基于知识及 A3C 算法的兵棋推演智能决策模型研究

彭益辉[1] 周献中[1,2] 孙宇祥[1] 李 斌[1]

摘 要 针对地图范围大、奖励稀疏、动作空间和状态空间复杂的智能兵棋推演环境，单纯的深度强化学习算法存在无法快速收敛、与规则智能体对抗效果差的问题，提出了异步优势动作评价算法（Asynchronous Advantage Actor-Ccritic，A3C）解决收敛速度问题。在 A3C 强化学习算法中添加嵌入式专家经验机制，通过设置特殊条件奖励和构建知识驱动子模块，进一步提高收敛速度和训练效果。文章在智能兵棋推演平台上进行仿真实验，验证了该算法的可行性和实用性。

关键词 深度强化学习，A3C 算法，知识驱动，智能兵棋

Intelligent Decision Model of Wargaming Based on Knowledge and A3C Algorithm

PENG Yi-Hui[1] ZHOU Xian-Zhong[1,2] SUN Yu-Xiang[1] LI Bin[1]

Abstract Aiming at the intelligent wargaming deduction environment with large map range, sparse rewards, and complex action space and state space, the simple deep reinforcement learning algorithm cannot converge quickly, and the confrontation effect with the rule agent is poor, Asynchronous advantage actor-critic algorithm is proposed to resolve convergence issues, In order to further improve the training effect of agents, an embedded expert experience mechanism is added to the A3C reinforcement learning algorithm, and the training effect is enhanced by setting special condition rewards and building knowledge-driven sub-modules. Finally, simulation experiments are carried out on the intelligent wargaming deduction platform, which verifies the feasibility and practicability of the algorithm.

Key words deep reinforcement learning, advantage actor-critic algorithm, knowledge-driven, intelligent wargaming

近年来，强化学习、深度学习等机器学习技术被广泛应用于对抗类游戏中，其中，最具代表性的如围棋领域的 AlphaGO 和星际争霸中的 AlphaStar，甚至多次击败人类世界冠军选手，兵棋推演作为实时对抗的军事指挥类游戏，更加需要快速的态势感知和决策能力，这恰恰是人工智能所擅长的[1-17]。因此，越来越多的研究将人工智能技术融入到兵棋推演当中，使算子具备一定的智能性，也取得了不错的成果[4,5,18]。然而，奖励稀疏[6,15,16]、状态空间巨大、决策空间巨大、战局信息不完全、评估调优难度大等因素导致强化学习算法在兵棋推演环境下无法快速收敛或存在智能决策水平较低等问题[7,10,13]。

为此，本文针对上述问题提出了一种基于 A3C（Asynchronous advantage actor critic）强化学习算法，采用强化学习算法模块与知识驱动子模块相结合的框架[8,9,19]，加快智能体收敛速度并提高智能决策效果[11,12,14]。

1 强化学习背景

强化学习（Reinforcement Learning，RL）是机器学习中的一大类，强化学习不被告知如何学习，而是通过不断的试错（Trial and Error），最后找到规律，学会达到目的的方法，强化学习中许多比较有名的算法，比如根据行为价值选择行为的 Q-learning 算法和直接输出行为的 policy gradient 算法。而 actor-critic 算法则是 Q-learning 和 policy gradient 的结合，A3C 算法就是 actor-critic 算法的优化算法，在 advantage actor-critic（A2C）算法基础上增加异步并行结构而来。

国家自然科学基金（61876079）资助
Supported by National Natural Science Foundation（61876079）

1. 南京大学工程管理学院 江苏 南京 210008 2. 智能装备新技术研究中心 江苏 南京 210008
1. College of Engineering Management, Nanjing University, Nanjing Jiangsu 210008, China 2. Intelligent Equipment New Technol-ogy Research Center Nanjing University, Nanjing Jiangsu 210008, China

