doi: 10.7690/bgzdh.2022.07.001

基于规划步数自适应 Dyna-Q 的多功能雷达干扰决策方法

朱霸坤，朱卫纲，李　伟，李佳芯，杨　莹
（航天工程大学电子与光学工程系，北京 101416）

摘要：针对基于强化学习的干扰决策方法存在着收敛速度过慢的问题，在 Dyna-Q 算法的基础上提出一种规划步数自适应的 Dyna-Q 干扰决策算法。在保证干扰策略有效性的前提下，提升强化学习算法的收敛速度，使算法能以更快的速度学习到最优干扰策略。实验与仿真结果表明：该算法能实现多功能雷达干扰的实时有效，也可扩展到其他强化学习应用领域，具有一定借鉴价值。

关键词：多功能雷达；干扰决策；强化学习；Dyna-Q；自适应
中图分类号：TN972　　**文献标志码：**A

Multi-function Radar Jamming Decision Method Based on Planning Steps Adaptive Dyna-Q

Zhu Bakun, Zhu Weigang, Li Wei, Li Jiaxin, Yang Ying
(*Department of Electronic and Optical Engineering*, *Space Engineering University*, *Beijing* 101416, *China*)

Abstract: Aiming at the problem of slow convergence speed of jamming decision method based on reinforcement learning, a jamming decision algorithm with selfadaptive planning steps based on Dyna-Q algorithm is proposed. On the premise of ensuring the effectiveness of the jamming strategy, the convergence speed of the reinforcement learning algorithm is improved, so that the algorithm can learn the optimal jamming strategy at a faster speed. The experimental and simulation results show that the algorithm can realize the real-time and effective jamming of multi-function radar, and can also be extended to other reinforcement learning applications, which has a certain reference value.

Keywords: multi-functional radar; jamming decision; reinforcement learning; Dyna-Q; selfadaptive

0　引言

多功能雷达干扰决策是指在多功能雷达对抗中选择干扰样式的过程。多功能雷达是现代战场上的重要用频设备，对于战争胜负起着至关重要的作用，而干扰决策是决定干扰成功与否的关键。基于 SVM[1]、博弈论[2-5]、D-S 证据理论[6]等传统的干扰决策方法依靠从大量历史数据中学习干扰策略，但对于波形灵活多变、自适应能力强的多功能雷达难以发挥作用；基于强化学习的干扰决策方法[7-10]，虽然具备自学习的能力，但算法收敛速度仍不够快，直接影响了 MFR 干扰的实时性和有效性。笔者提出基于规划步数自适应 Dyna-Q 的干扰决策方法，能够提升基于强化学习干扰方法的收敛速度。

1　强化学习与多功能雷达干扰决策问题

1.1　强化学习

强化学习是目前机器学习的热门领域之一。不同于监督学习和无监督学习，强化学习不需要事先准备的样本数据，也不需要样本标签。强化学习通过智能体与环境的交互产生数据，通过这些数据产生学习经验，进而利用经验改进策略。强化学习的目的是希望智能体在与环境的交互中，产生一个指导动作的行动策略，通过该行动策略产生的行动序列使智能体获得最大的累计收益。强化学习的交互过程如图 1 所示。智能体在 t 时刻对环境施加动作 A_t，环境在动作 A_t 的作用下，发生状态的转移，由状态 S_t 转移到状态 S_{t+1}；同时给智能体反馈一个收益 R_{t+1}，智能体在接收到来自环境的收益后，利用收益调整自身的行动策略，进而进行下一步工作。

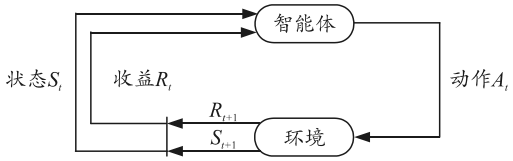

图 1　强化学习的交互过程

1.2　多功能雷达干扰决策 MDP 模型

通常，用一个马尔科夫决策过程（Markov decision process，MDP）来描述强化学习。一个 MDP 模型包括 5 个要素，表示为$\{S, A, P, \gamma, R\}$，其中：S

收稿日期：2022-03-22；**修回日期：**2022-04-28
基金项目：复杂电磁环境效应国家重点实验室项目（2020Z0203B）
作者简介：朱霸坤（1997—），男，云南人，硕士，从事认知电子战、多功能雷达、强化学习研究。E-mail: 2474519020@qq.com。

