

基于数学形态学的围棋地域划分算法^{*})

岳 鹏 李太华 邱玉辉

(西南大学智能软件与软件工程重点实验室 重庆 400715)

摘 要 地域是人类棋手必不可少的抽象概念,计算机围棋可以建立此概念以提高搜索效率。本研究在数学形态学的形式化基础上探讨了划分我领土、我领空、我领海、敌领土、敌领海、敌领空、公海、公空及未定区等 9 种地域的过程,并结合实战对局评价了此算法的性能,指出了此算法的应用价值。

关键词 数学形态学,计算机围棋,地域

Go Terrain Partition Algorithm Based on Mathematical Morphology

YUE Peng LI Tai-Hua QIU Yu-Hui

(The Key Laboratory of Intelligent Software & Software Project, Southwest University, Chongqing 400715)

Abstract Terrain is the essential abstract concept of human go player. Computer go can establish this concept to improve its searching efficiency. This study, based on the formalization of mathematical morphology, discusses the process of partitioning the 9 terrains: my territory, my airspace, my sea, his territory, his airspace, his sea, public airspace, public sea and undecided area. This paper, by linking with the actual combat, evaluates the capability of this method, and at the same time points out its application value.

Keywords Mathematical morphology, Computer go, Terrain

围棋是形象思维和逻辑思维兼备的游戏,地域是人类棋手必不可少的抽象概念^[1,2],盘面上尚未落子之处在人类棋手心目中可能已有所属。实战中,不同地域的转换取舍体现了棋手的水平高低。地域概念有其直观性,可以应用数学形态学方法来划分。数学形态学(Mathematical morphology)自 20 世纪 60 年代兴起以来,已广泛应用于图像处理如边缘提取、骨架提取及图像分割等各方面^[3],它形象直观,简化了图像处理,量化并保持了物体的主要形状特征;在围棋领域也有应用^[4]。

1 棋群与地域

棋局状态包括轮下方,棋子分布和劫点,劫点即为了防止全局同形再现而规定的禁着点。特定棋局状态的棋子分布可以分解为两个二值图像的组合^[5],以 19 路盘为例,即为两个 19×19 大小的二值图像,称为黑盘面(blackboard, B)和白盘面(whiteboard, W)。棋局上的棋子可以分解为棋块的组合,棋块指彼此直连的同色棋子。棋盘上与棋块直连的空白交叉点称为气(liberty),围棋规则规定“气尽棋亡”,所以同一棋块内的棋子同生同死。棋群有别于棋块,指的是由一个或多个棋块组成的功能集合体,常常作为一个整体参与战斗,也可能包括内部的对方死子。孤子可以看作特殊的棋群。

计算机围棋完成棋群聚类后,可继续进行地域划分。根据敌我及位置关系可以将地域大致分为 9 种:我领土、我领空、我领海、敌领土、敌领海、敌领空、公空、公海及未定区。其中前 6 种地域是依附于特定棋群的。核心概念是领土、领海及领空。特定棋群所包含的棋子形成的凸包的内部区域称为领土(territory, T),与棋盘边角(3 线及 3 线以下)之间的区

域称为领海(sea, S),棋群势力影响范围内的区域称为领空(airspace, A)。图 1 是业余二段棋手对这 3 种地域概念的理解。针对盘面上每个棋群求得领土、领海及领空后,其余的位于中腹的大片区域则为公空(public airspace, Pa),靠近边角的大片区域则为公海(public sea, Ps),剩下的小片区域(小于 3×1)则为未定区。这 9 种地域概念形象直观,在空间上有一定的相互依赖性,可以用数学形态学方法来划分。

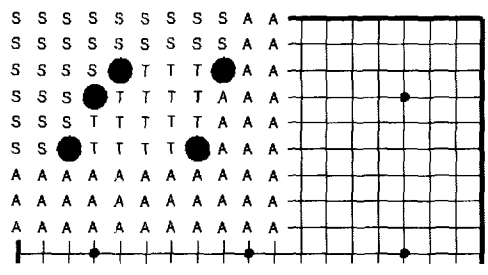


图 1 领土、领海及领空

2 膨胀与腐蚀运算

二维欧氏空间 e^2 上的二值图像中属于物体的像素构成了一个集合 X ,原点的坐标为(0, 0), X 中的任一点都可以看成是一个关于原点的向量。一个数学形态学变换由原始图像和另一个称为结构元素(structuring element, Se)的小点集之间的关系定义, Se 表示为一个关于基准点的区域,此基准点可以不属于 Se 。图 2 所示即为三个结构元素,标有 \square 处为基准点,其中第三个结构元素的基准点不属于此结构元素。数学形态学有膨胀、腐蚀等基本运算,这些运算的定义尚不完全