1.1 Advantage Actor-Critic 算法

Actor-Critic 算法结合了 Policy Gradient（actor）和 Q-learning（critic），其中 actor 基于概论选择对应的行为，而 critic 是基于 actor 选择的行为进行评价，actor 根据 critic 的评分修改选择行为的概率。Policy gradient 如式（1）所示：

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T_n}\left(\sum_{t'=t}^{T_n}\gamma^{t'-t}r_{t'}^n - b\right)\nabla \log p_\theta(a_t^n \mid s_t^n) \tag{1}$$

Policy gradient 是基于回合进行更新，因此，我们需要大量的时间与环境交互产生样本进行回合训练，而且每个回合并不是稳定的。

Q-Learning 学习的是每个动作的价值，要求动作必须是离散的，policy gradient 和 Q-learning 都有各自的优缺点，因此，可以将二者结合起来，引入 critic，用动作的价值替换回合奖励：

$$\sum_{t'=t}^{T_n}\gamma^{t'=t}r_{t'}^n = Q^{\pi_\theta}(s_t^n, a_t^n) \tag{2}$$

对于基准线 b（baseline）可以使用 critic 中的 V function 进行替换：

$$b = V^{\pi_\theta}(s_t^n) \tag{3}$$

将式（2）、式（3）代入式（1）中，得到 actor-critic 算法如下式：

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T_n}\left(Q^{\pi_\theta}(s_t^n, a_t^n) - V^{\pi_\theta}(s_t^n)\right)\nabla \log p_\theta(a_t^n \mid s_t^n) \tag{4}$$

actor-critic 算法仍然具有缺陷，在式（4）中，具有两个网络 $Q^{\pi_\theta}(s_t^n, a_t^n)$ 和 $V^{\pi_\theta}(s_t^n)$，两个网络同时更新，影响结果的稳定性。考虑到 $Q^{\pi_\theta}(s_t^n, a_t^n)$ 是衡量状态 s_t 下动作 a_t 的值，实际上等于动作 a_t 下的奖励 r 与下个状态 s_{t+1} 的价值之和：

$$Q^{\pi_\theta}(s_t^n, a_t^n) = r_t^n + V^{\pi}(s_{t+1}^n) \tag{5}$$

因此，在 actor-critic 算法上的进一步优化，将式（5）代入式（4）中，得到下式：

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T_n}\left(r_t^n + V^{\pi}(s_{t+1}^n) - V^{\pi_\theta}(s_t^n)\right)\nabla \log p_\theta(a_t^n \mid s_t^n) \tag{6}$$

A2C 算法的参数梯度更新方式如式（6）所示。

1.2 A2C 算法训练流程

A2C 算法流程如图 1 所示。

（1）actorπ 与环境做互动收集资料。

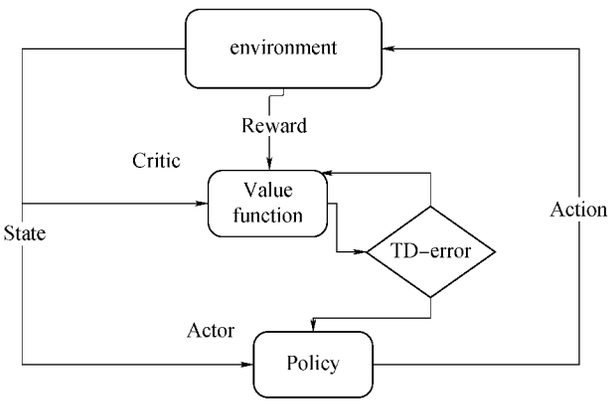

图 1 A2C 算法训练流程

Fig. 1 A2C algorithm training flow chat

（2）使用 TD 以步骤（1）中的资料去估计 value function。

（3）套用式（4）更新 π。

（4）生成新的 actorπ'，重复上述三个步骤……

2 基于知识驱动及 A3C 的算法框架

2.1 总体框架

在实时类兵棋推演环境中，地图范围大、动作空间复杂、奖励稀疏，如果直接采用随机初始化参数的动作神经网络进行训练，智能体训练前期需要漫长的时间去探索，其中大多为无效动作，难以获得奖励，这将影响收敛速度和训练效果[15]。为了解决这个问题，本文提出在强化学习算法模块中，嵌入基于知识驱动的子模块，通过细化行动细节，形成动作列表，降低动作维度，减少无效动作，引导获取奖励。改进后的强化学习训练框架如图 2 所示。