为状态集；A 为动作集；P 为环境状态转移概率；γ 为折扣率，表示未来收益的折现值；R 为收益集合。而在多功能雷达干扰决策问题中，也可以采用一个 MDP 模型来描述。

1) 状态集 S 对应雷达状态集。

由于多功能雷达特殊的任务调度机制[11]和自适应机制[12-13]，多功能雷达的工作过程可以描述为多功能雷达内部一系列雷达状态的相互切换过程。其中，每种雷达状态都是一个雷达信号参数相对稳定的过程，对于不同的雷达干扰样式，不同的雷达状态将会有不同的反应表现。用 s_1, s_2, …, s_m 来表示不同的雷达转态。

2) 动作集 A 对应干扰样式集。

干扰样式集是智能干扰机可以产生的干扰样式的集合，干扰决策的工作就是在对多功能雷达干扰的过程中选择合适的干扰样式。用 $j_1, j_2, \cdots, j_n$ 来表示不同的干扰样式。

3) 环境状态转移概率 P 对应雷达状态转移概率。

雷达状态转移概率是由多功能雷达内部工作机理所决定的，对于干扰方而言，雷达状态转移概率是未知的。通过对多功能雷达的任务调度机制和自适应机制的分析，多功能雷达状态之间存在着马尔可夫性，因此可以用 $p(S_{t+1}|S_t, J_t)$形式的概率来表示多功能雷达状态转移概率。

4) 收益集合 R 对应智能干扰机与多功能雷达交互对抗过程中产生的所有收益的集合。

收益反应了智能干扰机干扰任务所要达到的目标。在基于强化学习的智能干扰决策过程中，干扰的过程可以描述为是一个序列决策的过程，干扰任务的目标即为将当前的雷达状态转移到目标雷达状态，目标状态用 S_{aim} 表示。收益与雷达状态转移情况有关，用函数 r 表示。一般可以将 r 设置为[10]：

$$r(S_{t+1}|S_t)=\begin{cases}-1, & S_{t+1}\neq S_{\text{aim}}\\ 100, & S_{t+1}=S_{\text{aim}}\end{cases}。\tag{1}$$

当雷达状态转移到目标雷达状态 S_{aim}，干扰决策智能体获得收益 100，而其余情况获得的收益为-1。这样就可以保证，当雷达状态以最快的速度转移到目标雷达状态时，智能体获得的收益是最大的；因此，累计收益最大与多功能雷达干扰任务的目标保持一致。

通过上述分析可知，多功能雷达干扰决策问题被建模为一个 MDP 模型，可使用强化学习的方法来解决此类决策问题。

2 基于规划步数自适应 Dyna-Q 的干扰

2.1 Dyna-Q 算法

在广义的强化学习定义中，可根据算法对模型的依赖情况将强化学习算法分为模型相关的强化学习和模型无关的强化学习。前者依赖模型进行迭代，具有更快的收敛速度；后者由于不具备模型，需进行大量的交互学习才能完成策略的收敛。Dyna 架构的思想充分利用了 2 种算法的优势，不仅在与环境交互的过程中更新策略，而且也不断地构建和完善外界关于外界环境的评估模型，在产生评估模型后又利用该评估模型进行模型相关的强化学习加速算法收敛速度。Dyna-Q 算法是其中最为常用的算法，在 Dyna-Q 中，模型无关的强化学习算法为 Q-Learning 算法[14]，模型无关的强化学习在本文中采用随机采样单步表格型 Q 规划[15]。

在 Q-Learning 中，Q 值也称状态动作值。记为 $Q_\pi(s, a)$，智能体通过 Q 值选择动作和进行策略学习。在选择动作时，采用的策略为 ε-greedy 策略。ε-greedy 策略以 ε 的概率随机选择动作，以 $1-\varepsilon$ 的概率选择使得当前的 $Q_\pi(s, a)$最大动作。在进行策略学习时，采用的迭代公式为：

$$\begin{aligned}Q(s,a)\leftarrow Q(s,a)+\alpha[r(s,a)+\\ \gamma\max Q(s',a')-Q(s,a)]。\end{aligned}\tag{2}$$

其中：α 为学习率；s', a' 为下一时刻的状态和动作；r 为收益函数。

在采用随机采样单步表格型 Q 规划中，环境的估计模型被建模为一个表格。在多功能雷达干扰决策问题中，就是一个行索引为当前雷达状态，列索引为干扰样式的表格，表格中的元素为下一时刻的雷达状态，记为 Model(s, j)。

