控制理论与应用
Control Theory & Applications
ISSN 1000-8152,CN 44-1240/TP

《控制理论与应用》网络首发论文

题目：　实时格斗游戏的智能决策方法
作者：　唐振韬，梁荣钦，朱圆恒，赵冬斌
收稿日期：　2021-10-19
网络首发日期：　2022-07-26
引用格式：　唐振韬，梁荣钦，朱圆恒，赵冬斌．实时格斗游戏的智能决策方法[J/OL]．控制理论与应用．
https://kns.cnki.net/kcms/detail/44.1240.TP.20220726.0957.002.html

实时格斗游戏的智能决策方法

唐振韬, 梁荣钦, 朱圆恒†, 赵冬斌

(1. 中国科学院自动化研究所 复杂系统管理与控制国家重点实验室, 北京 100190;
2. 中国科学院大学 人工智能学院, 北京 100049)

摘要: 格斗游戏作为实时双人零和对抗博弈的代表性问题, 具有实时对抗和快速响应的重要研究特性. 相应针对性方法的提出有效反映了游戏人工智能领域的重要研究进展及发展方向. 本文以格斗游戏人工智能竞赛作为研究背景, 将智能决策方法分为启发式规则型、统计前向规划型与深度强化学习型三大类型, 介绍相应的智能决策方法在实时格斗游戏中的研究进展. 为分析格斗游戏智能决策方法的表现性能, 本文提出了胜率、剩余血量、执行速率、优势性和伤害性的5个性能因子, 系统分析智能决策方法的性能优势及不足. 最后, 对未来的在格斗游戏中研究发展趋势进行展望.

关键词: 实时格斗游戏; 统计前向规划; 深度强化学习; 性能因子; 智能决策

引用格式: 唐振韬, 梁荣钦, 朱圆恒, 等. 实时格斗游戏的智能决策方法. 控制理论与应用, 2022, 39(6): 969 – 985

DOI: 10.7641/CTA.2022.10995

Intelligent decision making approaches for real time fighting game

TANG Zhen-tao, LIANG Rong-qin, ZHU Yuan-heng†, ZHAO Dong-bin

(1. The State Key Laboratory of Management and Control for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China;
2. School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Fighting game is a classical real-time two player zero sum game, which has the obvious characteristics of real-time confrontation and extremely rapid decision response. Research and study conducted on this platform reflects the important research progress and development direction in the field of game artificial intelligence. In this paper, we focus on the application and development of intelligent decision-making in real-time fighting games. Methods that are applied in fighting games are categorized into three approaches, including heuristic rules, deep reinforcement learning, and statistical forward planning. Their corresponding process and development in the field of real-time fighting games is deeply studied. In order to systematically show the superiority and inferiority of different methods, five key metrics are proposed to analyze their performance, including win rate, remaining hit points, speed of wining, advantages of wining, and damage to the enemy. After the systematic analysis, the potential directions of intelligent decision-making in real-time fighting games are concluded for future research.

Key words: real-time fighting game; statistical forward planning; deep reinforcement learning; key metric; intelligent decision

Citation: TANG Zhentao, LIANG Rongqin, ZHU Yuanheng, et al. Intelligent decision making approaches for real time fighting game. *Control Theory & Applications*, 2022, 39(6): 969 – 985

1 引言

游戏平台因具有安全、快速、低成本、可复现以及对抗性等显著特点, 已经成为人工智能(artificial intelligence, AI)研究的重要验证及测试平台. 对抗类型游戏根据决策行为的执行方式, 可划分为轮流决策的回合制游戏[1]和同步决策的即时制游戏[2]两种基本类型.

回合制游戏根据回合制顺序, 要求玩家在己方回合内制定决策行为, 一般会给予玩家充分的思考时间制定策略方案. 大多数棋牌类游戏属于典型的回合制游戏. 针对回合制游戏, 以深度强化学习和统计前向规划为代表的智能决策方法, 无论是在完全信息环境

收稿日期: 2021−10−19; 录用日期: 2022−02−22.
†通信作者. E-mail: yuanheng.zhu@ia.ac.cn; Tel.: +86 10-82544764.
本文责任编委: 方浩.
科技部科技创新2030“新一代人工智能”重大项目(2018AAA0101005), 中国科学院战略性先导研究项目(XDA27030400), 中国科学院青年创新促进会项目(2021132)资助.
Supported by the National Key Research and Development Program of China (2018AAA0101005), the Strategic Priority Research Program of Chinese Academy of Sciences (XDA27030400) and the Youth Innovation Promotion Association CAS (2021132).

下的国际象棋AI (深蓝[3])和围棋AI (AlphaGo[4]与AlphaGo Zero[5]), 还是在非完全信息环境下的德州扑克AI (Libratus[6]与Pluribus[7]) 和麻将AI (Suphx[8]), 皆取得了备受瞩目的成就. 而针对即时制游戏, 王者荣耀AI (绝悟[9])、Dota2 AI (OpenAI Five[10])和星际争霸II AI (AlphaStar[11])均已达到甚至超越人类顶尖职业玩家水平. 综上, 面向对抗类游戏的智能决策方法研究已经在众多相关科研领域产生了举足轻重的影响, 但是美中不足的是上述研究对象通常需要依赖海量的数据信息和强大的计算资源作为支撑, 并且研究过程需要面临大量的工程技术性问题的挑战, 因而制约其在相关领域的有效复现和快速推广.

与回合制游戏相比, 即时制游戏要求智能体根据实时动态变化的系统环境以及对手策略制定有效及时的决策响应. 格斗游戏作为典型的实时策略同步博弈问题, 具有实时决策的基本需求, 并且可以有效降低实践工程难度, 使研究人员聚焦于博弈方法的设计及策略目标问题的优化, 有助于向相关研究领域进行推广和应用.

本文以格斗游戏人工智能竞赛平台[12]作为主要研究背景, 旨在介绍格斗游戏领域的研究脉络及进展, 重点分析该领域关键性方法所发挥的重要性作用, 并且展望面向实时格斗游戏的智能决策方法发展方向. 本文的主要贡献点为: 1) 系统分析了格斗游戏AI算法特点, 并且首次明确给出了格斗游戏AI的算法分类与模型特性; 2) 详细阐述了历届优秀格斗游戏AI的算法特点, 分析并展望格斗游戏AI算法的发展趋势; 3) 率先提出了性能因子, 以此作为模型性能测试和验证的重要工具, 全面地评估格斗游戏AI的决策风格和性能.

本文的篇章结构为: 第2节介绍实时格斗游戏的问题描述及特点; 第3节梳理格斗游戏研究的发展脉络; 第4节通过格斗游戏人工智能竞赛分析各类方法的应用特性及效果; 第5节思考并且展望用于解决实时格斗游戏问题的智能决策方法的主流研究方向及存在问题; 第6节对本文进行总结.

2 实时格斗游戏问题描述与特点

实时格斗游戏是一个极具挑战性和观赏性的即时制游戏问题. 要求智能体在极短的反应时间内, 从大量候选动作集中做出有效选择. 实时格斗游戏一般采用1对1的对抗形式, 初始血量值设置为固定数值, 初始能量值为零, 可以通过有效击打对手获取能量值, 并且根据累积能量值采取高额伤害动作, 最终凭借血量差的优势战胜对手. 根据动作空间维度, 格斗游戏可分为2维空间和3维空间两种类型. 尽管动作空间维度越高致使问题规模复杂度越高, 然而问题对象的特性与本质并未发生变化, 因此本文重点以2维空间下的格斗游戏场景作为研究背景.

格斗游戏通常以对抗双方的血量、能量、历史动作以及相对距离等信息作为模型观测状态. 动作信息主要为动作的伤害量、击打区域、消耗能量以及持续帧数等因素组成. 评分指标主要为双方的剩余血量和对抗剩余时间. 根据对抗的激烈程度可将策略模型分为激进型、中立型与保守型. 如图1所示, 实时格斗游戏问题的主要特点可归结为快速反应性、同步性、动作连续性以及角色属性多样性等, 具体表示为:

• 快速反应性: 要求智能体在极短的反应时间内做出决策. 假定即时制游戏的更新频率为60帧每秒, 则智能体的最长反应时间约为16.67 ms. 因此, 格斗游戏要求决策方法进行高效地决策行为.

• 同步性: 即时制游戏双方的决策过程是同步而非轮流进行. 体现在智能体无法从系统环境获悉当前对手采取的决策行为, 由此构成不完美信息博弈, 影响智能体的决策行为有效性.

• 动作连续性: 格斗游戏中的动作执行需要连续消耗一定帧数. 但是在动作执行过程中会受到对手行为的影响而被迫中断, 影响动作的执行结果, 从而干扰智能体的评估准确性.

• 角色属性多样性: 由于不同格斗角色的动作属性要求不同, 因而产生不同的动作组合策略, 使得格斗游戏模型需要具有角色属性泛化性和系统环境适应性.

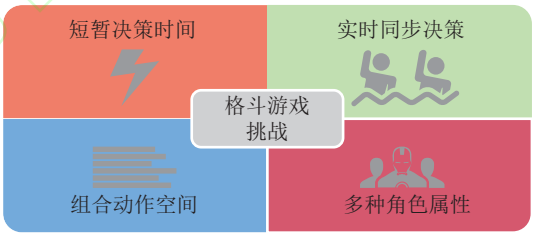


图 1 格斗游戏AI面临的挑战

Fig. 1 Challenges of fighting game AI

早期基于启发式规则的方法, 对复杂的状态空间进行特征压缩, 并且利用专家知识构建决策系统以应用到实时格斗游戏任务. 然而, 随着游戏复杂度的增加, 系统推理时间随之增长, 相应的专家规则系统设计愈发困难. 有趣的是人类玩家却可以在很短的时间内, 高效快速地学习出实时格斗游戏策略. 基于此, 科研人员设计了大量的实时决策模型以应对格斗游戏带来的挑战, 以期实现通用且高效的格斗游戏AI, 推动通用型AI的研究和发展.

3 格斗游戏人工智能方法

纵观格斗游戏人工智能方法研究进展, 其中采用的大多数人工智能方法可溯源于棋类游戏. 棋类游戏是人工智能方法早期的重点研究与应用的博弈任务. 20世纪50年代, Turning提出了极小化极大值算法(Mini-Max)并且成功应用到国际象棋[13]. Samuel基于

强化学习方法通过自我博弈的方式学习到国际跳棋的策略[14]. 1992年, Tesauro基于神经网络策略模型, 通过大量自我博弈的方式在系统环境进行数据采样, 并且采用时间差分强化学习方法[15]优化策略模型, 研制出TD-Gammon[16]在西洋双陆战棋上战胜人类顶尖选手. 1997年, IBM研制的国际象棋AI深蓝[3], 有效结合高质量专家经验规则与高性能计算搜索技术, 首次战胜人类职业冠军(Gary Kasparov)取得了里程碑式的进展. 纵观上世纪游戏AI的研究进展, 研究方法主要集中在2个方向: 第1类是在专家知识的基础上构建启发式规则系统, 设计高效的最优解搜索算法; 第2类是在机器学习方法的基础上构建策略模型, 通过交互数据驱动的方式优化模型决策过程. 在早期硬件计算资源相对落后且算力不足的情况下, 通常基于规则约束的方式减小问题解空间, 然后在约束解空间下通过启发式搜索找到可行最优解. 随着硬件计算性能的不断提升和数据信息存储的持续增加, 基于数据驱动和环境交互的最优化算法正发挥着举足轻重的作用. 2016 年, 谷歌DeepMind团队在围棋取得重大里程碑式进展(AlphaGo[4]), 通过人类专家数据初始优化深度神经网络模型, 结合深度强化学习、蒙特卡罗树搜索与自我博弈的方式, 迭代优化策略神经网络参数和价值神经网络参数, 最终首次战胜人类职业9段顶尖选手.