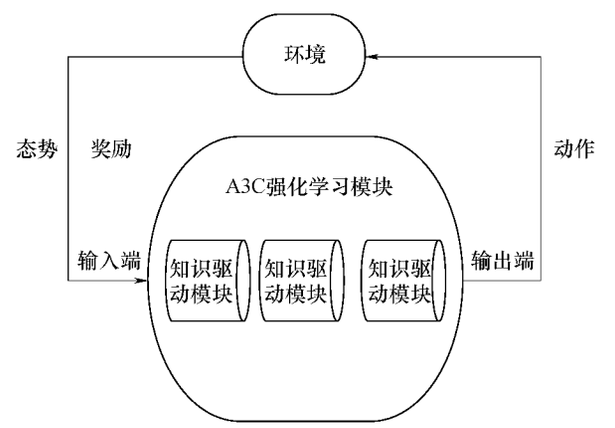

图 2 训练框架

Fig. 2 Training framework

在图 2 中，框架输入量为初始化的战场态势信息，获取到态势信息后，进行态势处理，数据经过清洗、筛选、提取、打包、归一化以及格式化表示，强化学习模块根据战场态势调用知识驱动的子模块，输出动作决策，控制智能体与环境进行交互，将新一轮的战场实时态势作为模块的输入量输入强化学习网络中，重复以上的过程进行训练。

2.2 A3C 算法模块

2.2.1 A3C 算法框架

A3C 算法属于 actor-critic 算法的一种优化算法，在 Advantage Actor-critic 算法的基础上增加异步框架优化而来，这种架构的提出更多是出于实际部署和算法收敛速度方面的考虑，A3C 算法的框架如图 3 所示。