规划学习的步数记为 λ，在每一个规划步数，从 Model(s, j)中随机选择雷达状态和干扰样式作为规划学习中当前的雷达状态和当前的干扰样式，则进一步可以通过 Model(s, j)得到下一时刻的雷达状态。通过规划得到的结果利用与 Q-Larning 相同的公式对策略进行迭代更新，从而完成策略的更新。

2.2 规划步数自适应 Dyna-Q 算法

在 Dyna-Q 算法中，规划步数是一个十分重要的参数，规划步数的设置会影响算法的性能。规划步数太小，加速算法收敛速度的效果会不明显；而规划步数太大又会造成算法陷入局部最优。强化学

习智能体在训练的过程中，会经历多个回合，每个回合到达雷达状态所需的时间称为时间步数。

在强化学习智能体的训练过程中，Model(s, j)是逐渐趋于完善的，Model(s, j)比较完善时，可以适当的加大规划步数 λ；反之，减小 λ。Model(s, j)的完善程度可以用训练过程中的时间步数波动来表示，时间步数波动越大，代表模型越不完善；时间步数波动越小，代表模型越完善。时间步数波动又可以简单的用前后回合之间的时间步数差来表示；因此，笔者将自适应的规划步数表示为：

$$\lambda_l=\left\lfloor \frac{32}{\omega_c^{\,2}(\mathrm{num}_{\mathrm{st}}(l-1)-\mathrm{num}_{\mathrm{st}}(l-2))^2+1}+0.5 \right\rfloor 。 \quad (3)$$

其中：ω_c 为可调参数，取值范围 0～8，调节该参数，使方法具有更普遍的适用性；$\mathrm{num}_{\mathrm{st}}$ 为时间步数，其下标代表训练过程中的回合数；$\lfloor\ \rfloor$ 为向下取整。

在上述分析的基础上，构建基于规划步数自适应 Dyna-Q 的干扰决策算法如图 2 所示。

Initialize：
1. 设置学习率 α，折扣因子 γ，探索率 ε，收益函数 R
2. 设置 ω_c，设置最大回合数 $l_{\max}$；
3. 初始化 Q 表；
4. 初始化多功能雷达信号模型 Model(s, j)；
5. 初始化回合数 $l=1$
While $l<l_{\max}$:
6. 通过侦察感知获取雷达的初始状态 S_0，随机的初始化一种干扰样式 J_0，对雷达实施干扰；
7. 通过侦察感知得到雷达的状态 S_t；
8. 设置时间步 $t=1$；
While $S_t \neq S_{aim}$ do:
9. 采用 ε-greedy 方法选取干扰样式, 对雷达实施干扰；
10. 通过侦察感知获取下一时间步的雷达状态 S_{t+1}；
11. 更新多功能雷达信号模型 Model(s, j)；
12. 收益 $R_t=r(S_t,S_{t+1})$；
13. 根据公式 $Q(s,a)\leftarrow Q(s,a)+\alpha[r(s,a)+\gamma\max Q(s',a')-Q(s,a)]$ 更新 Q 值；
14. If $l<3$:
15. $\lambda_l=0$；
16. else:
17. $\lambda_l=\left\lfloor \frac{32}{\omega_c^{\,2}\cdot(\mathrm{num}_{st}(l-1)-\mathrm{num}_{st}(l-2))^2+1}+0.5 \right\rfloor$
18. End
19. 初始化 $i=0$
While $i<\lambda_l$:
20. 随机选择雷达状态 S_t^- 和干扰样式 J_t^-；
21. 通过模型 Model(s, j) 得到下一时间步的雷达状态 S_{t+1}^-；
22. 收益 $R_t=r(S_t^-,S_{t+1}^-)$；
23. 根据公式 $Q(s,a)\leftarrow Q(s,a)+\alpha[r(s,a)+\gamma\max Q(s',a')-Q(s,a)]$ 更新 Q 值；
24. $i=i+1$；
End while
18. $t=t+1$；
End while
25. $l=l+1$
26. $\mathrm{num}_{st}(l)=t$
End while