^{*})岳 鹏 博士生,研究方向为博弈。李太华 博士生,研究方向为计算机辅助教学。邱玉辉 博士生导师,研究方向为模糊逻辑、多 Agent 系统、博弈及网格计算等。

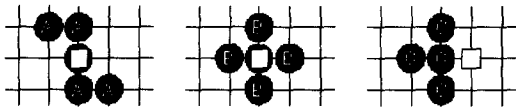


图2 结构元素

一致^[4]。这里,膨胀运算 \oplus 采用向量加法对两个集合进行合并,如 $(a, b) + (c, d) = (a + c, b + d)$,腐蚀运算 \ominus 对集合元素采用向量减法对两个集合进行合并,即

$$X \oplus Se = \{p \in e^2; p = x + s, x \in X, s \in Se\} \quad (1)$$

$$X \ominus Se = \{p \in e^2; p + s \in X, s \in Se\} \quad (2)$$

采用各向同性的结构元素进行膨胀运算可以描述为一个将所有与物体邻近的背景像素变为物体像素的变换。它可以用来填补物体中小的空洞和缝隙,改变了原来集合的连通性。采用各向同性珠结构元素进行腐蚀运算可以描述为原物体的收缩,还可以用来寻找物体的轮廓。

3 地域划分算法

以专业棋手实战对局中出现的棋形为例说明地域划分算法。求领土的基本运算类似于凸包运算。凸包的精确算法^[3]因包含较多的乘除运算比较耗时,而基于数学形态学的凸包算法可以克服此缺点。结构元素采用图2居中所示,即各向同性的4-领域结构,连续膨胀 n 次再连续腐蚀 n 次可求得凸包,即为领土。如图3为膨胀5次再腐蚀5次后的结果,标记区域即为领土。当棋群所包含的棋子呈弯曲带状分布时,这种算法和精确算法所求得区域往往相差较大,但前者更符合人类棋手的理解。

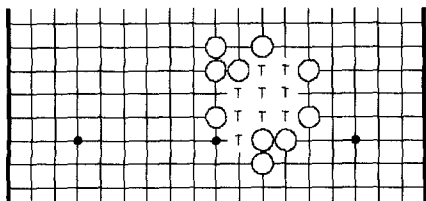


图3 领土

注意到领海概念的实质是为了强调边角易形成实地的特性而提出的,所以稍微对腐蚀运算作一变化(气为0的点不被腐蚀),为区别起见称为吸附运算 \odot ,即

$$X \odot Se = \{p \in e^2; p + s \in X, s \in Se \text{ or } \text{liberty}(p) = 0\} \quad (3)$$

这样,对棋群连续膨胀 n 次再连续吸附 n 次可求得领海和领土的混合区域,减去已经求得的领土,即可得到领海。如图3即为膨胀5次再吸附5次然后减去领土求得领海的结果。



图4 领海

以棋群与领海、领空组成的混合区域为原始图像连续膨胀若干次后,求得的新增区域即为领空。一般说来,开局阶段膨胀次数可稍大些如取5,中盘阶段可取小些。如图5所示,标记区域为膨胀次数取为3时求得的领空。一般说来,当同时考虑双方棋群求领土、领海及领空时,膨胀次数应该取小些,可以采用受限的膨胀运算达到这种适应性,即双方同时作

膨胀运算,某交叉点如已有对方膨胀点到达时则不予膨胀。对于双方同时到达的交叉点组成的棋块,根据其与黑、白双方相邻的实子数决定归属:属于相邻的实子数多的一方,相等时置为背景。这样可以保证所求得的地域不会有重合。针对盘面上每个棋群求得领土、领空和领海后剩下的区域即为公海、公空和未定区,根据定义可直接求得。