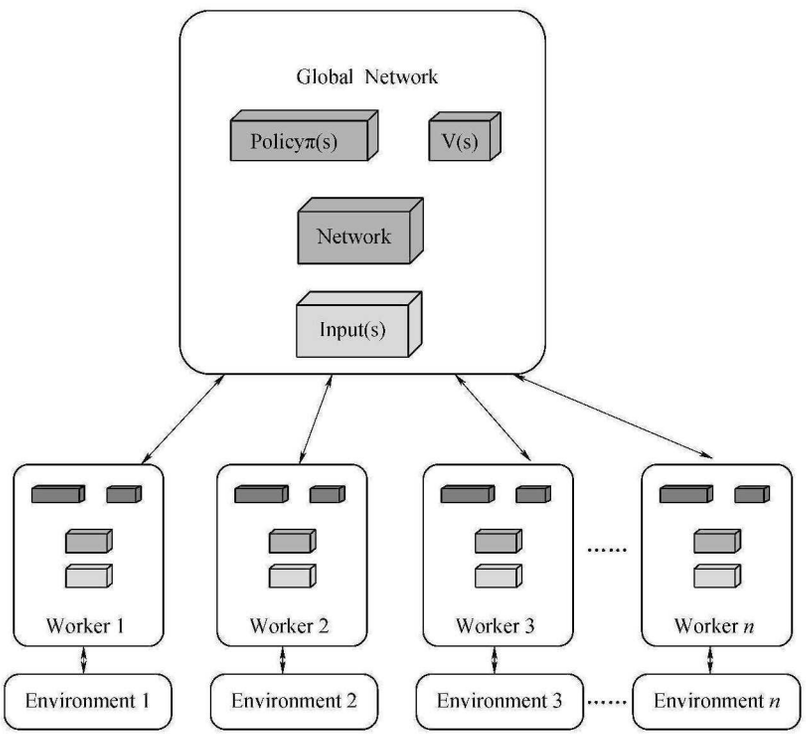

图 3 A3C 算法框架

Fig. 3 A3C algorithm framework

整个模型结构的上方是全局的网络，该网络并不直接与环境进行交互，下方若干个并行的网络是直接与环境进行交互的 worker。

每一个 worker 中都包括 Actor 和 Critic 两个部分，当 worker 与环境交互，得到一定量的数据后，计算在自己线程里的神经网络损失函数的梯度，参数梯度计算方式如上式（6），但是这些梯度却并不更新自己线程的神经网络，而是 n 个线程会独立的使用累积的梯度分别更新公共部分的神经网络模型参数。

每隔一段时间，线程会将自己的神经网络的参数更新为公共神经网络的参数，进而指导后面的环境交互。主结构的参数更新受到副结构提交更新的不连续性干扰，所以更新的相关性被降低，收敛性提高。

2.2.2 A3C 算法训练流程

A3C 算法实质上是将 Advantage Actor-critic 放进多个线程中同步训练，单线程中的经验上传给 global，更新 global 参数。因此，以一个线程的训练流程为例对 A3C 算法流程进行描述，单线程的算法流程如图 4 所示。

具体训练流程如下：

（1）每个 worker 从 global network 复制参数。

（2）不同的 worker 与环境互动。

（3）每个 worker 生成各自的 gradient，但不用于更新自身网络参数。

（4）不同的 worker 把自身的 gradient 传回 global network。

（5）Global network 接受 gradient 后更新自身参数。

A3C 算法伪代码如下所示：

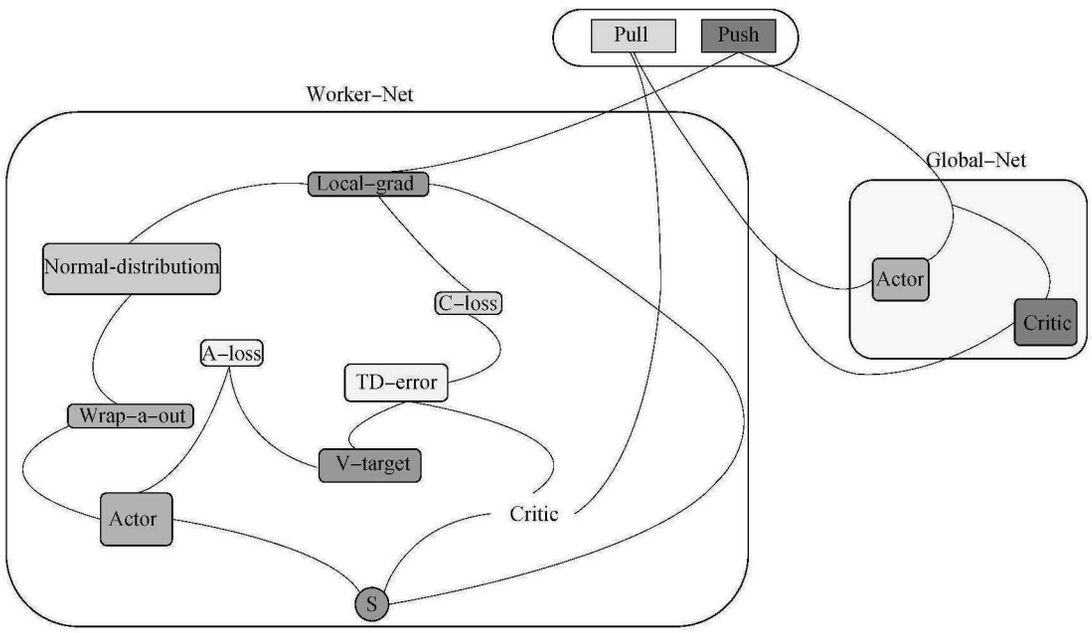

图 4 A3C 算法训练流程

Fig. 4 A3C algorithm training flow chat

Algorithm 1 Asynchronous advantage actor-critic-pseudocode for each actor-learner thread