图 2　基于规划步数自适应的 Dyna-Q 的干扰决策算法

3　实验与仿真

为验证所提算法的性能，构建仿真环境进行实验。仿真环境中，多功能雷达的雷达状态数量为 50，分别为 s_1, s_2, …, s_{50}；智能干扰机可以产生的干扰样式包括 9 种，分别为 j_1, j_2, …, j_9；雷达状态转移概率由随机生成的转移矩阵决定；干扰机任务的目标是将当前雷达状态转移到目标雷达状态，当前雷达状态设置为 s_1，目标雷达状态设置为 s_{25}。从当前雷达状态转移到目标雷达状态需进行至少 7 次转换，即最佳策略下，时间步数为 7。

通过仿真实验，对比基于 Dyna-Q 的干扰决策算法和基于规划步数自适应 Dyna-Q 的干扰决策算法。2 种算法的算法公共参数包括学习率 α，折扣因子 γ，探索率 ε，分别设置为 0.01，0.95，0.1。基于 Dyna-Q 的干扰决策算法规划步数 λ 设置为 0，2，4，8，16，32；基于规划步数自适应 Dyna-Q 的干扰决策算法中，ω_c 设置为 0，0.001，0.005，0.01，0.05，0.1，0.3，0.5，0.7，1，3，5，8。每次改变算法参数进行 100 次蒙特卡洛实验。

实验的统计结果包括最优收敛率和收敛总步数 2 项。最优收敛率是在 100 次蒙特卡洛实验中，算法收敛到最佳干扰策略的百分比(利用最后收敛的时间步数是否为 7 来判断是否为最佳干扰策略)，最优收敛率反应了算法的有效性；收敛总步数是从对抗开始到算法收敛，所有回合的时间步数之和，收敛总步数反映了算法的收敛性，在本文中用 100 次蒙特卡洛实验的平均收敛总步数来分析算法的收敛性。2 种算法的实验统计结果分别如表 1 和 2。

表 1　基于 Dyna-Q 的干扰决策算法实验数据统计

λ	平均收敛总步数	最优收敛率	λ	平均收敛总步数	最优收敛率
0	11 611	1	8	3 966	0.97
2	7 468	1	16	2 648	0.89
4	5 960	1	32	1 591	0.82

在表 1 中，随着规划步数从 0 到 32 的逐渐增加，平均收敛总步数从 11 611 步下降到 1 519 步，算法的收敛性得到了提高；但同时，最优收敛率也从 1 下降到了 0.82，算法有效性出现了下降。这表明在基于 Dyna-Q 的干扰决策算法中，其算法收敛性和有效性存在着明显的矛盾关系，难以做到二者的兼顾。

在表 2 中，随着 ω_c 从 8 减小到 0，平均收敛总步数从 10 337 减小到 1 690，算法的收敛性能随 ω_c 的减小明显提升；而在此过程中，算法的最优收敛率有轻微的下降，但始终保持在 0.95 以上。基于规划步数自适应 Dyna-Q 干扰决策算法的收敛性能够在参数 ω_c 变化时，始终保持在一个较高的水平。

表 2 基于规划步数自适应 Dyna-Q 的干扰决策算法实验数据统计

ω_c	平均收敛总步数	最优收敛率	ω_c	平均收敛总步数	最优收敛率
0	1 690	0.96	0.500	7 287	0.98
0.001	1 695	0.96	0.700	7 872	0.98
0.005	2 028	0.97	1.000	8 590	0.96
0.010	2 185	0.95	3.000	10 237	1.00
0.050	3 142	0.95	5.000	10 465	1.00
0.100	3 955	0.98	8.000	10 337	0.99
0.300	5 982	0.99			

在表 2 中，随着 ω_c 从 8 减小到 0，平均收敛总步数从 10 337 减小到 1 690，算法的收敛性能随 ω_c 的减小明显提升；而在此过程中，算法的最优收敛率有轻微的下降，但始终保持在 0.95 以上。基于规划步数自适应 Dyna-Q 干扰决策算法的收敛性能够在参数 ω_c 变化时，始终保持在一个较高的水平。

为了便于对比算法的性能和算法应用中算法参数的选择，引入最优收敛率的概念。最优收敛率阈值即在一次多功能雷达干扰任务中，干扰任务所能容忍的最优收敛率的最小值，最优收敛率≥最优收敛阈值是可以接受的，最优收敛率＜最优收敛率阈值是无法接受的。在实际的雷达对抗中，选取算法参数的原则为在保证最优收敛率满足最优收敛率阈值的基础上，尽可能选择平均收敛总步数更小的算法参数。