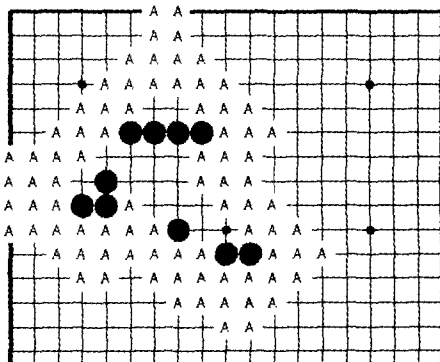


图5 领空

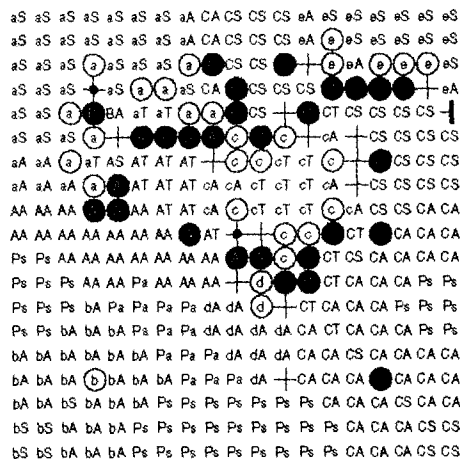


图6 最终划分结果

最后可以采取一些修正措施,如未定区可根据启发式策略归入邻近区域,或对区域边缘作光滑化处理。图6为此实战对局最终的划分结果,基本符合业余二段棋手的对地域概念的理解。其中,黑盘面由①、②、③个棋群组成,白盘面由④、⑤、⑥、⑦、⑧个棋群组成,标记BT、BS、BA即表示棋群①的领土、领海、领空,Ps为公海,Pa为公空,未定区无标记。

4 实验结果与讨论

以专业棋手实战对局100谱检测此算法,65%的划分结果符合业余二段棋手的理解,说明此算法是行之有效的。地域概念是指导棋手选择着点的必不可少的概念。布局阶段选择大场时应该在公共领空或公海里选择着点;补棋时多在领土中选择,或选择某个能使领土扩大的点以有足够眼位。中盘阶段则可根据对方走步所属区域大致确定走步意图:如对方子落在我方领海则可能为搜根攻击,落在领空则可能为封锁逼活。在形势评估时可直接采用这些地域概念所含的交叉点数以估计某方拥有区域的大小。计算机围棋只有在建立了地域这种抽象概念之后,才可能进一步应用这些启发性策略^[6,7]。

基于数学形态学的地域划分算法计算量小,可改变的参

数多,可根据具体应用环境适当改变膨胀或腐蚀准则,适应性
强,可以为后续的战略搜索提供进一步支持。

参 考 文 献

- 1 Niu Xiaozhen. Recognizing safe territories and stones in computer Go:[Master's thesis]. Department of Computing Science, University of Alberta, 2004
- 2 Niu Xiaozhen, Müller M. An improved safety solver for computer Go. In Computers and Games 2004, Ramat-Gan, Israel, 2004
- 3 Sonka M,等著. 图像处理、分析与机器视觉. 艾海舟,等译. 北京: 人民邮电出版社,2003

- 4 Bouzy B. Mathematical morphology applied to computer Go. International Journal of Pattern Recognition and Artificial Intelligence, 2003,17(2)
- 5 Zobrist A. A model of visual organization for game of Go. In: Proceedings of the Spring Joint Computer Conference, 1969, 34: 103~112
- 6 Bouzy B. History and territory heuristics for Monte-Carlo Go. In: Joint Conference on Information Sciences, Salt Lake City, 2005
- 7 Uiterwijk J W, van der Werf E C, van den Herik H J. Learning to estimate potential territory in the game of Go. In: 4th International Conference on Computers and Games, 2004

(上接第 165 页)

基于密度的聚类方法,很容易将此数据集划分为 2 个类,实际上此数据集中只存在 1 个主要聚类。造成聚类错误的主要原因是采用数据点计数计算网格单元的密度造成了数据点之间联系的丢失,尤其是对处于网格单元边界部分的数据点更是如此。此时再进行基于密度的聚类计算,很难将相对稀疏网格单元与纯噪音的网格单元相区分。相比较而言,本文方法的数据预处理过程可以有效提高聚类算法对相对稀疏网格单元的识别能力。从图 2d 中可以看到,此例中用本文方法的数据预处理过程计算得到的网格单元密度,其密集网格单元与相对稀疏单元之间的密度比值最高也没有超过 2:1。