1://Assume global shared parameter vectors θ and θ_v and global shared counter $T=0$

2://Assume thread-specific parameter vectors θ' and θ'_v

3://Initialize thread step counter $t\leftarrow 1$

4:repeat

Reset gradients: $d\theta\leftarrow 0$ and $d\theta_v\leftarrow 0$

Synchronize thread-specific parameters $\theta'=\theta$ and $\theta'_v=\theta_v$

$t_{start}=t$

Get state s_t

repeat

Perform a_t according to policy $\pi(a_t|s_t;\theta')$

Receive reward r_t and new state s_{t+1}

$t\leftarrow t+1$

$T\leftarrow T+1$

until terminal s_t or $t-t_{start}=t_{max}$

$$R=\begin{cases}0, \text{for terminal } s_t\\ V(s_t,\theta'_v), \text{for non-terminal } s_t \text{//Bootstrap from last state}\end{cases} \quad (1)$$

5:for $t=t-1,\cdots,t_{start}$ do

6: $R\leftarrow r_i+\gamma R$

7: Accumulate gradients wrt θ': $d\theta\leftarrow d\theta+\nabla\log\pi(a_i|s_i;\theta')(R-V(s_i;\theta'_v))$

8: Accumulate gradients wrt θ'_v: $d\theta_v\leftarrow d\theta_v+\delta(R-V(s_i;\theta_v))^2/\delta\theta'_v$

9:end for

10:Perform asynchronous update of θ using $d\theta$ and θ_v using $d\theta_v$

11:until: $T>T_{max}$

2.3 知识驱动子模块

强化学习需要智能体与环境交互并得到大量高质量的数据用于训练智能体，而智能兵棋推演中奖励稀疏和动作空间巨大的导致单纯的强化学习大部分动作均无法获得奖励。

针对以上问题，本文将由智能兵棋推演系统领域知识和规则构成的知识驱动子模块融入强化学习模块之中，将复杂推演过程、行为决策和单元控制等问题进行模块化、离散化。通过调用知识驱动的子模块来进行行为决策，达到降低动作维度、引导奖励获取、加速算法收敛的目的。图 1 中基于知识驱动的子模块由四类模块构成，分别为预警任务模块、防御巡逻任务模块、进攻巡逻任务模块和反舰任务模块，各个模块的具体构成和输出动作见 3.3.2 节。

3 仿真实验

本文实验仿真平台为“墨子·未来指挥官系统（个人版）”，该系统以现代空海战推演为主，覆盖海陆空天点全域联合作战，支持作战筹划、作战组织、兵力指挥和效能评估的全流程推演，实验台式机配置为 Intel（R）Core（TM）i7-8700K CPU @3.70GHz 3.70GHz 内存 32.0GB

3.1 实验想定

以某国东部海域为红方 HM 编队活动区域，区域大小约 150n mile×100n mile，区域中心距离某岛南部 430n mile，距离某国东海岸 235n mile，设置 HM 本舰活动路线沿区域边界运动，红色方兵力由

水面舰艇和航空兵组成，蓝方兵力由某岛航空兵组成，设置两个机场。想定示意图如图 5 所示。

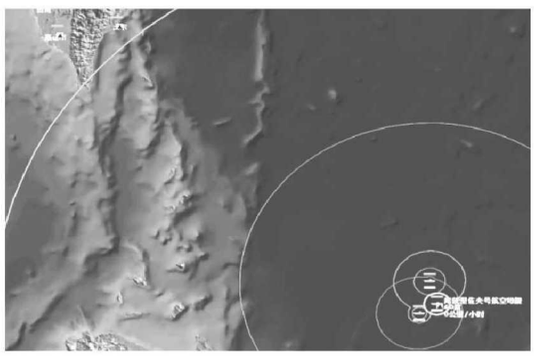

图 5 想定示意图

Fig. 5 Scenario diagram

红蓝双方兵力具体数量、机型和部署区域如表 1 和表 2 所示。

表 1 红方兵力表

Table 1 The red side troops table

类别	装备型号	数量	部署位置	备注
水面舰艇	库兹涅佐夫号航空母舰	1	设定区域	
	现代级导弹驱逐舰	1	11M 编队	
	克里瓦克级导弹护卫舰	1	11M 编队	
航空兵	米格-29 型战斗机	10	11M 本舰	
	卡-29 警戒直升机	4	11M 本舰	