与此类似, 格斗游戏问题的研究具有相似的发展历程, 同样经历了从启发式规则系统到数据驱动式智能决策方法的研究历程. 格斗游戏以即时制决策作为研究背景, 要求博弈模型准确分析当前局势以及对手行为, 制定有效合理的决策行为, 实现击败对手策略的目标. 根据格斗游戏AI的时间发展脉络, 核心的格斗游戏方法可划分为启发式规则型、统计前向规划型(可分为蒙特卡罗树搜索和滚动时域演化方法两类)和深度强化学习型的3种基本类型. 接下来依次介绍相关类型方法.

3.1　启发式规则型

启发式规则型方法直接从环境因素进行建模, 基于专家经验来定义一系列规则集合, 通过规则设计对状态空间进行分类, 由此降低搜索空间范围, 提升模型实时性决策效率. 启发式规则型的规则映射可抽象定义为

定义 1　定义规则映射$R: Z \to B \times A$, 其中Z表示可能出现的游戏状态, $B \in \{0, 1\}$表示是否执行对应动作, A表示对应的可选动作集合. 当$R(z) = (1, a)$时, 表示满足规则z时执行对应动作a.

定义 2　定义$E(R) = \{z \in Z | z$满足$R\}$表示满足规则映射R的游戏状态z. 当$E(R) = \emptyset$时, R为空规则. 反之, 当$E(R) = Z$时, 则R为默认规则.

定义 3　定义$S = \{R_1, \cdots, R_n\}(n \in N)$为有序规则集, 脚本优先级表示为$\mathrm{Prio}: S \to Z$. S为脚本时满足$U_{R \in S} E(R) = Z, \mathrm{Prio}(R_1) \leqslant \cdots \leqslant \mathrm{Prio}(R_n)$.

由此可知, 启发式规则型方法采用预先定义的规则集合, 将游戏状态空间直接映射到角色的候选动作集合. 当智能体的决策动作只能找到唯一对应的规则指引时, 启发式规则法退化为典型脚本方法. 脚本类型方法根据当前游戏状态和对应规则, 按照规则集合的优先级高低生成相应的动作执行序列, 随后按顺序执行动作序列直至对抗过程结束或动作执行完成.

启发式规则型方法在大量游戏AI设计中得到广泛应用, 具有可解释性、可调试性及鲁棒性等优势[17]. 实时格斗游戏通常需要考虑许多环境因素作为决策参考依据, 具体包括对战双方的角色攻击属性、位置、血量、能量、移动方向、相对距离、当前动作和历史动作等. 此外, 需要考虑系统延迟和状态随机转移过程.

作为启发式规则型方法的典型代表, 动态脚本法[18−19]的核心思想是根据智能体在环境中的表现, 动态调整智能体的策略优先级权重以适应环境变化. 动态脚本法预先根据经验构建专家规则库以应对游戏状态的不同需求. 然后按照规则库中的规则权重值高低, 通过轮盘赌算法选择对应规则集合, 并且令AI决策系统根据规则集合制定角色采取的具体行为. 最后根据动作决策结果, 再次评估并调整规则库中的优先级权重, 以适应环境属性要求, 具体算法流程如图2所示. 虽然动态脚本法可通过环境交互的方式适应对手模型策略, 然而该方法较难快速调整并适应多变的对手策略模型, 致使其在与人类玩家对抗时的表现情况不佳. 此外, 若专家规则库设计不当同样会对算法性能造成严重影响.

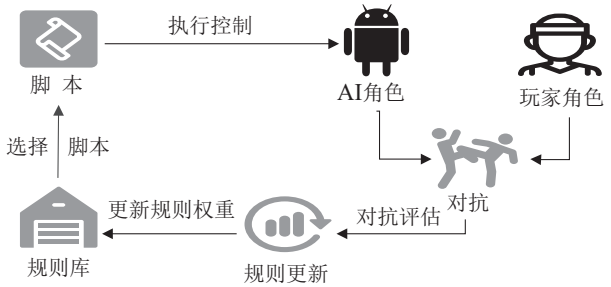


图 2　动态脚本法

Fig. 2　Dynamic Scripting

为解决实时格斗游戏动态变化带来的策略模型环境自适应问题, Spronck等[20]提出自适应动态脚本法, 引入不同类型的异构多智能体游戏环境, 通过构建专家规则库, 并采用强化学习方法选取规则子集, 从而动态调整模型的行为策略. Majchrzak等[21]则在此基础上, 设计更加高效的专家规则库, 统计多步决策序列产生的累积收益, 降低价值评估偏差, 然后通过强

化学习方法最大化累积收益期望, 动态更新集合权重以提升方法自适应性.

启发式规则型方法解决实时格斗游戏问题的方式类似于专家系统方法, 通过有限状态机和行为决策树等方式将环境信息提取后, 根据状态类别进行决策. 但是, 无论是有限状态机还是行为决策树, 本质上都是根据预先定义的规则进行决策, 需要设计者具有专家级别的游戏理解能力, 并且只能针对特定人物角色进行设计. 因此, 这类启发式规则型方法不具备良好的环境属性自适应性, 并且通常需要根据游戏参数的变化进行人为设计及调整. 综上所述, 通常启发式规则型方法不具备灵活的环境自适应性和模型自优化性. 当智能体所处环境无法被相应规则准确描述时, 有时会表现出不合理的应对行为, 并且容易被自适应对手模型获悉从而进行有效策略针对.

3.2 统计前向规划型

以蒙特卡罗树搜索 (Monte-Carlo tree search[22], MCTS)和滚动时域演化算法(rolling horizon evolution algorithm[23], RHEA)为代表的统计前向规划方法, 通过前向模型进行统计前向规划推理而找到最优解, 具备较强的模型泛化性和环境快速适应性[24].

统计前向规划型方法的核心思想是基于系统环境构建的前向模型, 进行高效准确地前向推理规划. 这类方法需要在前向模型进行大量的采样与迭代, 通过最优化启发式目标函数找到系统可行的近似最优解. 对于统计前向规划型方法, 前向模型的系统辨识度是影响这类方法推理准确性的重要因素. 面向即时制游戏需求时, 这类方法以牺牲一定的系统辨识度换取较快的前向推理效率, 并且保持较好的表现性能.

前向模型的定义: 一个确定性的状态转移函数f, 在已知的系统状态空间S和动作空间A下, 给定一个当前系统状态$s_t \in S$和智能体动作$a_t \in A$时, 通过前向模型f进行前向推理得到下一个系统状态$s_{t+1} \in S$, 记为

$$s_{t+1} = f(s_t, a_t). \tag{1}$$

式(1)给出的前向模型定义与马尔科夫决策过程的状态转移的主要区别是前向模型通常是确定性状态转移, 并且不产生相应的奖赏或回报信号. 通常, 前向模型来源于系统仿真环境, 将仿真环境的推理过程经过抽象简化得到, 从而使相应仿真环境下的前向推理过程更加快速高效. 此外, 可引入数据驱动的学习型方法构建快速且高效的前向模型, 通过高效拟合环境历史数据分布而得到对应前向模型. 简而言之, 前向模型是统计前向规划方法的推理基础, 构建前向模型的关键是有效平衡模型的系统辨识度和推理速率的矛盾关系.

通常, 启发式规则型方法需依赖专家规则系统的设计质量. 然而, 当游戏任务的状态及策略表征空间复杂度较高时, 通过手工编码方式设计高质量的专家系统较为困难, 需要消耗较高的人力资源和对游戏设计较为深刻的理解. 与之相比, 统计前向规划型方法基于前向模型, 在环境状态空间中进行高效探索和采样, 利用采集到的数据信息进行有效地前向推理与规划, 并且根据推理规划结果制定适宜的决策行为.

与基于神经网络模型训练优化的对抗博弈类算法相比, 统计前向规划型方法无需针对新的系统环境或角色属性重新训练, 可采用相同一套算法框架和模型参数来适应多种环境[25]. 统计前向规划型方法具有较强的系统可解释性, 通过启发式函数评估决策序列对当前状态影响所产生的未来价值, 根据专家知识或经验合理解释当前决策序列的价值性, 有利于提升模型推理和分析的效率. 此外, 统计前向规划型方法的模型性能具备可调节性, 可通过调节前向推理长度和优化迭代次数改变系统评估水平[26–27], 有效丰富策略表现的多样性.

3.2.1 蒙特卡罗树搜索算法

蒙特卡罗树搜索算法是一种在决策行为空间中进行蒙特卡罗随机采样, 并根据采样结构构建搜索树模型, 从而在指定空间内找到最佳决策行为的方法. 该方法可以有效平衡探索与利用之间的关系. 可通过预先指定迭代次数或者运行时间等多个条件约束来限制搜索时长, 在模型搜索结束后指向评价最优或访问次数最高的根节点的对应动作. 蒙特卡罗树搜索的核心是博弈双方均采用随机决策的方式进行对抗[28–29].

以实时格斗游戏作为研究背景, 相应MCTS方法的迭代优化过程如图3所示. 与面向回合制的MCTS方法相比, 格斗游戏的MCTS方法的主要区别在于需要同时考虑对抗双方的决策行为进行采样与利用.

尽管如此, 相应的迭代更新过程与回合制游戏类似, 需要经历以下4个阶段, 分别是选择阶段、扩展阶段、仿真阶段和反向传播.

步骤 1 选择阶段: 根据预先定义的树策略(tree policy), 选择并指向下一个节点作为待扩展节点, 利用上限置信区间(upper confidence bound, UCB)方法作为基本的树策略, 策略选择公式可定义为

$$\mathrm{UCB}_{\mathrm{val}} = \bar{r}_i + c\sqrt{\frac{2\ln N_i^p}{N_i}}, \tag{2}$$

其中: c为平衡系数, N_i表示节点i的访问次数, N_i^p表示节点i的父节点访问次数, $\bar{r}_i$表示节点i的平均奖赏, 其可定义为

$$\bar{r}_i = \frac{1}{N_i}\sum_{j=1}^{N_i}\mathrm{val}_j, \tag{3}$$

其中val_j表示第j次仿真的获得奖赏, 在格斗游戏中可用双方血量差表示, 具体为

$$\mathrm{val}_j = (\overline{hp}_m - hp_m) - (\overline{hp}_o - hp_o), \quad (4)$$

其中hp_m与$\overline{hp}_m$表示我方的当前血量和前向推理后血量, hp_o与$\overline{hp}_o$表示对方的当前血量和前向推理后血量, 等号右边两个括号反映对抗双方的血量变化.

步骤 2　扩展阶段: 若当前选择节点所处搜索深度低于阈值$D_{\max}$且非终止状态时, 对该选择节点添加一个(或多个)子节点进行扩展.

步骤 3　仿真阶段: 博弈双方利用启发式规则降低策略空间探索难度. 然后通过随机策略的方式, 在现有前向推理路径的基础上引入对手随机动作进行博弈推演, 直至达到仿真时限或最大搜索深度, 从而得到新的仿真对抗结果.

步骤 4　反向传播: 仿真阶段结束后, 从当前扩展的叶节点处反向传播更新其所对应的各层级父节点表示的UCB值, 直至反传到根节点处. 随后返回到步骤1, 进入到选择阶段继续进行更新采样, 直至满足系统预先定义的迭代要求.

