ゲーム情報学 6－10
(2001. 6. 7)

Dodgem ゲームのレトログレード解析による解明

小谷 善行

東京農工大学工学部
情報コミュニケーション工学科

Dodgem（ドッジェム）ゲームという２人で行うボードゲームを分析した。このゲームの局面空間が、駒の進行パターンという、推移の可能性よる半順序関係によって小さな多くの部分局面空間に分けられることをまず示した。さらにその一つ一つを終局面から始局面の方に向かって後退解析(retrograde analysis)する方法を設計した。今までの後退解析法は、終盤の限られた部分局面に適用する事例が多かった。しかし、本方法は、こうした多数の部分空間を定義できるゲームならばどれにも通用する一般的方法であるだけでなく、ゲーム全体を解く新しい手段を提供する。本システムの実現法としては、部分局面集合を単位としたキャッシュメカニズムを使った。

Retrograde Analysis of Dodgem Game

Yoshiyuki Kotani

Department of Computer, Information
and Communication Sciences
Tokyo University of Agriculture and Technology

The board game DODGEM, which is played by two, is analysed. It is shown that the game position space is divided into many small subspaces by the half order relation which is derived from the possibility of next-move transition, and whose element positions have the same material advancing patterns. A retrograde analysis method is designed on the basis of this subspace class, tracking them from the final position subspaces reversely. It is a new general mathod, applicable to any game where such a class of many subspaces can be defined. It also gives a new way to solve the whole game. To implement the system, we use a cach mechanism to make the subspaces the units of swapping.

１．はじめに

将棋や囲碁のような２人完全情報確定的零和ゲームを考えるとき、通常の解決法は$\alpha\beta$法に始まるゲーム木探索である。一方、終盤のゲーム空間が小さい場合、後退解析（ retrograde analysis ）と呼ばれる方法がある。チェス（１)、チェッカー、アワリなどのアフリカ系石置きゲーム（２)、象棋（３)、古い日本将棋（５）などは、駒の取り合いがあり、かつ駒を取ることなどのために、終盤に駒が少なくなり、結果として局面数が少なくなることがある。また、取った駒を再利用するなどのルールがないために、少なくなった駒が多くなることが論理的に生じないといえる。

したがって駒の数の同一性で区切った局面の部分空間を考えると、広い全体のゲーム局面空間に関係なく、その小さい部分空間を調べ尽くすことができる。チェスではそのデータベースもできている。

後退解析を成立させるためのキーになるのが、局面が単調減少（または増加）の特徴量をもつことであり、チェスではそれぞれの種類の、それぞれの側の駒数がそれにあたる。この特徴量が同一である局面集合が、解析の対象となる部分空間である。

ここでは、Dodgem（ドッジェム）というゲームを扱う。これは、チェスのように終盤に駒が少なくなるという性質をもつゲームであるが、駒数以上に顕著な特徴量があるということを見いだした。それは駒がどこまで進行したかのパターンである。

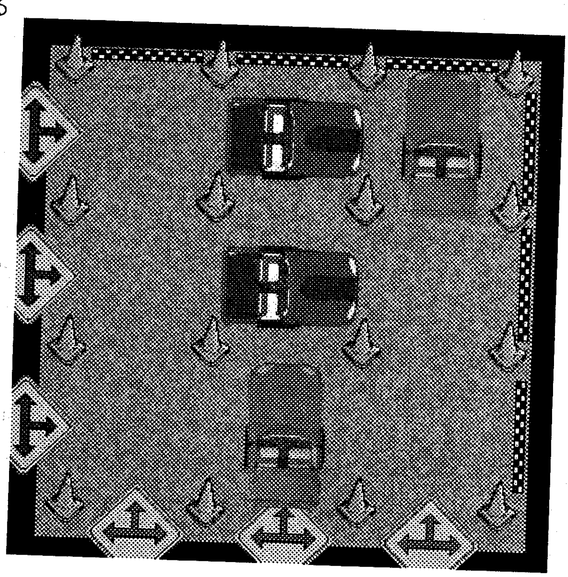
２．Dodgem ゲームのルール

図のように互いにｋ個の駒を、向かい合わせでなく、９０度の方向で配置し、縦横ｋ＋１の正方形の盤で戦う。初期配置は、このように手前の段に並べ、進行するにしたがって相手の進み先を妨害するようになっている。普通、２個ずつの駒で縦横３の正方形のものがよく遊ばれるが、もっと駒が多くて盤が大きい方が面白いかもしれない。便宜上、次に指す先手が下から上に向かい、後手が左から右に向かう形で図示する。先手の駒は上向き矢印↑、後手は右向き矢印→であらわす。