表 2 蓝方兵力表

Table 2 The blue side troops table

类别	装备型号	数量	部署位置	备注
航空兵	F-16A 型战斗机	9	1 号机场	
	F-16A 型战斗机	9	2 号机场	
	EC-130H 电子战飞机	1	1 号机场	
	EC-130H 电子战飞机	1	2 号机场	
	E-2K 型鹰眼预警机	1	1 号机场	

3.2 模型构建

根据第 2 节的模型框架，构建基于知识驱动的 A3C 强化学习算法模型，并对强化学习中的要素进行设置和相关说明。

3.2.1 奖励函数

为了解决兵棋推演过程中奖励稀疏而导致训练收敛速度慢的问题，本文中将专家经验引入到强化学习的奖励函数中，从而引导智能体获取奖励，加快训练收敛速度。将奖励设置为两个部分，战损奖励 score 和基于专家经验奖励。

基于专家经验的额外奖励设置为我方战斗机与敌方单元之间的距离减少量，奖励函数 r 设置如下式：

$$r=\text{score}+\gamma(\text{dis}_{pre}-\text{dis}_{next}) \tag{7}$$

$$\gamma=(\gamma-(5\text{e}-5))/100000 \tag{8}$$

式（7）中，dis_{next}为下一位置与敌方目标单元的距离，dis_{pre}为当前位置与敌方目标单元距离，γ 为衰减系数，如式（8）所示，随着时间的增加而衰减。

在训练初期，由于没有战损奖励，因此，主要由专家经验的额外奖励为引导，控制智能体向敌方所在方向飞行，到接战阶段，此时的奖励便由战损奖励主导。

3.2.2 动作空间

为了降低决策动作的维度，将智能体在连续空间中的控制用任务的形式进行离散化。动作空间由任务类型、任务目标位置、行动单元组成。

（1）任务类型。预警、防御巡逻、进攻巡逻、反舰。

（2）任务目标位置。我方舰队位置，我方空中单元位置，敌方舰队位置，敌方空中单元位置。

（3）行动单元。F-16A，E3C-130H，E-2K。

因此，基于知识的子模块动作空间为 336（4×4×21）和 nothing（不采取动作），共计 337 维。

3.2.3 状态空间

状态空间中包含己方、敌方两部分信息，具体信息类型如下：

（1）己方单元。速度、高度、经度、纬度、航向、任务类型、区域内己方单元、区域内敌方单元、己方空闲单元，共计 9 维信息。

（2）己方导弹。速度、高度、经度、纬度、航向、剩余量，共计 6 维信息。

（3）敌方单元。速度、高度、经度、纬度、航向共计 5 维信息。

（4）敌方导弹。速度、高度、经度、纬度、航向，共计 5 维信息。

因此，状态空间 1350 维（9×6×5×5），由于战争迷雾的存在，兵棋推演中双方信息不对等，因此，暂未获得的信息均记作 0。

3.2.4 参数及作战条令设置

实验中所涉及的各种超参数设置如表 3 所示。

表 3 实验超参数设置

Table 3 Experimental hyperparameter setting

参数名称	参数值
网络优化器	SharedAdam
学习率	5e-5

续表

参数名称	参数值
betas	0.92-0.99
折扣率	0.95
Epsilon	1e-5
衰减系数	1.0

4 实验结果分析

本文为了验证 A3C 强化学习算法结合知识驱动子模块算法的有效性，设置对比实验，记录 A3C+rule、原始 A3C、传统规则 AI 三种模型在相同场景下，10 轮（每轮 200 局）训练的平均得分，具体得分如图 6 所示。