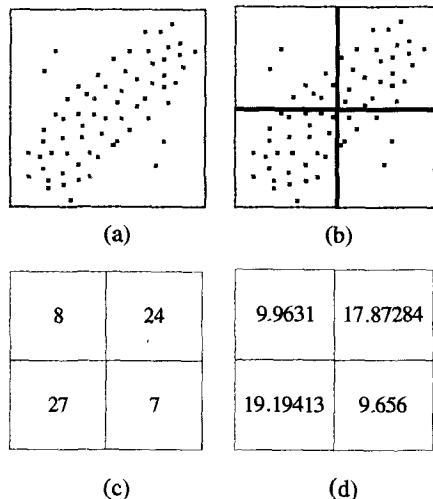


图 2 密度计算对比图

另外,为了验证本文提出的算法并方便观察,本文采用多个 2 维数据集对算法进行了实验验证。典型的实验数据集及实验结果图如图 3 所示。

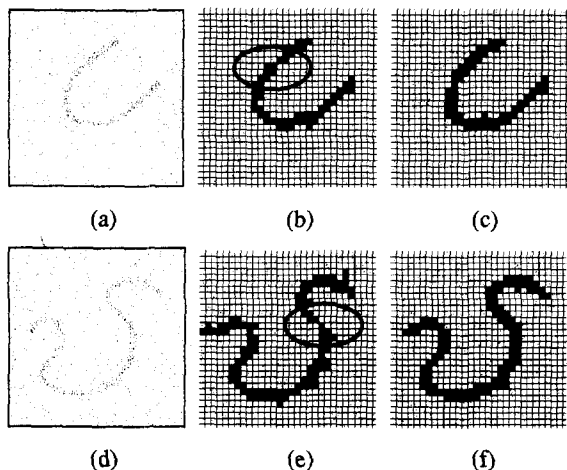


图 3 对任意形状的聚类结果

图 3a 所示数据集中包含 366 个 U 形聚类数据点以及 600 个随机的噪音数据点。图 3d 所示数据集中包含 429 个 S 形聚类数据点以及 600 个随机的噪音数据点。在每个实验中,每个维度被分为 30 个单元。图 3b 和图 3e 显示的是利用简单计数方法计算网格单元密度后得到的聚类结果。在结果图中可以看到 U 形聚类和 S 形聚类均被分割为两个聚类(分割点在图中椭圆处)。而在相同的设置条件下,采用本文的方法则可以得到正确的聚类结果(如图 3c 和图 3f 所示)。

通过以上实验可知,相对于数据点简单计数的网格单元密度计算方法,本文在数据预处理过程中所采用的网格单元密度计算方法具有更好的精度。同时,这种算法能够发现任意形状的聚类结果,对于噪音不敏感。

结论 本文分析了常用的网格单元密度计算方法的不足,借鉴密度函数思想,提出了一种新的网格单元密度的计算方法,并将其与微粒群算法相结合,实现了一种基于网格和密度的微粒群混合聚类算法。实验结果说明,相对于常用的网格单元密度计算方法,这种新的网格单元密度计算方法更能保留数据点之间的联系,具有更高的精度,并可以推广应用到其它基于网格的聚类方法中。当然,这种方法也存在一定的局限性,在应用时需要网格单元边长 Eps_i 和阈值 ξ 两个参数进行人为设定。两个参数的设定情况会较明显地改变聚类的结果,这也是将来研究工作的重点。

参 考 文 献

- 1 Han Jiawei, Micheline K. 数据挖掘概念与技术[M]. 范明,孟小峰,等译. 北京:机械工业出版社,2001. 223~262
- 2 Kennedy J, Eberhart R C. Particle swarm optimization[C]. In: Proceedings of the 1995 IEEE International Conference on Neural Networks, Piscataway, NJ, Perth, Australia: IEEE service center, 1995. 1942~1948
- 3 CHEN Ning, CHEN An, ZHOU Long-xiang. An Incremental Grid Density-Based Clustering Algorithm[J]. Journal of Software, China, 2002, 13(1): 1~7
- 4 LI Cun-Hua, SUN Zhi-Hui. A Mean Approximation Approach to a Class of Grid-Based Clustering Algorithms. Journal of Software, China, 2003, 14(7): 1267~1274
- 5 王鹏,曾振柄,谢千河. 采用蚁群爬山法进行聚类分析的算法. 计算机工程, 2003, 29(10): 79~80
- 6 谭勇,荣秋生. 一个基于 DBSCAN 聚类算法的实现. 计算机工程, 2004, 30(13): 119~121
- 7 陈梅兰. 基于网格和密度聚类算法研究. 计算机与现代化, 2005, 2: 1~9
- 8 朱倩,黄志军. 一种改进的基于密度和网格的高维聚类算法. 船舶电子工程. 2005, 25(5): 55~59
- 9 王生生,刘大有,曹斌,等. 一种高维空间数据的子空间聚类算法. 计算机应用, 2005, 25(11): 2615~2617