ルールは次の通りである。

１．駒はどれも同じで、指し手は前か右か左に１マス動かすことである。
２．初期配置の反対側の辺から外へ出る指し手ができる（その駒を取り除く)。
３．自分や相手の駒のあるところには行けない。
４．自分の番のとき、指し手がなくなれば勝ち。

図１．Dodgem の盤面
（ Zillions of Games (7) より）



勝つパターンで一番多いのが自分のすべての駒を外へ出した場合である（当然指し手はなくなる）が、そうでない場合つまり残った駒が相手に囲まれて指し手が無くなる場合もある。

基本的戦略は、なるべく自分の駒を前に進めて、外へ出すことである。逆にいうと、相手の駒の進行を妨げるようにすることである。ただそれがまずいこともある。

現在遊ぶことができるソフトウェアとしてはZillions of Games(7)があり、駒が２個、３個、４個のものがある（なお、このソフトウェアはたくさんのゲームが入っていてDodgemはその一つである)。駒数が多いゲームのほうが、人間にとって勝ちやすいようである。

３．駒の進行パターンとゲームの局面空間

先手後手それぞれについて、段（将棋の用語）にある駒の数の列をみる。段は先手側では、横方向の一列、後手側では縦方向の一列である。初期配置の段を第０段とし、１歩進むと第１段に入り、第ｋ段が反対側の辺であり、その上は盤の外である。

先手の第ｉ段にある駒の数をX_i、後手の第ｉ段にあるそれをY_iとし、それらを並べたもの

$$X_0X_1\cdots X_k : Y_0Y_1\cdots Y_k$$

を駒の「進行パターン」とよぶ。どの局面に対しても一つの進行パターンが定まる。逆に、一つの進行パターンは、一つの局面集合を与える。　たとえば、

□□□→
□□□□
→↑↑□
□□→□

という局面は、０２００：１０１１ という局面集合に含まれる。

駒数ｋ（辺がｋ＋１）のゲームでは、先手、後手のそれぞれの段の駒数パターンの数は組合せ${}_{2k+1}C_k$であたえられる。したがってその部分局面集合の数は${}_{2k+1}C_k$の二乗である。このようにこのゲームでは、全局面空間が非常に小さい部分局面空間に分けることができる。

４．親局面と子局面およびその進行パターン

それぞれの局面に対し可能な着手を１手指すことにより至る局面を子局面という。逆に１手戻して得られる局面を親局面という。たとえば、上の局面の親局面は次の５個がある（手番が変わるために裏返しになっている)。逆にこれらの局面の子局面の一つが上の局面である。

□□□↑	□→□↑	□□□↑	□□□↑	□□□↑
↑→□□	↑□□□	↑→□→	↑→□□	↑→□□
→□□□	□→□□	□→□□	□→□→	□→□□
□↑□□	□↑□□	□↑□□	□↑□□	□↑□→

親局面の構成法としては３種類ある。すなわち、駒を①一歩後ろに戻す、②よこに一歩動かす、③盤外に出たものを辺に戻す、である。上の例では①が先頭、②が２番目、③が３～５番目である。

当該局面の進行パターンの変化は次のようになる。$X_0X_1\cdots X_k : Y_0Y_1\cdots Y_k$というパターンに含まれる局面の親局面の進行パターンは、上記の①、②、③に対応して、

$$Y_0Y_1\cdots Y_k : X_0X_1\cdots(X_i+1)(X_{i+1}-1)\cdots X_k$$
$$Y_0Y_1\cdots Y_k : X_0X_1\cdots X_k$$
$$Y_0Y_1\cdots Y_k : X_0X_1\cdots(X_k+1)$$