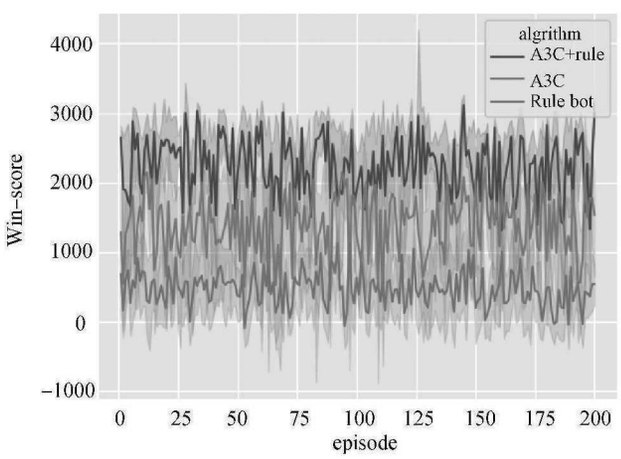

图 6 得分示意图

Fig. 6 Score diagram

从图 6 中可以看到，经过 2000 局的训练后，A3C+rule 算法模型平均得分集中在 2000～3000，而原始 A3C 算法模型和 Rule-bot 模型平均得分分别在 1000～2000 和 0～1000。A3C+rule 算法模型和原始 A3C 算法模型与规则 AI 相比，在得分上均有不同程度的提升。

三种算法的胜率曲线如图 7 所示，A3C+Rule 算法模型在训练初期，受到专家经验奖励和基于知识驱动的子模块的指导，胜率接近 60%，原始 A3C 算法胜率不足 50%，低于传统规则 AI。经过 2000 局的训练，A3C+Rule 算法和原始 A3C 算法随着训练次数的增加在胜率方面均有不同程度的提升，A3C+Rule 算法达到 86.15%，原始 A3C 算法达到 63.8%。

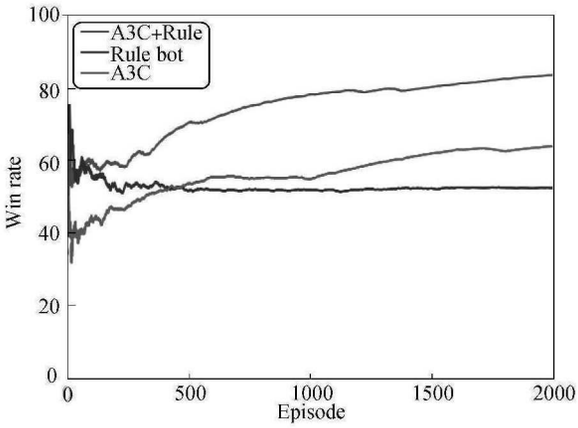

图 7 胜率示意图

Fig. 7 Win rate diagram

三种算法模型的胜场、胜率统计如表 4 所示：

在有限的训练次数下基于知识驱动及 A3C 算法胜率和得分提升更为显著，收敛速度优于原始 A3C 算法模型。

表 4 胜率表

Table 4 The win rate table

算法模型	胜场	总场次	胜率
A3C+Rule	1723	2000	86.15%
A3C	1276	2000	63.8%
Rule bot	1009	2000	50.45%

5 结论

针对强化学习在复杂兵棋推演环境中无法快速收敛、对抗规则智能体效果差的问题，本文提出了 A3C 强化学习算法模块和知识驱动子模块相结合的算法框架并融入专家经验，用以加快收敛速度、提高实验效果。