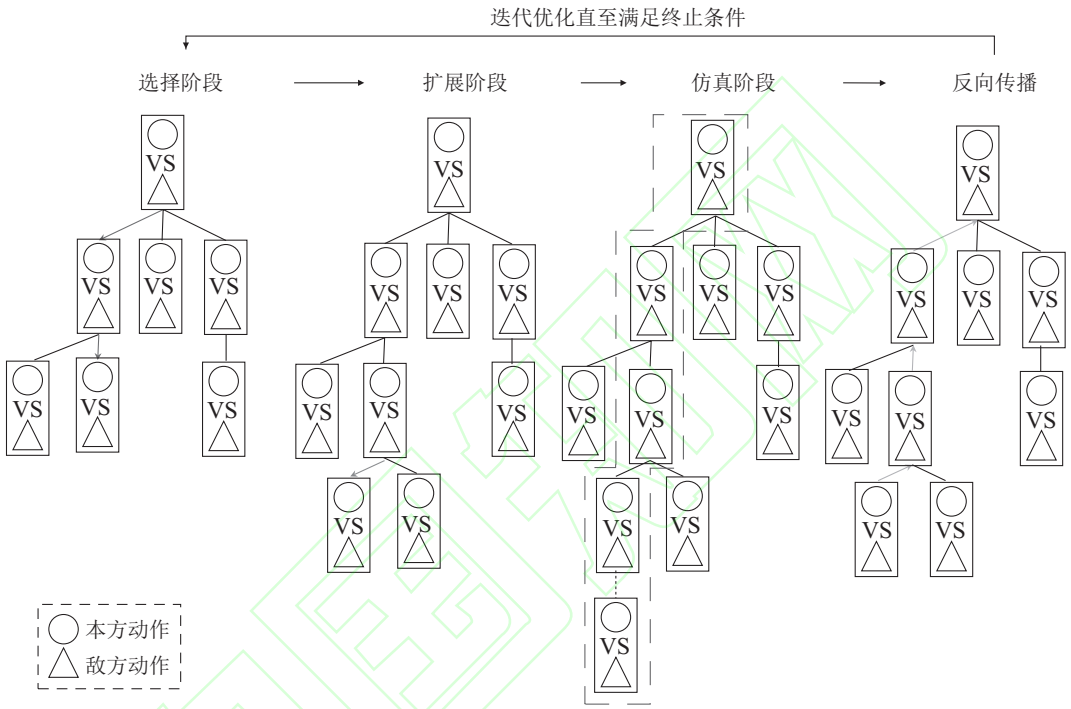


图 3　蒙特卡罗树搜索在格斗游戏的应用

Fig. 3　Monte-Carlo tree search for fighting game

3.2.2　滚动时域演化算法

滚动时域演化算法[30–31]是另一种常见的统计前向规划方法, 利用前向模型进行遗传演化过程. 通常将种群中的所有个体定义为动作序列, 基于前向模型采用前向推理规划的方式评估种群中的所有个体价值. 通常以当前状态作为决策目标起始点, 采取遗传演化的方式改变基因序列表示, 并根据所有个体的基因(动作序列)先后顺序进行执行, 直至达到环境终止状态或者执行完基因序列上的所有动作. 然后通过预先定义的启发式适应度函数来评估每个个体所能产生的价值, 最终选择评估价值最高个体的第1项动作作为决策动作, 并且在环境中执行该决策动作.

滚动时域演化算法的优化流程如图4所示, 种群中的不同个体由随机初始化得到, 个体基因表示相应动作. 根据适应度函数值的高低选择一定数量的个体, 然后将这些被选择的个体进行交叉变异, 以生成新的子代个体, 接着将新子代个体输入到前向模型进行前向推演, 得到推演后的环境状态, 通过适应度函数评估此刻推演状态, 得到新子代的适应值. 最终重复上述迭代优化, 直至达到系统优化时间上限或者种群个体表现处于完全收敛阶段.

将滚动时域演化算法应用到实时格斗游戏问题, 适应度函数定义为

$$f_{\mathrm{fit}}(s_t, \overrightarrow{z}^l, \overrightarrow{o}^l) = (1-\lambda) f_{\mathrm{sco}}(f_{\mathrm{FM}}(s_t, \overrightarrow{z}^l, \overrightarrow{o}^l)) + \lambda f_{\mathrm{div}}(\overrightarrow{z}^l), \quad (5)$$

其中:

$$f_{\mathrm{sco}}(s) = \begin{cases} -1, & \text{如果状态}s\text{为失利}, \\ 1, & \text{如果状态}s\text{为胜利}, \\ \alpha(hp_m(s) - hp_o(s)), & \text{其他情况}, \end{cases} \quad (6)$$

$$f_{\mathrm{div}}(\overrightarrow{z}^l) = 1 - \frac{1}{nl}\sum_{j=1}^{l} f_{\mathrm{cnt}}(\overrightarrow{z}^l(j)), \quad (7)$$

$$f_{\rm FM}(s_t, \overrightarrow{z}^l, \overrightarrow{o}^l) = s_{t+l}. \tag{8}$$

其中: $\lambda \in (0,1)$表示个体评分和多样性分数的均衡权重, $\overrightarrow{z}^l$与$\overrightarrow{o}^l$分别表示我方和对方智能体的序列长度为l 的动作序列, s_t为t时刻状态. $f_{\rm fit}$为适应度函数, 由评分函数$f_{\rm sco}$与多样性函数$f_{\rm div}$加权求和得到. 值得一提, 评分函数$f_{\rm sco}$用于评估对应状态的价值, α为归一化因子表示最大血量值的倒数. 多样性函数$f_{\rm div}$用于抑制种群个体同质化. n表示种群个体数量, $\overrightarrow{z}^l(j)$表示动作序列的第j个动作, 计数函数$f_{\rm cnt}$用于统计个体中的每个基因在当前种群中的出现次数. $f_{\rm FM}$表示格斗游戏的前向推理模型.

初代种群由随机初始化得到, 种群中所有个体的动作序列长度保持一致. 从n个体的种群中, 取前k个评分最高的个体作为精英保留到下一子代, $n-k$个剩余个体和精英个体一同进行交叉变异得到新一轮子代. 接着将新一轮子代通过前向模型推理, 再用适应度函数进行评估, 从而更新当前个体的评分. 经过上述步骤的反复迭代优化, 最终执行个体评分最高的动作序列对应的第一个动作[32–33].

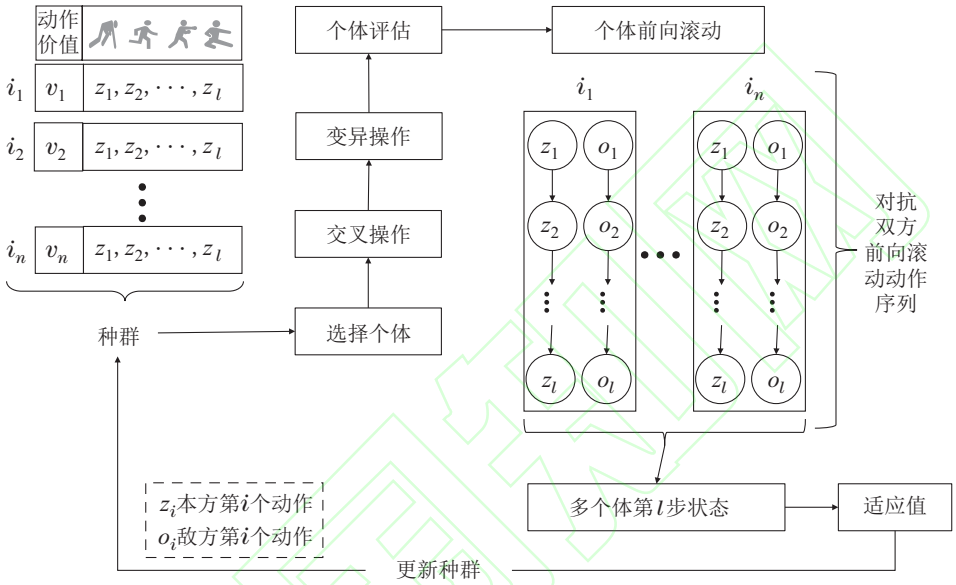


图 4 滚动时域演化算法在格斗游戏的应用

Fig. 4 Rolling horizon evolution algorithm for fighting game

不同于启发式规则型方法, 统计前向规划型方法(如蒙特卡罗树搜索MCTS和滚动时域演化算法RHE)具备良好的环境自适应能力. 该类型方法通过前向模型以构建状态动作转移推理规划器, 采用蒙特卡罗树搜索或者演化计算的统计优化方法, 经过前向模型推理规划, 预估策略推演收益来做出有效的决策行为. 此类方法无需进行大量训练优化便能自适应不同角色属性, 但是需要依靠大量采样以探索环境状态空间, 才能进行准确的收益评估, 对硬件系统要求较高.

3.3 深度强化学习型

近些年, 深度强化学习方法无论在回合制游戏还是即时制游戏均取得显著成果, 成为众多游戏AI的最高水平代表[34–37]. 深度强化学习将深度神经网络和强化学习的各自优势有效结合, 用于解决智能体在高维状态空间下的端到端序列决策优化问题. 深度强化学习方法主要由深度学习和强化学习两部分构成, 深度学习起源于人工神经网络, 采用多层神经网络融合, 通过梯度反向传播技术优化神经网络, 伴随着硬件计算资源的性能提升与高效深度模型优化算法的涌现, 深度学习强大的状态表征能力和泛化性能得到充分体现, 已经在图像识别、目标检测、语音识别、自然语言处理等领域取得了一系列重大突破. 强化学习通过与环境进行试错性交互, 有效平衡模型未知环境下的探索与利用间的关系, 通过最大化累积采样得到的奖赏信号来学习最优策略, 适用于序贯决策博弈.

深度强化学习的数据3元组为状态、动作及奖赏信号. 游戏引擎提供模型的输入状态信息, 可以是一维物理数值信息, 也可以是二维游戏画面信息. 格斗游戏状态包括: 角色属性、技能属性、距离属性与时间属性等. 其中角色属性为血量、能量、位置、速度、动作、角色状态(如站、蹲、倒、空)和剩余动作帧数; 技能属性为消耗能量、技能伤害量以及技能攻击属性(如近攻或远程); 距离属性为双方的相对物理距离以及位置关系; 时间属性为游戏剩余时间或者帧数. 动作空间采用离散化形式, 以可执行动作表示为候选动作集. 奖赏信号起到引导作用, 将最终胜负信号、双方血量差及步数惩罚项等作为奖赏导向, 促使智能体模型掌握格斗策略. 深度强化学习的具体推理及优化过程如图5所示. 因而, 深度强化学习方法可以直接适配于格

斗游戏任务求解过程.

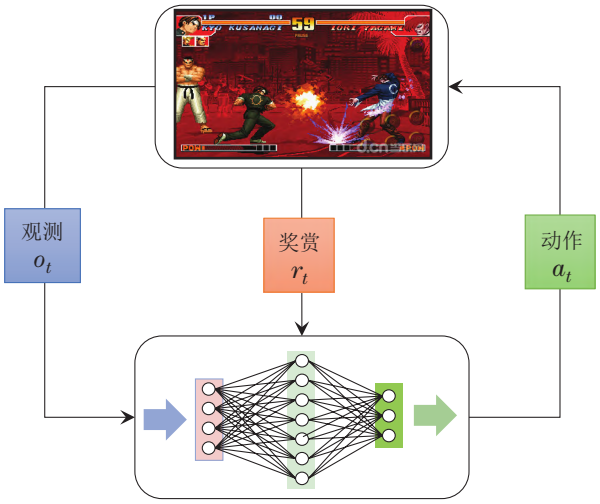


图 5　深度强化学习在格斗游戏的应用

Fig. 5　Deep reinforcement learning for fighting game

常用的深度强化学习方法可分为两类, 一类是以DQN[38]框架为代表的值优化型强化学习; 另一类是以Actor-Critic[39]架构为代表的策略梯度型强化学习. 由这两类方法衍生出的多种改进型算法已成功应用到实时格斗游戏, Tri等[40]将已有的格斗AI间的对抗数据作为训练样本[41], 设计卷积神经网络模型用于对应的状态行为预测, 通过监督学习或强化学习的方式优化目标预测模型, 以此构建端到端的博弈优化方法. Yoon等[42]以96 × 64的图像数据作为深度神经网络模型输入, 并且高效压缩动作输出空间. 通过与脚本AI对抗的方式进行环境交互, 利用DQN方法优化决策模型, 使策略模型具备环境适应与学习优化能力. 然而由于受制于对手的策略水平, 致使决策模型的水平提升有限. Takano等[43] 根据双方血量的下降过程, 设计对应的进攻和防守两类奖赏, 然后将两类奖赏通过加权和的方式来更新决策神经网络模型, 并且在2017年格斗游戏人工智能竞赛(fighting game AI competition, FTGAIC)中首次战胜MCTS型格斗游戏智能体. Kim等[44]设计两阶段的优化方式, 结合近似策略优化[45] (proximal policy optimization, PPO)算法优化智能体策略模型. 首先在第1 阶段将深度神经网络模型与MCTS型模型进行博弈对抗, 在交互过程中最大化累积奖赏信号以使神经网络模型掌握基本的格斗技能. 然后在第2阶段使神经网络智能体间相互自我对弈[46]以继续提升模型性能, 同时保留MCTS型智能体作为基本陪练以避免模型发生策略崩塌, 并且在胜率表现上显著优于2017年与2019年FTGAIC的大部分格斗游戏AI. Zhu等[47]将博弈论、动态规划与深度强化学习相结合, 通过在线学习的方式来求解两人零和马尔科夫博弈问题的纳什均衡策略, 并成功应用到格斗游戏中掌握具有竞争力的策略.