の３種類がある。左右が入れ替わっているのはこれも手番が変わるためである。親局面の進行パターンの数は高々それぞれ、ｋ個、１個、１個で、計ｋ＋２個である。

例においては、親局面の進行パターンは

１０１１：１１００、１０１１：０２００、１０１１：０２０１

の３種である。

５．局面空間の組における半順序関係

進行パターンで規定される部分局面空間において、ある部分空間のすべての勝ち負け引き分けデータを生成するには、その子局面を含むすべての部分空間の勝ち負け引き分けデータがそろっていればよい。

この部分空間の計算が可能であるためには、部分空間間の親子関係がループになっていなければよい。進行パターンだけではループが存在する。すなわち上記の

Ｙ０Ｙ１・・・Ｙｋ：Ｘ０Ｘ１・・・Ｘｋ　及び　Ｘ０Ｘ１・・・Ｘｋ：Ｙ０Ｙ１・・・Ｙｋ

は相互に親子関係の局面対をもつ。これらを合体して「部分空間の組」と呼び、

Ｙ０Ｙ１・・・Ｙｋ　＊　Ｘ０Ｘ１・・・Ｘｋ４．

と書く。先手後手の進行パターンが同じ場合は、一つだけで組を構成する：

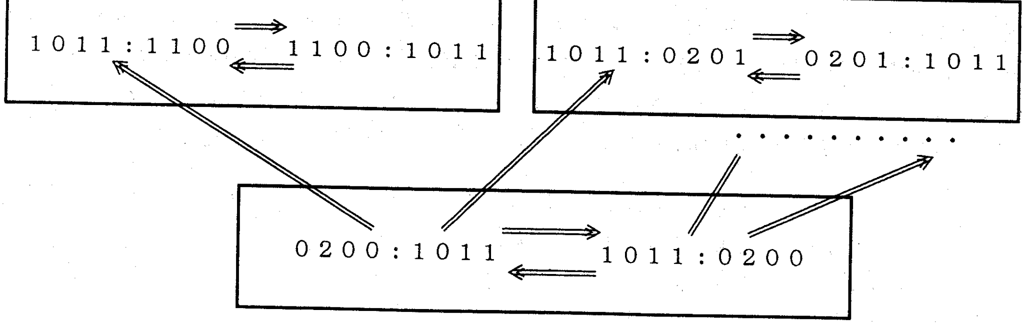
Ｘ０Ｘ１・・・Ｘｋ　＊　Ｘ０Ｘ１・・・Ｘｋ

ここで残りステップ数という数値を考える。これは駒の進行駒数の先手後手の合計である（ただし、盤から出た駒はステップ数ｋ＋１として数える)。である。この値は進行パターンから一意に決まる。そして、親局面を持つ部分空間の組の残りステップ数は、もとの組より１少ない。また子局面を持つ部分空間の組のそれは１多い。

従って部分空間の組単位で考えると、親子のつながりをもつ組間の関係はループが存在せず、部分順序関係になる。よって、子局面のつながりのない部分空間の組、すなわち終局面の空間の組からスタートして、つながっている組だけを参照して、逆順に勝ち負けパターンを、計算していくことができる。

上の局面の例では、部分空間及びのその組は図２のようになる。矢印の先が親の部分空間である。

図２．部分空間の親子関係



６．後退解析の方法

全局面空間を調べるという方針で、次のように後退解析方法を設計した。処理の単位は上で定めた局面空間の組である。目的は、各局面が必勝局面か、必敗局面か、それ以外かを確定させることである。

それ以外とは、この場合千日手すなわち、最善を尽くすと無限の繰り返しを生じる場合のことである。この３分類を、勝ち、負け、引き分けと呼ぶことにする。アルゴリズムの概略を述べる。

残りステップ数を１ずつ増やしながら以下を行う。

　その残りステップ数の局面空間の組それぞれについて次のことを行う。

　　１．それが終局面の空間の組ならその通りに勝ち負けのデータを生成する。

　　　通常の空間の組の場合、

　　　　新規の組なら、

　　　　　そのなかの局面すべてを生成する。

　　　　　これは進行パターンから求め、駒の重なりのある局面を排除することで得られる。

　　　　既知の組なら

　　　　　必要ならロードする。

　　２．その組のなかの空間の勝ち局面とわかる局面、すなわち指し手がない局面すべてにマークを付ける。

　　　　　この時点では、各局面は未定、勝ち、負けのどれかのマークがついている。勝ちマークについては、今指し手がない局面につけたものと、他の局面空間の組の処理により付けられたものがある。負け局面はその後者で付けられたものだけである。