为了验证算法的可行性，设置了两个对比实验，验证该改进算法在作战单元众多、奖励稀疏、动作空间和状态空间巨大的对抗环境中收敛问题，实验结果验证，本文提出的 A3C 改进算法相比于传统规则 AI、原始 A3C 算法具有明显的优势，能够在有限的训练回合中取得良好的效果。在模型的训练过程中，涉及了海空天电单元的协同合作问题，不同类型作战单元之间的协同战术战法，对指挥员在瞬息万变的战场态势下如何快速、合理决策有着较大的参考价值。

为了降低动作空间的维度，本文采取了对作战单元高度、航向、速度等参数的离散化处理，以作战编队的形式控制作战单元，虽然该算法模型在实

验中取得了不错的成果，但是并未对不同的强化学习算法进行分析对比，在未来的工作中，可以进一步地拓展该框架的算法种类和增加对作战单元的细节控制等。

References

1 孙宇祥，周献中，唐博建，等. 智能指挥与控制系统发展路径与未来展望［J］. 火力与指挥控制，2020，45（11）：60-66.

2 孙宇祥，黄孝鹏，周献中，等. 基于知识的海战场态势评估辅助决策系统构建［J］. 指挥信息系统与技术，2020，11（4）：15-20.

3 胡晓峰，贺筱媛，陶九阳. Alpha Go 的突破与兵棋推演的挑战［J］. 科技导报，2017，35（21）：49-60.

4 戴勇，黄杏花. 人工智能在计算机兵棋推演领域的应用［J］. 集成电路应用，2020，37（5）：67-69.

5 崔文华，李东，唐宇波，等. 基于深度强化学习的兵棋推演决策方法框架［J］. 国防科技，2020，41（2）：113-121.

6 杨惟轶，白辰甲，蔡超，等. 深度强化学习中稀疏奖励问题研究综述［J］. 计算机科学，2020，47（3）：182-191.

7 李晨溪，曹雷，张永亮，等. 基于知识的深度强化学习研究综述［J］. 系统工程与电子技术，2017，39（11）：2603-2613.

8 冯超，景小宁，李秋妮，等. 基于隐马尔科夫模型的空战决策点理论研究. 北京航空航天大学学报（自然科学版），2017，43（3）：615-626.

9 何旭，景小宁，冯超. 基于蒙特卡洛树搜索方法的空战机动决策 . 空军工程大学学报（自然科学版），2017，18（5）：36-41.

10 KARELAHTI J，VIRTANEN K，RAIVIO T. Near－optimal missile avoidance trajectories via receding horizon control［J］. Journal of Guidance Control and Dynamics，2015，30（5）：1287-1298.

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

12 LIU P，MA Y. A deep reinforcement learning based intelligent decision method for UCAV air combat［J］. In：Proceeding of Asian Simulation Conference，Berlin：Springer，2017. 274-286.

13 ONTANÓN S，SYNNAEVE G，URIARTE A，et al. A survey of real-time strategy game ai research and competition in starcraft［J］. IEEE Transactions on Computational Intelligence and AI in games，2013，5（4）：293-311.

14 JADERBERG M，CZARNECKI W M，DUNNING I，et al. Human－level performance in 3d multiplayer games with population－based reinforcement learning［J］. Science，2019，364（6443）：859-865.

15 PATRICK M，et al.，Reward shaping for knowledge based multi-objective multi-agent reinforcement learning［J］. The Knowledge Engineering Review，2018.

16 ZHANG Y，ROSENDO A. Tactical Reward Shaping：Bypassing Reinforcement Learning with Strategy-Based Goals［J］. CoRR，2019.

17 WILLEMSEN D，BAIER H，KAISERS M. Value targets in off-policy AlphaZero：a new greedy backup［J］. Neural Computing and Applications，2021.

18 HEREDIA P C，MOU S. Distributed Multi-Agent Reinforcement Learning by Actor-Critic Method［J］. IFAC PapersOnLine，2019，52（20）.

19 WILLEMSEN D，BAIER H，KAISERS M. Value targets in off-policy AlphaZero：a new greedy backup［J］. Neural Computing and Applications，2021.