与启发式规则型方法和统计前向规划型方法不同, 深度强化学习方法的系统输入直接来源于环境模型提供的高维数据信息, 通过深度神经网络模型进行状态特征提取, 再将提取到的特征信息传至决策层进行前向推理. 深度强化学习的方法特点在于通过最大化累积奖赏对模型决策系统进行优化, 然后通过环境交互使决策模型动态自适应优化. 然而, 深度强化学习方法本身存在样本利用率及采样效率低下、硬件计算资源需求高、训练过程不稳定、以及学习过程中的策略遗忘或崩塌等问题. 此外, 深度强化学习方法身为环境交互型方法, 容易受到环境系统模型的制约, 例如考虑两人零和博弈问题时, 深度强化学习智能体容易受到对手策略水平的限制, 当对手策略水平不高或相对单一时, 容易造成模型的过拟合, 因而削弱智能体的对抗泛化性能, 无法使智能体得到持续性提升.

综上所述, 纵观当前格斗游戏的研究现状, 单一的计算智能方法具有自己独特的优势属性, 但是也存在相应不足. 如何将目前已有的智能方法进行有机融合, 从而形成优势互补, 这是当前格斗游戏AI的一个主要研究方向.

4　格斗游戏竞赛和方法评价

格斗游戏人工智能竞赛是由日本立命馆大学智能电脑娱乐实验室 (intelligent computer entertainment lab, ICE实验室)自2013年发起, 在每年的游戏智能领域权威会议IEEE Conference on Games上举办, 该竞赛旨在探究在有限家用计算资源平台上, 通用型格斗游戏AI的研究思路及具体实现方法.

4.1　格斗游戏人工智能平台

为了鼓励更多科研工作者投入到格斗游戏人工智能领域的研究, 设计并且实现满足即时制需求的人工智能对抗决策方法, 以便不同算法或模型之间相互比较. ICE实验室于2013年发布格斗游戏人工智能平台[12], 提供统一的算法测试环境, 找到适应性强且计算资源需求较低的具备通用属性的格斗游戏AI设计方案. 该平台早期只支持一个原生格斗模型KFM-prototype, 现今已发展成3种各具特色的格斗人物, 分别为ZEN, LUD和GARNET. 格斗人物具体介绍如下:

ZEN角色: 格斗技巧综合全面, 同时具备近战和远攻的两重攻击类型, 攻击速度介于GARNET和LUD之间;

LUD角色: 所有攻击动作皆会消耗自身属性能量, 需要通过有效击打或被动击打积蓄能量, 攻击速度较为迟缓, 但是攻击动作杀伤性较强;

GARNET角色: 擅长空中格斗搏击, 连招组合技能较多, 攻击速度较快, 但是攻击杀伤性较弱;

综上所述, 每种格斗角色属性不尽相同, 有效的进攻策略也不一致. 在格斗游戏过程中, 智能体需要结

合角色属性与动作特点, 根据场上局势和对手行为制定有效格斗策略, 以期在最短时间内击败对手赢取胜利.

4.2 竞赛规则

格斗游戏竞赛的比赛时长为60 s, 每场比赛须打满3局, 每局开始前提供5 s准备时间. 即时制要求为60帧每秒(即16. 67 ms决策时间), 并且为模拟人类玩家的正常反应时间, 模型输入信息具有15帧固定系统延迟构成非完美信息博弈环境. 每局结束后系统将重置对抗双方的原始信息和起始位置. 取计分高者为胜, 具体计分公式Score为

$$\text{Score} = \frac{\text{loss\_}hp_o}{\text{loss\_}hp_m + \text{loss\_}hp_o} \times 100, \qquad (9)$$

其中: $\text{loss_}hp_o$与$\text{loss_}hp_m$分别表示敌方失去的血量与我方失去的血量.

随着格斗游戏的比赛形式愈加丰富, 从2017年开始, 比赛赛道增加为两大基本类型. 一类为标准赛道, 参赛AI间进行相互对抗, 取胜场数越多则相应排名越高, 目标是测试AI取胜场数. 另一类为快速赛道, 参赛AI与组办方提供的AI模型进行对抗, 在同为胜场结果的前提下, 消耗时间越短则相应排名越高, 目标是考察AI取胜效率. 比赛环境提供了3种不同属性角色并且由于赛道模式分为2类, 因此总计有6项赛道. 由于赛道数量的增加及相应要求有所区别, 计分方式变更为Formula-1 (F1)计分标准, 每项赛道前10名具有相应积分, 具体计分标准如表1所示.

表 1 F1计分系统

Table 1 Formula-1 scoring system

名次	积分	名次	积分	名次	积分
1	25	5	10	9	2
2	18	6	8	10	1
3	15	7	6		
4	12	8	4		

4.3 格斗游戏人工智能竞赛历年冠军方案介绍

为进一步分析各类格斗游戏AI的方法特性, 如表2所示, 给出了从2013年到2020年的历年冠军格斗游戏AI的特性. 历年的FTGAIC冠军AI技术细节归纳如下所述, 括号中的内容表示年份.

表 2 FTGAIC历年冠军AI特性

Table 2 Champion AI characteristics in FTGAIC over the years

年份+AI名称 (算法)	快速反应性	同步性	动作持续性	多角色自适应性	对手模型分析	可学习性
2013-T (有限状态机)	√	×	×	×	×	×
2014-CodeMonkey (动态脚本)	√	×	×	×	×	×
2015-Machete (规则集合)	√	×	×	×	×	×
2016-Thunder01 (脚本策略+MCTS)	√	√	√	√	×	×
2017-GigaThunder (规则空间约束MCTS)	√	√	√	√	×	×
2018-Thunder (启发式规则+MCTS)	√	√	√	√	×	×
2019-ReiwaThunder (MiniMax+启发式规则+MCTS)	√	√	√	√	×	×
2020-ERHEA_PI (RHEA+自适应对手建模)	√	√	√	√	√	√

T (2013): 基于有限状态机[48]方法, 根据对抗双方的相对距离和对手历史动作信息, 在候选动作集中选择最佳反击招式.

CodeMonkey (2014): 基于动态脚本法, 根据人类专家经验, 预先定义离线规则库, 设计自适应权重动态选择执行脚本. 在博弈对抗过程中每次间隔3 s时间, 根据博弈结果反馈回的奖赏信号, 动态更新脚本选择权重, 以此优化模型博弈水平.

Machete (2015): 基于规则集合法, 特点与T (2013)类似, 但是对能量值因素更加敏感, 风格上更考虑有效规避对手的攻击行为并且制造反击机会.

Thunder01 (2016): 基于MCTS并融合脚本策略方法, 引入2015年Machete的脚本规则作为指导, 并且结合MCTS方法自适应不同属性角色, 增强模型的泛化

性.

GigaThunder (2017): 在2016年格斗游戏AI冠军Thunder01的基础上, 根据不同角色属性和赛道要求来设计独立的启发式规则, 并且将启发式规则作为约束引入到决策模型, 降低动作搜索空间并且减小能量消耗, 加强有效动作被选择概率.

Thunder (2018): 在2017年格斗游戏AI冠军GigaThunder的基础上, 引入高伤害技能施放条件, 并且增加防御机制以避免陷入到对手的逼墙角策略.

ReiwaThunder (2019): 在2018年格斗游戏AI冠军Thunder的基础上, 进一步优化前向模型的系统辨识度, 并且引入极小化极大值法加强博弈双方行为评估准确性.

ERHEA_PI (2020): 在2019年格斗游戏AI亚军RHEA_PI的基础上, 基于优化后的前向模型, 将滚动时域演化算法与策略梯度式自适应对手建模高效结合, 参考Thunder的动作规则集约束式来简化搜索区域, 加快模型的前行推理与迭代更新效率.

随着深度强化学习的研究与应用在越来越多不同类型游戏上取得的突出表现, 2018年起, 格斗游戏人工竞赛开始出现了基于深度强化学习的博弈策略模型, 并且在部分赛道上的表现性能优于统计前向规划型方法, 然而在整体表现仍然略逊一筹. 在2019年, 以RHEA型为代表的另一类统计前向规划型方法开始出现, 虽然其在当年的比赛上的表现仍略逊于MCTS型模型, 但是整体性能已十分接近. 到2020年, RHEA型算法实现性能上的超越, 打破了MCTS型算法长达4年的统治地位.