　　３．その組のなかのすべての局面に対して次のことを行う。

　　　　それが負け局面ならその親局面すべてに勝ちのマークをつける。

　　　　（その空間の組が存在しなければ生成する。メモリになければロードする。以下同様）

　　４．その組のなかのすべての局面に対して次のことを行う。

　　　　それが勝ち局面ならその親局面すべてについて、その子局面すべてに勝ちマークが付けてあるか調べる。その場合、その親局面に負けのマークをつける。

　　５．以上を変化が無くなるまで繰り返す。

このアルゴリズムにより、順に開始局面を含む空間の組まで計算が進むことになる。

７．粗い部分局面空間

進行パターンの組は非常に多いため、観察しにくい。そのためにこの空間を合併させたものを考えることが重要である。

その一つは、上記の残りステップ数である。先手後手の残りステップ数の組はやはり局面空間を与える。これも半順序をなす。もう一つは、残りの駒数である。先手後手の残り駒数の組も局面空間を与える。これも半順序をなす。

なお、この両者に上下関係はない。これらで、結果を集計をとることとする。

８．実験結果

上記のアルゴリズムにより、駒数３までを計算した。その結果を述べる。
２駒（３×３）のゲームでは初期局面からはじめると、先手必勝のようである。

2 →□□	2 →□□	2 □→□
1 □→□	1 □□□	1 →□□
0 □↑↑	0 →↑↑	0 □↑↑
０１２	０１２	０１２
勝ち	勝ち	負け

１手進めたあとの勝敗は上のようになった。つまり、端の駒を進めると初期局面で勝てる。左の駒を動かすと負ける。

３駒（４×４）のゲームでは引き分け（千日手）のようである。盤を大きくすると引き分けがかなり増える。これはツークツワンク状態になっても手待ちできる駒がある場合が多くなるからであろう。ただし、端を突き合ったりするなど、ちょっとしたことで勝ちや負けになることがかなりある。

個々の進行パターンの結果も出ているが多すぎるので、それを粗い局面空間で集計したものを示す。

図３．残りのステップ数による
駒２（３×３）の場合の
引き分け、勝ち、負けの局面数
（上段から、引き分け、勝ち、負けの順、以下同様）

後＼先	1	2	3	4	5	6
6	0	0	0	0	0	0
5	0	0	0	0	0	0
4	0	0	0	0	0	0
3	0	0	0	0	0	0
2	0	0	0	0	0	0
1	0	0	0	0	0	0
6	7	12	22	21	16	4
5	21	37	64	58	34	4
4	28	49	82	88	21	0
3	29	50	110	28	14	2
2	15	38	7	4	0	0
1	8	0	0	0	0	0
6	0	0	1	0	0	1
5	0	0	4	5	13	12
4	0	0	9	28	42	21
3	0	2	29	63	54	21
2	0	3	45	45	37	12
1	0	15	29	28	21	7

図５．残り駒数による
駒２（３×３）の場合の
引き分け、勝ち、負けの局面数

後＼先	1	2
2	0	0
1	0	0
2	236	482
1	45	110
2	44	306
1	27	170

図４．同　駒３（４×４）の場合

後＼先	1	2	3
3	156	2760	23648
2	0	134	3807
1	0	0	349
3	7109	35684	91624
2	1610	6962	17548
1	146	546	417
3	106	9539	57432
2	224	4504	26628
1	94	1288	6605