若干扰任务的最优收敛率阈值为 1，有 λ=0, 2, 4 时的基于 Dyna-Q 的干扰决策算法和 ω_c=5, 3 时的基于规划步数自适应 Dyna-Q 算法满足这一要求，并且 ω_c=3 时的平均时间总步数最低，所以在该次雷达对抗任务中应选用基于规划步数自适应 Dyna-Q 的干扰决策算法，参数 ω_c 的值设置为 3。若干扰任务的最优收敛率阈值为 0.95，则 λ=0, 2, 4, 8 的基于 Dyna-Q 的干扰决策算法和所有基于规划步数自适应 Dyna-Q 的干扰决策算法都满足该条件，通过对比平均收敛总步数，在该次雷达干扰任务中应选用基于规划步数自适应 Dyna-Q 的干扰决策算法，参数 ω_c 的值设置为 0。同理，当干扰任务的最优收敛率阈值为 0.9, 0.85 时，也应选用基于规划步数自适应 Dyna-Q 的干扰决策算法。通过在相同的最优收敛率阈值下对比 2 种算法，可得出基于规划步数自适应 Dyna-Q 的干扰决策算法在绝大多数情况下，算法性能都优于基于 Dyna-Q 的干扰决策算法。通过对规划步数采取自适应的机制，在提高算法收敛性能的同时，也保证了算法的有效性。

4 结束语

通过分析仿真实验结果可知：笔者提出的基于规划步数自适应 Dyna-Q 的干扰决策方法不仅收敛速度快，而且还能在保持高收敛速度的同时保留极高的有效性，可以实现多功能雷达干扰决策的实时、有效。此外，该方法也可以扩展到强化学习的其他应用领域，在不损失算法实用性能的前提下，用于提升强化学习算法的收敛性能。

参考文献：

[1] 邢强, 朱卫纲, 贾鑫. 干扰规则库未知条件下的干扰决策[J]. 系统工程与电子技术, 2019, 41(2): 298-303.

[2] 赖中安, 周刚峰. 矩阵博弈应用于雷达有源干扰策略选择的研究[J]. 航天电子对抗, 2010, 26(5): 16-18, 53.

[3] LI K, JIU B, LIU H. Game theoretic strategies design for monostatic radar and jammer based on mutual information[J]. IEEE Access, 2019, 7: 72257-72266.

[4] SONG X, WILLETT P, ZHOU S, et al. The power game between a MIMO radar and jammer: proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), F, 2012[C]. IEEE, 2012.

[5] 周脉成. 基于博弈论的雷达干扰决策技术研究[D]. 西安: 西安电子科技大学, 2014.

[6] 孙宏伟, 童宁宁, 孙富君. 基于D-S证据理论的电子干扰模式选择[J]. 弹箭与制导学报, 2003(S2): 218-220.

[7] 李云杰, 朱云鹏, 高梅国. 基于 Q-学习算法的认知雷达对抗过程设计[J]. 北京理工大学学报, 2015, 35(11): 1194-1199.

[8] 邢强, 贾鑫, 朱卫纲. 基于 Q-学习的智能雷达对抗[J]. 系统工程与电子技术, 2018, 40(5): 1031-1035.

[9] XING Q, ZHU W G, JIA X. Research on method of intelligent radar confrontation based on reinforcement learning[C]// 2017 2nd IEEE International Conference on Computational Intelligence and Applications(ICCIA). IEEE, 2017.

[10] XING Q, ZHU W G, JIA X. Intelligent countermeasure design of radar working-modes unknown: 2017 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)[C]. IEEE, 2018.

[11] MIRANDA S, BAKER C, WOODBRIDGE K, et al. Comparison of scheduling algorithms for multifunction radar[J]. 2007, 1(6): 414-24.

[12] ORMAN A, POTTS C N, SHAHANI A, et al. Scheduling for a multifunction phased array radar system[J]. 1996, 90(1): 13-25.

[13] NGUYEN N H, DOĞANçAY K, DAVIS L M J S P. Adaptive waveform and Cartesian estimate selection for multistatic target tracking[J]. 2015, 111: 13-25.

[14] WATKINS C J, DAYAN P J M L. Q-learning[J]. Machine Learning, 1992, 8(3-4): 279-92.

[15] SUTTON R S, BARTO A G. Reinforcement learning: An introduction[M]. 2nd Edition, MIT press, 2018: 45-66.