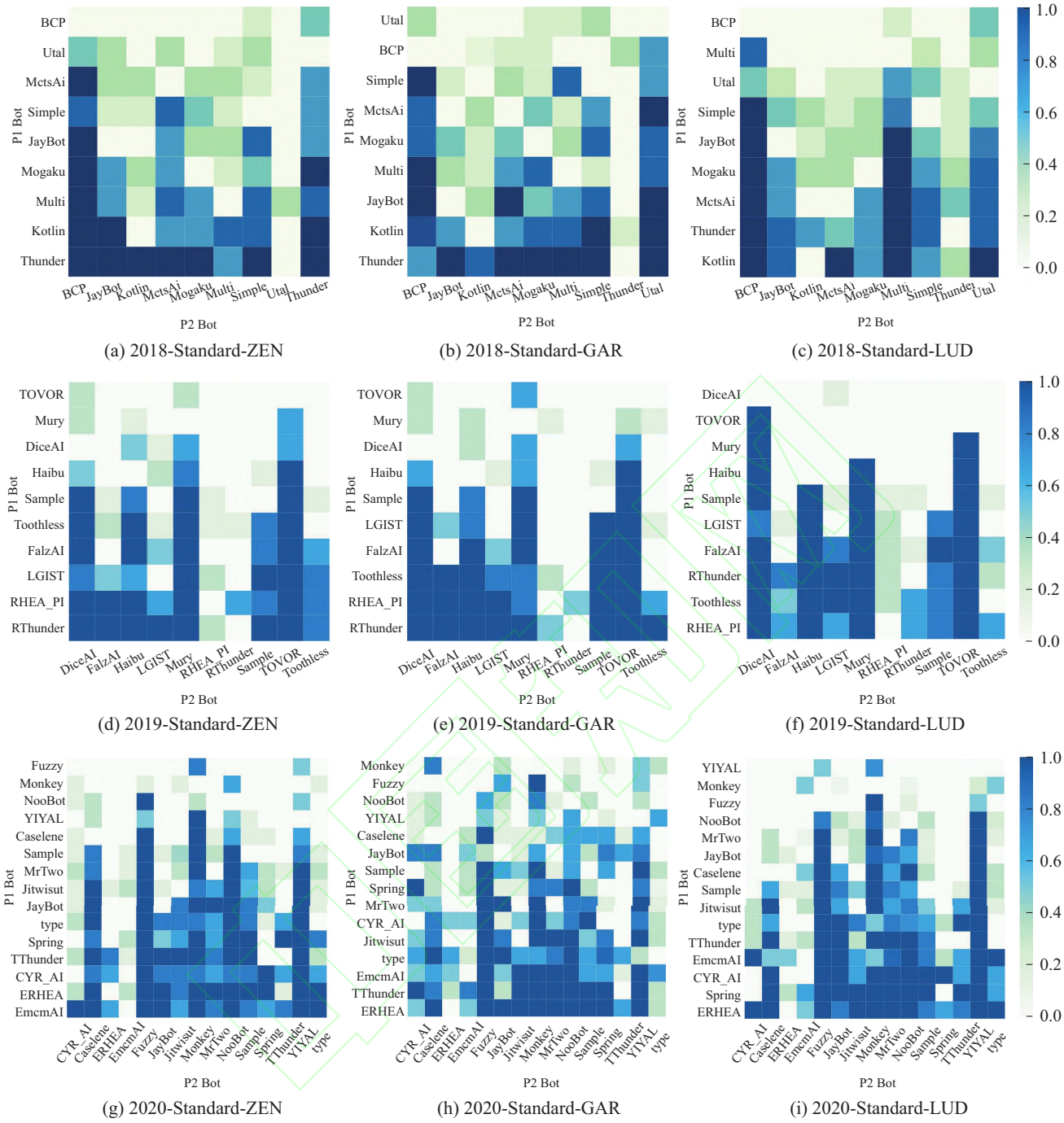
4.4　2018年到2020年F1-积分分析

根据格斗游戏主办方的比赛积分统计方式对2018年到2020年的格斗游戏AI进行分析. 格斗游戏人工智能竞赛采用F1-积分标准对不同赛道的表现进行统计, 将各项排名所获的积分总和作为最终排名, 排名计算规则已在第3.2节描述, 对应积分规则如表1所示, 累积积分越高的AI排名越靠前. F1-积分表的统计方式分为标准赛道和快速赛道两类. 标准赛道重点考察在不同对手对抗环境下的格斗游戏AI的最终胜率表现, 而快速赛道则是重点考察对应同一个对手的格斗游戏AI取胜效率. 表3所示为从2018年到2020年FTGAIC所有赛道的前6名格斗游戏AI积分表.

表 3　2018–2020年FTGAIC所有赛道前6名积分表

Table 3　Scores of top-6 bots for all tracks in FTGAIC from 2018 to 2020

2018-标准赛道积分					2018-快速赛道积分					2018-最终积分排名	
AIs	ZEN	GARNET	LUD	sum	AIs	ZEN	GARNET	LUD	sum	AIs	sum
Thunder	**25**	**25**	**18**	68	KotlinTestAgent	**18**	**18**	**25**	61	Thunder	1st–128
KotlinTestAgent	**18**	**18**	**25**	61	Thunder	**25**	**25**	10	60	KotlinTestAgent	2nd–122
MogakuMono	12	12	12	36	JayBot_GM	**15**	12	12	39	MogakuMono	3rd–74
JayBot_GM	10	**15**	10	35	MogakuMono	12	6	**18**	36	JayBot_GM	4th–72
MultiHeadAI	**15**	12	4	31	SampleMctsAi	6	10	**15**	31	MultiHeadAI	5th–60
SampleMctsAi	6	8	**15**	29	MultiHeadAI	10	**15**	4	29	SampleMctsAi	5th–60
2019-标准赛道积分					**2019-快速赛道积分**					**2019-最终积分排名**	
AIs	ZEN	GARNET	LUD	sum	AIs	ZEN	GARNET	LUD	sum	AIs	sum
ReiwaThunder	**25**	**25**	**18**	68	ReiwaThunder	**25**	**25**	**15**	65	ReiwaThunder	1st–133
RHEA_PI	**18**	**18**	**25**	61	RHEA_PI	**18**	**18**	**25**	61	RHEA_PI	2nd–122
Toothless	10	**15**	**18**	43	Toothless	**15**	**15**	**18**	48	Toothless	3rd–91
LGIST_Bot	**15**	12	10	37	FalzAI	12	10	10	32	FalzAI	4th–68
FalzAI	12	12	12	36	LGIST_Bot	10	12	8	30	LGIST_Bot	5th–67
SampleMctsAi	8	8	8	24	SampleMctsAi	8	8	12	28	SampleMctsAi	6th–52
2020-标准赛道积分					**2020-快速赛道积分**					**2020-最终积分排名**	
AIs	ZEN	GARNET	LUD	sum	AIs	ZEN	GARNET	LUD	sum	AIs	sum
ERHEA_PI	**18**	**25**	**25**	68	ERHEA_PI	**25**	**25**	10	60	ERHEA_PI	1st–128
EmcmAI	**25**	**15**	12	52	TeraThunder	**18**	**18**	12	48	TeraThunder	2nd–88
TeraThunder	12	**18**	10	40	type	**15**	12	**18**	45	type	3rd–73
CYR_AI	**15**	8	**15**	38	CYR_AI	4	4	**25**	33	EmcmAI	4th–72
SpringAI	10	4	**18**	32	SpringAI	12	0	**15**	27	CYR_AI	5th–71
type	8	12	8	28	MrTwo	2	**15**	4	21	SpringAI	6th–59



(a) 2018-Standard-ZEN (b) 2018-Standard-GAR (c) 2018-Standard-LUD

(d) 2019-Standard-ZEN (e) 2019-Standard-GAR (f) 2019-Standard-LUD

(g) 2020-Standard-ZEN (h) 2020-Standard-GAR (i) 2020-Standard-LUD

图 6 2018–2020 FTGAIC标准赛道胜率热力图

Fig. 6 Win rate heatmap of standard tracks in FTGAIC from 2018 to 2020

标准赛道部分, 格斗双方智能体在同等初始条件下进行1vs1实时对抗. 不同的角色属性可分为ZEN, GARNET 和 LUD的3种不同的角色类型. 2018年到2020年格斗游戏AI相互间的对抗胜率热力图如图6所示, 纵轴坐标代表本方智能体, 横轴坐标代表对手智能体. 青色热力色度表示本方智能体对抗对手智能体的交战胜率, 颜色深度越深代表胜率越高. 此外, 自下而上的纵轴顺序表示对应智能体由高到低的胜率排名. 2018年到2020年的FTGAIC所有赛道的积分表如表3所示, 直接反映了所有格斗游戏AI在不同赛道的表现结果.

2018年, Thunder取得了总积分第1的成绩, 在其中5个子项赛道上取得第2名的成绩, 但是在快速赛道的LUD角色上表现不佳. 快速赛道的LUD角色为官方MCTS型模型AI, 并且其动作属性值在比赛中会进行调整, 为的是考察策略模型的环境泛化性和适应性. KotlinTestAgent位列第2, 在快速赛道的LUD角色取得最佳表现, 反映了该模型具有良好的模型泛化性和环境适应性. 分析Thunder与KotlinTestAgent的异同点发现, Thunder与KotlinTestAgent都采用MCTS与启

发式规则集合相结合的方法, 通过动作连续性、候选动作集和双方相对距离等因素构成启发式规则型策略集合, 然后将策略集合与MCTS型算法进行结合, 由策略集合负责处理极端情况, MCTS型算法负责处理一般情况, 而不是完全依赖MCTS型算法解决模型泛化性问题. 但也正是启发式规则型策略集合设计的不同, 导致了最终的性能表现存在差异, 并且KotlinTestAgent基于Kotlin 编程语言设计, 使得模型前向推理速度和效率更高. 尽管如此, Thunder凭借更合理全面的策略规则集合作为指导, 在预先已知角色属性ZEN和GARNET的性能表现更佳. 剩余的其它AI的性能表现则有不小的差距. 胜率结果方面反映了MCTS型模型与启发式规则结合的方法具有高效的环境适应性和合理性. 通过MCTS模型可有效处理多数对抗环境场景, 利用启发式规则针对性解决特定场景. 综上而言, 统计前向规划型与启发式规则型方法的结合在2018年FTGAIC中成为格斗游戏模型的典型范式.

2019年, ReiwaThunder在Thunder的基础上得到进一步改进, 在标准赛道和快速赛道的总积分排名皆为第1. MCTS型模型与启发式规则结合的格斗游戏模型仍然处于领先地位. RHEA_PI作为首次参加的RHEA型格斗游戏AI, 它的积分排名紧跟ReiwaThunder, 从累计积分值上, 二者处于第1梯队, 并且要大幅领先其他参赛队伍. 以外, 在调整参数的角色LUD赛道上, RHEA_PI在标准赛道和快速赛道均取得了最佳成绩. 体现了RHEA与自适应对手建模算法结合所具有较好的模型泛化性. 尽管如此, ReiwaThunder模型凭借更有效的启发式规则作为指导, 在角色属性值确定的ZEN和GARNET上表现更为突出, 仍然保持了一定的领先地位.

2020年, 涌现出大量以PPO (proximal policy optimization)[45]和SAC (soft actor-critic)[49]为代表的深度强化学习方法. 不同以往, 该届FTGAIC中GARNET和LUD的角色属性会进行调整, 目的是更进一步考验格斗游戏模型的环境适应性. 根据表3所示, type, EMCMAI和CYR_AI等深度强化学习方法均受到不同程度的影响致使性能发挥受限, 而以RHEA和MCTS为代表的统计前向规划方法则表现相对较好. 尤其是将RHEA与自适应对手建模高效结合的ERHHEA_PI, 在总共六项子赛道的五项子赛道上位居第2, 并且在标准赛道和快速赛道的积分排名要大幅领先各赛道第2名, 因而总积分排名上优势显著.

4.5　性能因子构造与分析

格斗游戏人工智能竞赛的赛事主办方按照相互对抗的胜负关系和同一对手的对抗效率关系的两个维度对每个格斗游戏AI进行性能评估. 然而, 基于F1计分方式得到的最终积分结果无法完全分析格斗游戏AI的具体特性和对抗风格. 因此, 根据比赛结果, 本文从对抗胜负、血量关系、对抗时长、血量优势以及伤害关系等维度, 进一步具体分析格斗游戏AI算法的主要特点, 构造5个具体的性能因子, 可归结为从胜率、剩余血量、执行速率、优势性和伤害性的5个指标, 可有效准确评价格斗游戏AI算法的性能, 用于指导算法测试验证.

4.5.1　性能因子设计

下面具体介绍每个性能因子的具体含义与定义方式:

胜率(Win_rate): 反映格斗游戏AI的对抗取胜次数. 根据比赛结果统计我方智能体的获胜概率, 这里胜场数记为1, 平局数记为0.5, 负场数记为0, 具体计算为

$$\text{Win\_rate} = \frac{\text{win}_{\text{count}} + 0.5 \times \text{ti}_{\text{count}}}{\text{total}}, \tag{10}$$

其中: $\text{win}_{\text{count}}$表示我方智能体的胜场数, ti_{count}表示我方智能体的平局数, total表示我方智能体的对局总数.

剩余血量(Remain_hp): 反映格斗游戏AI的防御或躲避伤害的能力. 游戏结束时我方智能体的平均剩余血量$\overline{hp}_m$, 具体计算为

$$\text{Remain\_}hp = \frac{\overline{hp}_m}{hp_{\max}}, \tag{11}$$

其中$hp_{\max}$表示最大血量值.

执行速率(Speed): 反映格斗游戏AI的为取胜以结束对抗过程的效率. 游戏结束时我方对局取胜所剩余的比赛时长$\overline{\text{remain_time}}$. 若我方为落败时, 则游戏剩余时间取值为0, 具体计算为

$$\text{Speed} = \frac{\overline{\text{remain\_time}}}{\text{full\_time}}, \tag{12}$$

其中full_time为单局总比赛时长.

优势性(Advantage): 反映对抗双方的对抗性能差距. 游戏结束时, 以双方的剩余血量之差作为评判依据, 再进行归一化后的具体计算为

$$\text{Advantage} = \frac{1}{2}\left(\frac{\overline{hp}_m - \overline{hp}_o}{hp_{\max}} + 1\right), \tag{13}$$

其中$\overline{hp}_o$表示敌方智能体的平均剩余血量.