図３．残りのステップ数による
駒３（４×４）の場合の
引き分け、勝ち、負けの局面数

	1	2	3	4	5	6	7	8	9	10	11	12
12	0	0	0	0	0	0	0	1	0	4	8	4
11	0	0	1	7	13	42	75	112	156	135	168	32
10	0	0	6	33	63	159	315	523	659	900	351	24
9	0	4	23	100	213	476	979	1455	2553	1289	425	24
8	0	1	4	7	28	192	558	1701	1891	1214	347	1
7	0	2	7	32	173	604	1491	1035	1785	874	217	1
6	0	3	13	64	302	744	678	738	1174	422	117	1
5	0	0	0	25	149	279	596	306	530	184	39	0
4	0	0	4	14	35	177	168	72	218	85	21	0
3	0	0	0	4	4	35	34	19	53	18	3	0
2	0	0	0	0	0	7	6	4	10	2	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
12	13	28	62	118	180	247	273	245	190	92	40	3
11	78	171	375	707	1073	1431	1549	1353	968	443	108	3
10	156	342	746	1395	2109	2787	2928	2369	1531	750	55	1
9	299	655	1422	2641	3959	5158	5184	3831	2878	418	46	0
8	385	849	1864	3547	5382	7092	6999	7512	1855	265	24	1
7	420	929	2044	3879	5760	6917	9665	2823	640	50	5	0
6	376	833	1833	3462	4856	8335	2833	811	66	0	1	0
5	268	600	1336	2491	5381	1659	466	57	0	0	0	0
4	170	384	847	2351	532	139	18	6	15	3	0	0
3	84	192	603	111	24	0	0	0	0	0	0	0
2	36	121	11	3	0	0	0	0	0	0	0	0
1	15	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	1	3	6	8	17	13
10	0	0	0	0	0	0	7	44	70	108	180	71
9	0	0	0	4	0	0	51	332	831	553	659	166
8	0	0	0	1	0	36	520	2710	1872	1457	1097	244
7	0	0	0	3	33	562	3838	4219	3789	2326	1403	272
6	0	0	0	6	234	2960	4572	5771	4394	2524	1355	246
5	0	0	0	58	1392	3454	4904	5047	3642	1988	1047	180
4	0	0	10	448	2007	3216	3728	3477	2512	1340	693	118
3	0	1	57	746	1308	1811	2017	1849	1392	734	373	62
2	0	5	182	381	600	829	925	846	649	340	171	28
1	0	36	84	170	268	376	420	385	299	156	78	13

このなかで、初期配置局面を含む状態空間を含むのはそれぞれ左上角のデータである。これでわかるのは、残りステップ数が少ないほうが勝つ場合が多い。また残りステップ数が同数の場合は先手側が勝つ場合が多い（当然である）。残り駒数でも同様である。
上の例では、通常のゲーム探索でできるかもしれない。

9．議論およびまとめ

今回、新しいスタイルで後退解析を行うする方法を設計した。従来の後退解析法は、チェスのエンドゲームなど、終盤の限られた部分局面に適用する事例が多かった。しかし、しかし、ゲーム全体が多数の小さい部分局面でカバーできるなら、全体を分析することも可能なのである。

それは、その部分空間を定義できるゲームであるかによる。うまく、半順序関係とその空間を見つけることが重要である。それができれば、ゲーム全体を解く新しい手段をこれにより得ることができる。

本システムの実現法としては、部分局面集合を単位としたキャッシュメカニズムを使った。本方法では、使用数の部分空間の組が主メモリにロードされるだけなので、二次記憶へのアクセスは少ない。通常のゲーム木探索とトランスポジションテーブルによる方法が二次記憶アクセスに適さないのとは異なる。

今後の問題としては、部分空間のうち、一部だけを計算することにより、初期局面に至り、ゲームを解くという方法として確立することが考えられる。

参考文献

(1) Ken Thomson, Retrograde Analysis of Certain Endgames, Journal of ICCA,vol. 9, pp.131-139, 1986.
(2) Roel van der Goot, Awari Retrograde Analysis, Proc. of Conference on Computers and Games, pp.15-24, 2000.
(3) Hwa-ren Fang, Tsan-sheng Hsu, Shun-chin Hsu, Construction of Chinese Chess Endgame Databases by Retrograde Analysis, Proc. of Conference on Computers and Games, pp.25-44, 2000.
(4) 松田道弘編、世界のゲーム事典、東京堂出版、1989.
(5) Hiroyuki Iida, Kazuro Morita, Jin Yoshimura, J.W.H.M. Uiterwijk Retrograde Analysis of the KGK Endgame in Shogi; its implication for ancient Heian Shogi ,First International Conference, CG'98, Tsukuba, Japan, November 1998.
(6) R. Lake, J. Schaeffer and P. Lu Solving Large Retrograde-Analysis Problems Using a Network of Workstations , Advances in Computer Chess 7, 2000.
(7) http://www.zillions-of-games.com/