学部助教授、マーストリヒト大
学客員教授。科学技術振興機構
さきがけ研究員など。現在、北
陸先端科学技術大学院大学情報
科学研究科教授。

2007年 3月31日 論文受理
2007年12月19日 採録決定