伤害性(Damage): 反映本方格斗游戏AI的伤害制造程度. 游戏结束时造成对方的血量伤害, 具体计算为

$$\text{Damage} = 1 - \frac{\overline{hp}_o}{hp_{\max}}. \tag{14}$$

当各项因子均取最大值1时, 综合战力理论上限约为2.38 (由5个腰长为1且顶角为72°的等腰三角形构成, 合为正五边形面积, 计算表示为$2.38 = 0.5 \times 5 \times \sin(72°)$). 然而, 由于五项指标中的Speed项较难为1 (表示游戏开始后立刻结束), 因而计算战力值必然会

低于理论上限. 相比于单一胜率表示, 五项相关的整体战力值因子分析无疑更具全面性.

4.5.2 基于性能因子的战力值计算

根据性能因子构造方式, 按照不同角色类型和赛道模式, 统计计算每个AI的性能因子后得到2018至2020年FTGAIC所有赛道前6名战力值积分表, 具体数据结果如表4所示, 通过图7的表示方式将具体数据可视化如图8和图9所示. 按照不同赛道与角色分别统计每个AI的各项性能因子, 经过累积和方式得到对应角色战力值积分, 并且通过平均化形式得到对应赛道下的平均积分, 然后根据平均积分由高到低排序并列出对应AI.

通过将表3和表4进行对比发现, F1统计积分表与战力值积分表的具体排名存在一定差异. 战力值积分涵盖了格斗游戏AI的攻击性、优势性、稳定性和效率性等多类考量, 相比于仅考虑胜率和进攻效率的F1积分形式无疑更加全面有效.

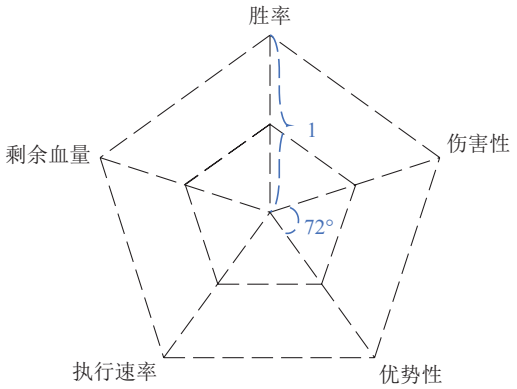


图 7 性能因子可视化

Fig. 7 Visualization of performance factor

4.5.3 讨论分析

2018年, 战力值统计结果排名与主办方提供的赛道F1积分排名基本一致. 在标准赛道上, Thunder是唯一战力值均值大于1的AI. KotlinTestAgent紧随其后. 在战力值总和排名上, Thunder与KotlinTestAgent分别占据标准赛道与快速赛道的第1名, 整体水平位于第1梯队.

表 4 2018–2020年FTGAIC所有赛道前6名战力值积分表

Table 4 Combat strength scores of top-6 bots for all tracks in FTGAIC from 2018 to 2020

2018-标准赛道战力					2018-快速赛道战力				
AIs	ZEN	GARNET	LUD	avg	AIs	ZEN	GARNET	LUD	avg
Thunder	**1.288**	**1.246**	**0.836**	1.123	KotlinTestAgent	**1.048**	**1.089**	**0.812**	0.983
KotlinTestAgent	**0.903**	**0.945**	**1.044**	0.964	Thunder	**1.245**	**1.249**	0.349	0.948
JayBot_GM	0.516	**0.711**	0.621	0.616	JayBot_GM	0.751	0.946	**0.432**	0.71
MogakuMono	0.544	0.454	0.729	0.576	MogakuMono	0.582	**1.001**	0.027	0.537
MctsAi	0.411	0.488	**0.793**	0.564	MctsAi	**0.856**	0.361	0.324	0.514
MultiHead	**0.656**	0.485	0.098	0.413	MultiHead	0.4	0.469	**0.406**	0.425
2019-标准赛道战力					2019-快速赛道战力				
AIs	ZEN	GARNET	LUD	avg	AIs	ZEN	GARNET	LUD	avg
RHEA_PI	**1.135**	**1.355**	**1.103**	1.198	ReiwaThunder	**1.458**	**1.522**	**0.861**	1.28
ReiwaThunder	**1.129**	**1.108**	0.791	1.009	RHEA_PI	**1.166**	**1.465**	**0.866**	1.166
Toothless	0.649	**1.111**	**1.063**	0.941	Toothless	**1.164**	**1.494**	**0.834**	1.164
FalzAI	**0.709**	0.56	**0.975**	0.748	FalzAI	0.836	1	0.583	0.806
LGIST_Bot	0.674	0.6	0.626	0.633	LGIST_Bot	0.542	1.028	0.443	0.671
SampleMctsAi	0.439	0.387	0.594	0.473	SampleMctsAi	0.454	0.551	0.543	0.516
2020-标准赛道战力					2020-快速赛道战力				
AIs	ZEN	GARNET	LUD	avg	AIs	ZEN	GARNET	LUD	avg
ERHEA_PI	**1.066**	**1.241**	**1.201**	1.169	ERHEA_PI	**1.502**	**1.52**	0.51	1.178
TeraThunder	**0.93**	**1.031**	0.726	0.896	TeraThunder	**1.353**	**1.213**	0.869	1.145
EmcmAi	**0.902**	0.677	0.85	0.81	type	**1.201**	0.712	**1.169**	1.027
CYR_AI	0.852	0.493	**0.955**	0.767	CYR_AI	0.941	0.586	**1.315**	0.947
Spring	0.842	0.427	**0.986**	0.752	Spring	1.165	0.333	1.13	0.876
type	0.688	**0.683**	0.658	0.676	EmcmAi	0.985	0.622	0.499	0.702

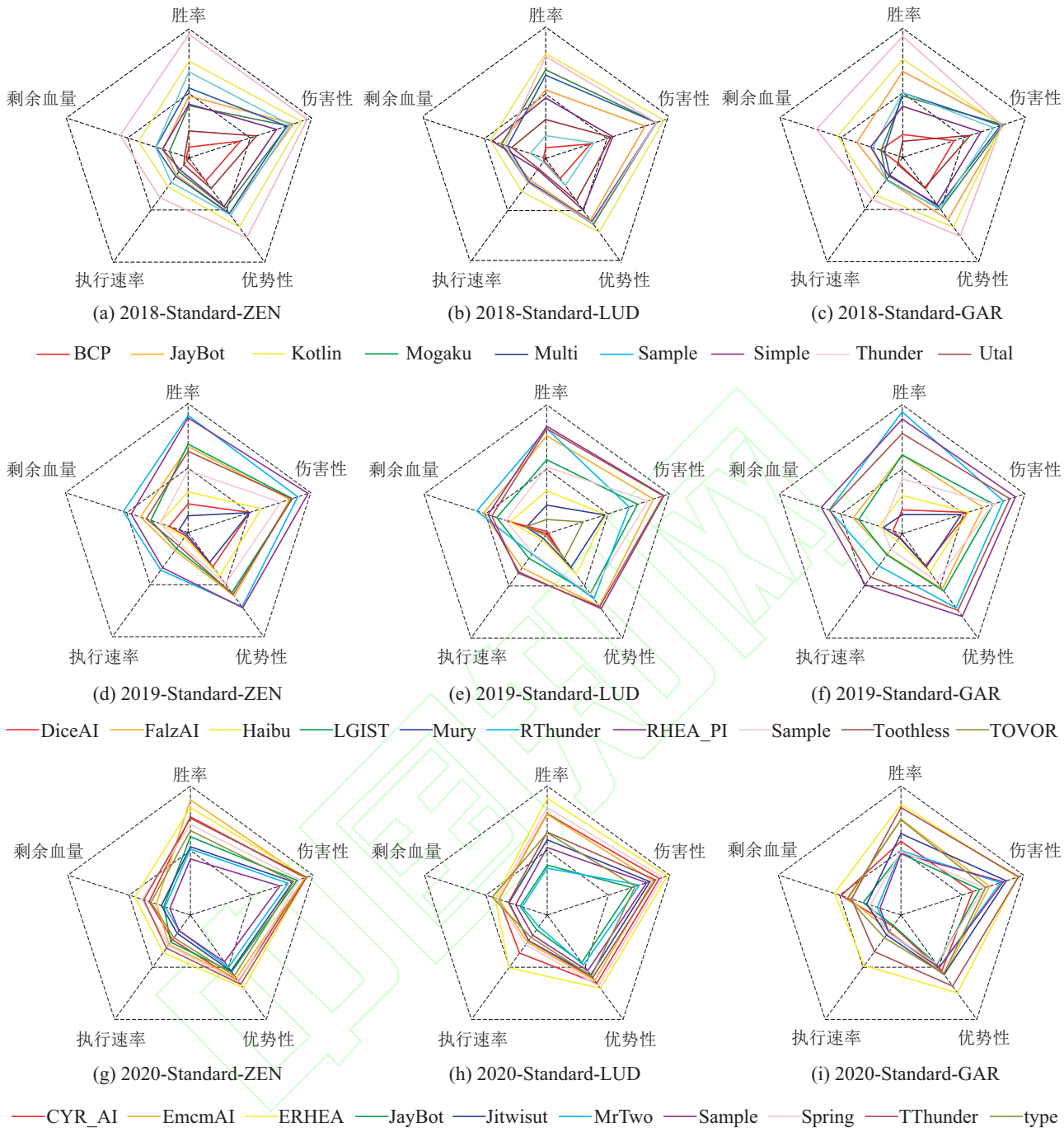


图 8　2018–2020 FTGAIC标准赛道性能因子可视化

Fig. 8　Visualization of performance factor of standard tracks in FTGAIC from 2018 to 2020

2019年, RHEA_PI与ReiwaThunder分列为标准赛道与快速赛道的战力值第1名. 与2018年的战力值结果相比, 2019年的前两名格斗游戏AI的战力值均值皆超过1. RHEA_PI在标准赛道的战力值排名要高于其对应的赛道积分排名, 原因是其在优势性和伤害性两项指标要显著优于其他类型AI, 直观反映该模型捕捉有效换血进攻时机的能力, 然而高回报也产生高风险, 其在剩余血量控制上则表现较差, 表现其较为激进的进攻策略风格. 因此, 由于RHEA_PI设计的启发式评估函数较为激进, 导致了其的整体性能与ReiwaThunder存在一定差距. 由此观之, 在2019年的FTGAIC上, 尽管MCTS型方法依然占据领先地位, 但是RHEA型方法与对手模型的结合表现出较强的模型泛化性和环境适应性, 具有性能上升潜力.

2020年, ERHEA_PI的战力值排名与F1统计积分排名一致, 均位列第1位. EHREA_PI除了在快速赛道的角色LUD表现较差外, 在其他不同角色子赛道均位于榜首, 因而在标准赛道和快速赛道的平均战力值排名均为领先地位, 尤其是在快速赛道下的ZEN和GARNET角色的战力值均已突破至1.5. 再次佐证了

RHEA与自适应对手模型相结合产生的极佳的系统泛化性和对抗适应性, 表现出突出的格斗游戏水平. 尽管如此, EHREA_PI在快速赛道下的LUD角色属性表现不佳, 反映了该模型仍存在一定不稳定性, 有待进一步改进和完善, 很可能与自适应对手模型的系统拟合度有关. 由于ZEN的角色属性值始终保持不变, 因此, 在标准赛道上, 基于PPO深度强化学习方法的EmcmAI表现性能最佳, 体现该算法在与多陪练对手对抗适应的前提下, 可使策略模型掌握ZEN角色下的适应性较强的策略. 然后, 基于统计前向规划方法的ERHEA_PI和TeraThunder紧随其后, 体现了较强的环境适应性. 其它的深度强化学习型模型由于训练过程不够充分, 导致实际的策略表现水平有限. 而在角色属性产生变化的GARNET和LUD角色, ERHEA_PI在标准赛道的性能表现占据领先地位, 整体表现全面优于其它模型, 体现了RHEA与自适应对手建模结合带来的环境适应性表现. EHREA_PI是在2019年RHEA_PI的基础之上, 通过进一步丰富对手模型观测状态以及优化对手模型初始化策略, 引入本方实际动作与对手历史动作的双奖赏机制进行优化, 使模型适应性得到进一步提升.

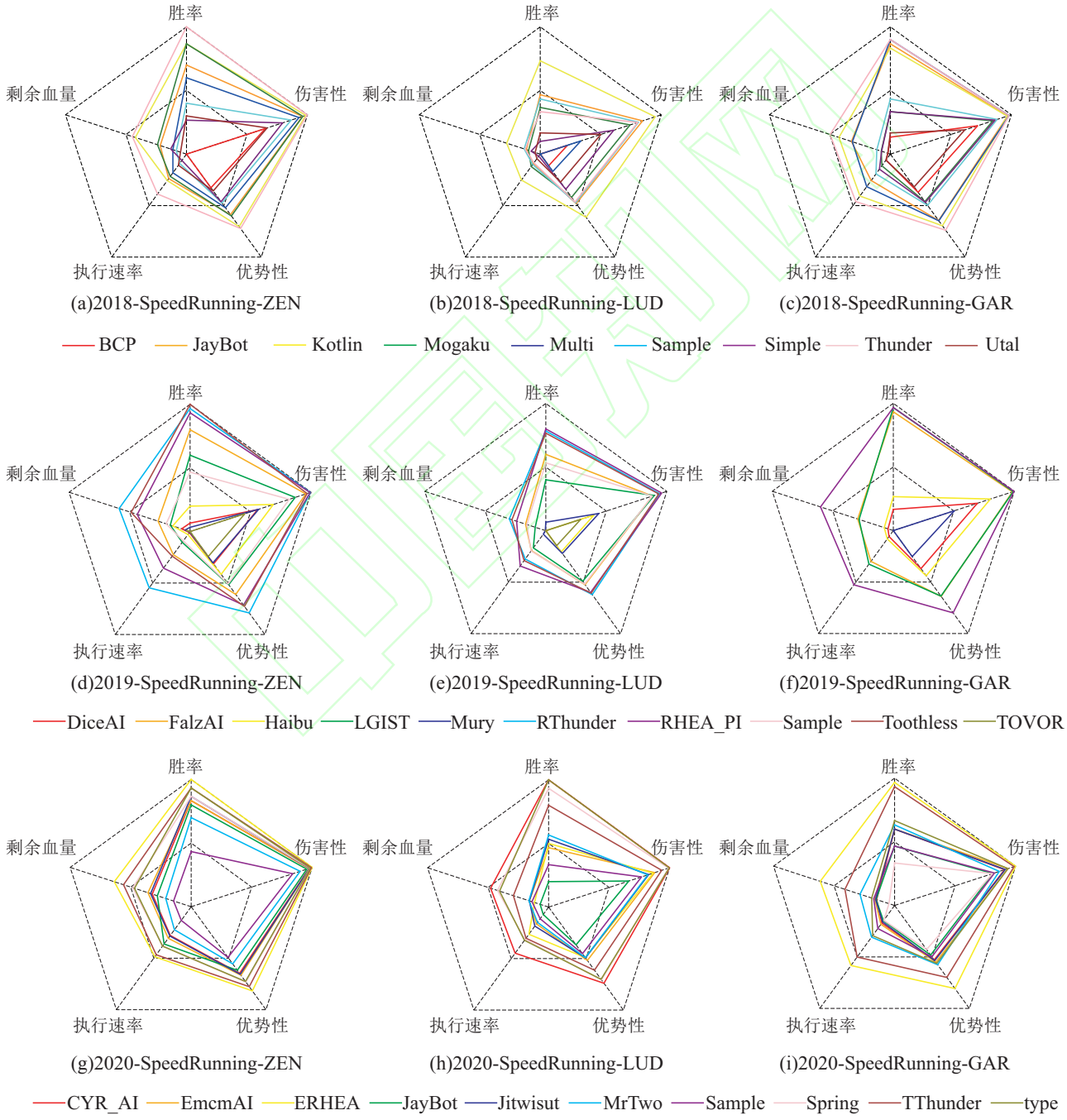


图 9 2018–2020 FTGAIC快速赛道性能因子可视化

Fig. 9 Visualization of performance factor of speedrunning tracks in FTGAIC from 2018 to 2020

结合F1积分统计结果与性能因子构造的战力值计算结果, 分析2018到2020连续三年的比赛结果, 前半段以Thunder为首的MCTS型算法占据统治地位. 随后RHEA型算法逐步展现出极佳的适应潜力, 并且在后来首次实现性能上的超越. 另外, 深度强化学习型方法也涌现出不错的战力与可优化潜力. 然而, 该类型方法未能取得第1名. 究其原因主要有3点: 一是角色多样性与动作属性参数变化, 使在原有训练环境得到的策略模型的泛化性无法得到有效保证; 二是受限于硬件计算资源的限制, 使基于对手池训练的格斗策略集合无法保证得到完全覆盖; 三是格斗游戏的状态空间与动作组合空间较大, 在训练对手样例不够充分或未知角色属性的情况下, 基于强化学习的策略模型的性能泛化性和环境适应性无法得到有效保证. 因此, 对应的强化学习模型泛化性能力不足和模型解空间有限的问题仍有待进一步研究. 值得注意的是, 虽然统计前向规划型方法长期占据了FTGAIC的主导地位, 但是仍然没有出现一个智能体在所有场景下均取得第1名的情况. 模型间策略表现存在相互制约的关系, 并且实际的战力值水平与理论上限值仍然存在一定差距. 策略模型的角色属性泛化性与对抗角色的适应性仍然有待进一步的提高.

5　格斗游戏方法的思考与展望

近年来, 随着深度强化学习与统计前向规划型方法的迅猛发展与应用, 使得游戏AI领域的研究取得巨大进展. 作为具有实时博弈游戏常规元素的经典场景, 实时格斗游戏吸引了大量研究学者的关注, 并且在近些年得到快速发展. 相较于始终保持一致的启发式规则型策略搜索方法, 统计前向规划型方法与深度强化学习方法具备良好的环境自适应性和策略模型优化性. 尽管如此, 智能决策方法在处理格斗游戏时仍存在下述问题有待解决:

5.1　构建合理高效的前向模型

从历年FTGAIC的比赛结果可知, 统计前向规划型方法一直表现出良好的模型前向推理与环境过程适应性, 不需要大量的训练优化便可适应不同角色和任务属性下的格斗游戏需求. 然而, 统计前向规划模型需要依靠系统辨识度较高的前向模型作为因果推理器进行统计采样, 获得最优可行解. 同时, 为满足格斗游戏即时制要求, 需要简化前向模型. 这使得设计者需要平衡系统辨识度和实时性两个矛盾点. 可通过神经网络或模糊系统建模的方法设计前向模型逼近器, 快速有效地提供前向推理结果, 从而优化系统推理时间. 然而, 该方式对原始系统生成的数据分布质量和多样性要求较高, 当数据分布较为单一时, 容易影响前向建模的系统辨识准确度, 导致模型鲁棒性较差. 并且, 随着数据量的增大, 模型复杂程度也随之提高, 构成模型高系统辨识度与快前向推理. 因而在保证实时性的前提下, 进一步提升前向模型的系统辨识度是当前的研究重点之一.

5.2　深度强化学习的模型泛化性

由2020年FTGAIC比赛结果可见, 深度强化学习方法在其中的部分赛道上表现性能较优, 但是在其它一些赛道由于格斗目标的单一性和不确定性, 导致最终训练的模型表现参差不齐. 此外, 当格斗对象的角色属性发生变化时, 也会影响模型在对抗环境下的整体表现, 并且无法有效快速地调节模型的整体适应性. 目前, 为有效解决深度强化学习模型泛化性不足的问题, 常用L1与L2范数正则化[50]方式、以及增加训练环境噪音的方式来增强训练模型的鲁棒性, 通过信息熵最大化[51]与自模仿学习[52]等手段增强策略模型的环境探索能力, 采用多任务[53]的方式增训练样本的多样性, 以及自我博弈[54]的方式避免陷入局部极值解等方式进行处理. 此外, 还可设计合理高效地系统元模型训练框架, 并且构建元模型[55]学习方法以促进策略模型的系统泛化迁移以适配新环境的能力. 综上而言, 通过优化模型训练效率改善深度强化学习的模型泛化性同样是一项亟待解决的研究问题.

5.3　统计前向规划与深度强化学习结合

统计前向规划与深度强化学习作为实时格斗游戏的两类重要代表性方法, 如何高效地将统计前向规划型方法具有的环境自适应性与深度强化学习方法具有的模型优化性相结合, 增强博弈模型的快速适应和调节能力, 并且使得博弈策略模型可兼具良好的环境自适应性和模型可优化性, 已经发展成为了一个重要的研究方向. 在这方面目前有些初步性尝试, Tang等[32,56]将RHEA的前向推理能力与神经网络自适应对手模型的可优化性特点相结合, 通过改进对手模型的预测效率来改善RHEA方法的前向推理性能, 以此提升算法模型的整体性能, 并且已经成功应用在实时格斗游戏. 谷歌DeepMind提出的Muzero[57]算法将深度强化学习与MCTS方法进行深度融合, 通过深度神经网络构建前向推理规划模型, 提高神经网络模型的多层规划能力, 已经在回合制游戏和即时制游戏上取得显著成果. 进一步佐证了该研究方向的可行性以及未来潜力.

6　结论

实时格斗博弈在游戏AI领域以及实时决策博弈领域具有重要的研究意义和应用价值, 统计前向规划与深度强化学习是目前最热门且有效的机器学习方法, 被大量科研人员广泛研究与应用. 本文综述了实时格斗游戏的研究进展、方法及展望. 首先描述了实时格斗游戏的问题及特点. 然后重点介绍实时格斗游戏的主流研究方法及相关研究进展. 接着以格斗游戏人工

智能竞赛作为背景, 重点分析并梳理主流格斗游戏模型的方法特性与其对应的关键性能因子, 可用于指导算法模型性能验证, 最后对实时格斗游戏方法的未来发展趋势进行了思考与展望.

尽管本文的大部分研究方法聚焦于FTGAIC平台, 但是通过统一的算法模型测试平台, 可以直观清晰地比较算法模型间性能差异. 另外, 统计前向规划与深度强化学习作为当今最先进的游戏人工智能方法, 在形式表达和模型设计上各有利弊, 因而, 将这两大类主流计算方法进行深度融合, 形成优势互补, 必将会对整个游戏人工智能乃至通用人工智能应用发展起到极大推动作用.

参考文献:

[1] CHEN Xingguo, YU Yang. Reinforcement learning and its application to the game of Go. *Acta Automatica Sinica*, 2016, 42(5): 685 – 695.
(陈兴国, 俞扬. 强化学习及其在电脑围棋中的应用. 自动化学报, 2016, 42(5): 685 – 695.)

[2] SHAO K, TANG Z, ZHU Y, et al. A survey of deep reinforcement learning in video games. *arXiv Preprint*. Arxiv: 1912.10944v2, 2019.

[3] HSU F. IBM's Deep blue chess grandmaster chips. *Micro IEEE*, 1999, 19(2): 70 – 81.

[4] SILVER D, HUANG A, MADDISON C, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 2016, 529(7587): 484 – 489.

[5] SILVER D, SCHRITTWIESER J, SIMONYAN K, et al. Mastering the game of go without human knowledge. *Nature*, 2017, 550(7676): 354 – 359.

[6] BROWN N, SANDHOLM T. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 2018, 359(6374): 418 – 424.

[7] BROWN N, SANDHOLM T. Superhuman AI for multi-player poker. *Science*, 2019, 365(6456): 885 – 890.

[8] LI J, KOYAMADA S, YE Q, et al. Suphx: Mastering mahjong with deep reinforcement learning. *arXiv Preprint*. Arxiv: 2003.13590, 2020.

[9] YE D, LIU Z, SUN M, et al. Mastering complex control in MOBA games with deep reinforcement learning. *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. New York, USA: AAAI, 2020: 6672 – 6679.

[10] BERNER C, BROCKMAN G, CHAN B, et al. Dota 2 with large scale deep reinforcement learning. *arXiv Preprint*. Arxiv: 1912.06680, 2019.

[11] VINYALS O, BABUSCHKIN I, CZARNECKI W, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 2019, 575(7782): 350 – 354.

[12] LU F, YAMAMOTO K, NOMURA L, et al. Fighting game artificial intelligence competition platform. *Proceedings of the 2nd IEEE Global Conference on Consumer Electronics (GCCE)*. Tokyo, Japan: IEEE, 2013: 320 – 323.

[13] TURNING A. Digital computers applied to games. *Faster Than Thought*, 1953, 25: 286 – 310.

[14] SAMUEL A. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 1959, 3(3): 210 – 229.

[15] TESAURO G. Practical issues in temporal difference learning. *Machine Learning*, 1992, 8(3/4): 257 – 277.

[16] TESAURO G. Temporal difference learning and TD-gammon. *Communications of the ACM*, 1995, 38(3): 58 – 68.

[17] YANNAKAKIS G, TOGELIUS J. *Artificial Intelligence and Games*. New York, USA: Springer, 2018.

[18] SATO N, TEMSIRIRIRKKUL S, SONE S, et al. Adaptive fighting game computer player by switching multiple rule based controllers. *Proceedings of 2015 3rd International Conference on Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI)*. Okayama, Japan: IEEE, 2015: 52 – 59.

[19] KANETSUKI Y, THAWONMAS R, NAKATA S. Optimization and simplification of dynamic scripting with evolution strategy and fuzzy control in a fighting game AI. *Proceedings of the 2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*. Osaka, Japan: IEEE, 2015: 330 – 331.

[20] SPRONCK P, PONSEN M, SPRINKHUIZEN-KUYPER I, et al. Adaptive game AI with dynamic scripting. *Machine Learning*, 2006, 63(3): 217 – 248.

[21] MAJCHRZAK K, QUADFLIEG J, RUDOLPH G, et al. Advanced dynamic scripting for fighting game AI. *Proceedings of the 2015 International Conference on Entertainment Computing (ICEC)*. Trondheim, Norway: Springer, 2015: 86 – 99.

[22] BROWNE C, POWLEY E, WHITEHOUSE D, et al. A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 2012, 4(1): 1 – 43.

[23] PEREZ D, SAMOTHRAKIS S, LUCAS S, et al. Rolling horizon evolution versus tree search for navigation in single-player real-time games. *Proceedings of the 2013 Genetic and Evolutionary Computation Conference (GECCO)*. Amsterdam, Netherlands: ACM, 2013: 351 – 358.

[24] LUCAS S, SHEN Tianyu, WANG Xiao, et al. General game AI with statistical forward planning algorithms. *Chinese Journal of Intelligent Science and Technology*, 2019, 1(3): 219 – 227.
(LUCAS S, 沈甜雨, 王晓, 等. 基于统计前向规划算法的游戏通用人工智能. 智能科学与技术学报, 2019, 1(3): 219 – 227.)

[25] THUAN L, LOGOFTU D, BADIC C. A hybrid approach for the fighting game AI challenge: balancing case analysis and Monte Carlo tree search for the ultimate performance in unknown environment. *Proceedings of the International Conference on Engineering Applications of Neural Networks (EANN)*. Crete, Greece: Springer, 2019: 139 – 150.

[26] DEMEDIUK S, TAMASSIA M, RAFFE W, et al. Monte Carlo tree search based algorithms for dynamic difficulty adjustment. *Proceedings of the 2017 IEEE Conference on Computational Intelligence & Games (CIG)*. New York, USA: IEEE, 2017: 53 – 59.

[27] ISHIHARA M, ITO S, ISHII R, et al. Monte-Carlo tree search for implementation of dynamic difficulty adjustment fighting game AIs having believable behaviors. *Proceedings of the 2018 IEEE Conference on Computational Intelligence and Games (CIG)*. Maastricht, Netherlands: IEEE, 2018: 1 – 8.

[28] ISHIHARA M, MIYAZAKI T, CHU C, et al. Applying and improving Monte-Carlo tree search in a fighting game AI. *Proceedings of the 13th International Conference on Advances in Computer Entertainment Technology (ACE)*. Osaka, Japan: ACM, 2016: 1 – 6.

[29] KIM M, KIM K. Opponent modeling based on action table for MCTS-based fighting game AI. *Proceedings of the 2017 IEEE Conference on Computational Intelligence and Games (CIG)*. New York, USA: IEEE, 2017: 178 – 180.

[30] GAINA R, LUCAS S, PEREZ D. Rolling horizon evolution enhancements in general video game playing. *Proceedings of 2017 IEEE Conference on Computational Intelligence and Games (CIG)*. New York, USA: IEEE, 2017: 88 – 95.

[31] GAINA R, LIU J, LUCAS S, et al. Analysis of vanilla rolling horizon evolution parameters in general video game playing. *Proceedings of European Conference on the Applications of Evolutionary Computation (EvoApplications)*. Amsterdam, Netherlands: Springer, 2017: 418 – 434.

[32] TANG Z, ZHU Y, ZHAO D. Enhanced rolling horizon evolution algorithm with opponent model learning. *IEEE Transactions on Games*, 2021, DOI: 10.1109/TG.2020.3022698.

[33] NOGUCHI H, ISHII R, HARADA T, et al. Improving rolling horizon evolutionary algorithm in a fighting game. *Proceedings of 2019 Nicograph International (NicoInt)*. Yangling, China: IEEE, 2019: 118 – 118.

[34] ZHAO Dongbin, SHAO Kun, ZHU Yuanheng, et al. Review of deep reinforcement learning and discussions on the development of computer Go. *Control Theory & Applications*, 2016, 33(6): 701 – 717.
(赵冬斌, 邵坤, 朱圆恒, 等. 深度强化学习综述: 兼论计算机围棋的发展. 控制理论与应用, 2016, 33(6): 701 – 717.)

[35] TANG Zhentao, SHAO Kun, ZHAO Dongbin, et al. Recent progress of deep reinforcement learning: from AlphaGo to AlphaGo Zero. *Control Theory & Applications*, 2017, 34(12): 1529 – 1546.
(唐振韬, 邵坤, 赵冬斌, 等. 深度强化学习进展: 从AlphaGo到AlphaGo Zero. 控制理论与应用, 2017, 34(12): 1529 – 1546.)

[36] LIU Quan, ZHAI Jianwei, ZHANG Zongchang, et al. A survey on deep reinforcement learning. *Chinese Journal of Computers*, 2018, 41(1): 1 – 27.
(刘全, 翟建伟, 章宗长, 等. 深度强化学习综述. 计算机学报, 2018, 41(1): 1 – 27.)

[37] SUN Changyin, MU Chaoxu. Important scientific problems of multi-agent deep reinforcement learning. *Acta Automatica Sinica*, 2020, 46(7): 1301 – 1312.
(孙长银, 穆朝絮. 多智能体深度强化学习的若干关键科学问题. 自动化学报, 2020, 46(7): 1301 – 1312.)

[38] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning. *Nature*, 2015, 518(7540): 529 – 533.

[39] MNIH V, BADIA A, MIRZA M, et al. Asynchronous methods for deep reinforcement learning. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*. New York, USA: ACM, 2016: 1928 – 1937.

[40] TRI N, QUANG V, IKEDA K. Optimized non-visual information for deep neural network in fighting game. *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART)*. Setubal, Portugal: SciTePress, 2017: 676 – 680.

[41] PARK H, KIM K. Learning to play fighting game using massive play data. *Proceedings of the 2014 IEEE Conference on Computational Intelligence and Games (CIG)*. Dortmund, Germany: IEEE, 2014: 1 – 2.

[42] YOON S, KIM K. Deep Q networks for visual fighting game AI. *Proceedings of the 2017 IEEE Conference on Computational Intelligence and Games (CIG)*. New York, USA: IEEE, 2017: 306 – 308.

[43] TAKANO Y, OUYANG W, ITO S, et al. Applying hybrid reward architecture to a fighting game AI. *Proceedings of the 2018 IEEE Conference on Computational Intelligence and Games (CIG)*. Maastricht, Netherland: IEEE, 2018: 1 – 4.

[44] KIM D, PARK S, YANG S. Mastering fighting game using deep reinforcement learning with self-play. *Proceedings of the 2020 IEEE Conference on Games (CoG)*. Osaka, Japan: IEEE, 2020: 576 – 583.

[45] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms. *arXiv Preprint*. Arxiv:1707.06347, 2017.

[46] TAKANO Y, ITO S, HARADA T, et al. Utilizing multiple agents for decision making in a fighting game. *Proceedings of 2018 7th Global Conference on Consumer Electronics (GCCE)*. Nara, Japan: IEEE, 2018: 594 – 595.

[47] ZHU Y, ZHAO D. Online minimax Q network learning for two-player zero-sum Markov games. *IEEE Transactions on Neural Networks and Learning Systems*, 2022, 33(3): 1228 – 1241.

[48] XU Xiaoliang, WANG Leyu, ZHOU Hong. Implementation framework of finite state machines. *Journal of Engineering Design*, 2003, 10(5): 251 – 255.
(徐小良, 汪乐羽, 周泓. 有限状态机的一种实现框架. 工程设计学报, 2003, 10(5): 251 – 255.)

[49] HAARNOJA T, ZHOU A, ABBEEL P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *Proceedings of the 35th International Conference on Machine Learning (ICML)*. Stockholm, Sweden: ACM, 2018, 1861 – 1870.

[50] LI L, LI D, SONG T, et al. Actor-critic learning control based on L2-regularized temporal-difference prediction with gradient correction. *IEEE Transactions on Neural Networks and Learning Systems*, 2018, 29(12): 5899 – 5909.

[51] ZHAO R, SUN X, TRESP V. Maximum entropy-regularized multi-goal reinforcement learning. *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Long Beach, USA: ACM, 2019: 7553 – 7562.

[52] LI W, ZHU Y, ZHAO D. Missile guidance with assisted deep reinforcement learning for head-on interception of maneuvering target. *Complex & Intelligent Systems*, 2021, DOI: 10.1007/s40747-021-00577-6.

[53] ZHANG Yu, LIU Jianwei, ZUO Xin. Survey of multi-task learning. *Chinese Journal of Computers*, 2020, 43(7): 1340 – 1378.
(张钰, 刘建伟, 左信. 多任务学习. 计算机学报, 2020, 43(7): 1340 – 1378.)

[54] BAI Y, JIN C. Provable self-play algorithms for competitive reinforcement learning. *Proceedings of the 37th International Conference on Machine Learning (ICML)*. Vienna, Austria: ACM, 2020: 551 – 560.

[55] SCHWEIGHOFER N, DOYA K. Meta-learning in reinforcement learning. *Neural Networks*, 2003, 16(1): 5 – 9.

[56] LIANG R, ZHU Y, TANG Z, et al. Proximal policy optimization with elo-based opponent selection and combination with enhanced rolling horizon evolution algorithm. *IEEE Conference on Games (CoG)*. Copenhagen, Denmark: IEEE, 2021: 1 – 4.

[57] SCHRITTWIESER J, ANTONOGLOU I, HUBERT T, et al. Mastering atari, Go, chess and shogi by planning with a learned model. *Nature*, 2020, 588(7839): 604 – 609.

作者简介:

唐振韬　博士, 研究方向为强化学习、深度学习等, E-mail: tangzhentao2016@ia.ac.cn;

梁荣钦　硕士研究生, 研究方向为强化学习、深度学习等, E-mail: liangrongqin2020@ia.ac.cn;

朱圆恒　博士, 副研究员, 研究方向为深度强化学习、自适应动态规划等, E-mail: yuanheng.zhu@ia.ac.cn;

赵冬斌　博士, 研究员, 研究方向为深度强化学习、自适应动态规划、智能交通、机器人等, E-mail: dongbin.zhao@ia.ac.cn.