

基于最大策略熵深度强化学习的 通信干扰资源分配方法

饶宁, 许华, 齐子森, 宋佰霖, 史蕴豪

(空军工程大学 信息与导航学院, 陕西 西安 710077)

摘要:针对通信组网对抗中干扰资源分配的优化问题,提出了一种基于最大策略熵深度强化学习(MPEDRL)的干扰资源分配方法。该方法将深度强化学习思想引入到通信对抗干扰资源分配领域,并通过加入最大策略熵准则且自适应调整熵系数,以增强策略探索性加速收敛至全局最优。该方法将干扰资源分配建模为马尔可夫决策过程,通过建立干扰策略网络输出分配方案,构建剪枝孪生结构的干扰效果评估网络完成方案效能评估,以策略熵最大化和累积干扰效能最大化为目标训练策略网络和评估网络,决策干扰资源最优分配方案。仿真结果表明,所提出的方法能有效解决组网对抗中的干扰资源分配问题,且相比于已有的深度强化学习方法具有学习速度更快,训练过程波动性更小等优点,干扰效能高出DDPG方法15%。

关键词:干扰资源分配;深度强化学习;最大策略熵;神经网络

中图分类号: TN975

文献标志码: A

文章编号: 1000-2758(2021)05-1077-10

随着各种电子信息技术在军事领域的广泛应用,电子对抗在现代战争中发挥的作用显得愈加重要。为了确保信息的安全传输,涌现出了如跳频、智能组网、猝发通信等各种抗干扰通信技术^[1-3]。在通信组网对抗背景下,干扰方的对抗目标已由单一链路变为通信网络,如何利用有限干扰资源对抗整个通信网络获得最优干扰效果,需要制定合理的资源分配方案来实现干扰资源利用效益最大化。而对抗通信网的干扰资源分配问题决策复杂度高,仅靠人工调度效率较低。当前,遗传算法、离散布谷鸟算法、模拟退火算法、人工蜂群算法等智能算法已被广泛用于解决这类如雷达辐射源干扰资源分配、认知无线电频谱资源利用等决策问题^[4-6]。对于非线性组合优化问题,上述算法都需要较完备的先验信息且需对数据分布作出假设,这些假设随着无线网络的复杂度提升与实际情况的差异会逐渐变大,并且在组网对抗中干扰方难以获得通信方的先验信息,此类算法实用性受限,不能很好地解决通信干扰资源分配问题。

强化学习作为人工智能领域的重要研究方向,可在无先验信息条件下求解决策问题。深度强化学习融合了深度学习的特征提取能力^[7],在强化学习框架中利用神经网络拟合目标函数来决策复杂高维空间的资源分配问题已成为研究热点,相关成果可分为以下2类:①基于单智能体深度强化学习的资源分配方法^[8-12],如文献[8-10]针对无线网络中的信道接入问题、功率分配等问题均采用基于深度Q网络(deep Q network, DQN)算法的分配方法来达到最大化频谱利用效率、最小化功耗等目的。但是DQN算法只适用于离散动作空间的场景,不适合动作空间过大的联合优化问题。为解决连续空间的决策问题,文献[11]提出基于深度确定性策略梯度(deep deterministic policy gradient, DDPG)的多用户无线蜂窝网络功率控制方法,并通过理论分析证明了DDPG算法可以应用于多种通信网络的用户调度、信道管理和功率分配等问题。此外,文献[12]提出在深度强化学习框架下构建资源分配模型,利用图卷积网络抽取底层关键的拓扑特征来学习最佳

资源分配策略;②基于多智能体强化学习的资源分配方法^[13-16],如针对认知无线网络中主基站和认知基站共存导致的聚集干扰问题,文献[13]提出了多智能体 Q 学习的信道和功率分配方法,将多个认知基站建模为多个智能体,以集中训练、分散执行的方式获得节能资源分配策略。文献[14]提出基于分布式近端策略优化的功率控制方法,设置多个智能体在多线程中与环境交互以提升学习速率。多智能体强化学习方法多用于智能体之间存在非合作博弈的场景,如文献[15]将车联网中的每条车辆与车辆(vehicle to vehicle, V2V)链路分别视为单智能体,各智能体在不具备全局网络信息情况下均利用 DDPG 算法来获得各自最优分配策略。文献[16]提出分布式多智能体的深度竞争双 Q 网络算法,各用户在随机博弈模型中达到纳什均衡,在满足各用户服务质量要求的同时最大化长期的整体网络效用。

现有研究大都面向认知无线电、雷达对抗等领域且多为非合作博弈场景,很少考虑通信对抗的协同干扰场景。本文针对对抗组网通信场景下的通信干扰资源分配问题,提出一种基于最大策略熵深度强化学习的干扰资源分配方法,通过将策略熵引入神经网络的策略梯度中,使得算法在期望最大化干扰策略效能的同时兼顾最大化策略熵,提升策略的探索性以更快地收敛至全局最优。通过仿真对比,本文所提算法相比于其他算法收敛速度更快,可更高效地完成资源分配。

1 对抗场景与决策模型构建

1.1 对抗场景

在无线通信环境中,假设干扰方有 N 台干扰机, $n = \{1, 2, \dots, N\}$ 表示干扰机的集合,干扰机采用瞄准式干扰模式。通信方采用 TCP/IP 协议通信,并使用 M 条通信链路进行组网通信, $m = \{1, 2, \dots, M\}$ 表示通信链路的集合,这些通信链路使用互不干扰且正交的等带宽信道,且各通信链路相对重要性指数可表示为

$$W = [\omega_1, \omega_2, \dots, \omega_M] \quad (1)$$

假设干扰方通过通信侦察并经过情报分析综合,掌握了敌方各通信链路所使用的中心频率,并确定了各通信链路的接收机所处位置,本文假设各接收机均为固定站。对于干扰方而言,通信方所使用的各通信链路相对重要性指数未知。干扰方期望在

资源受限条件下,合理分配干扰资源,获得最大干扰效能,对抗场景如图 1 所示。

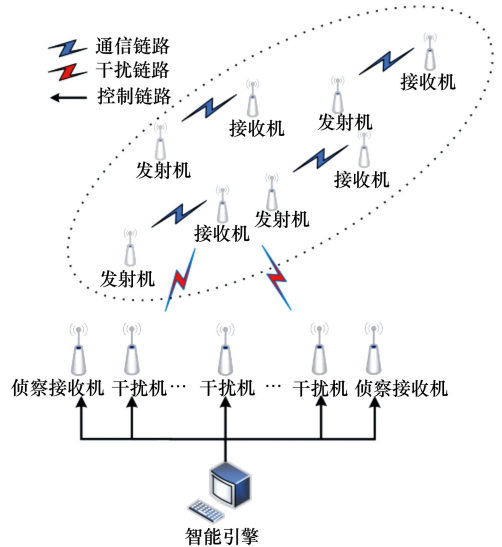


图 1 对抗场景

每台干扰机至多同时干扰 U 条通信链路, t 时刻设通信链路 i 的发射机信号功率为 P_i , 链路信道增益为 H_i , P_j 和 H_j 分别表示干扰机信号发射功率和干扰链路信道增益。由于一条链路可能受到多个干扰机的干扰,故通信链路 i 接收机处的信干比为

$$R_{SN_i}(t) = \frac{P_i(t)H_i(t)L_j}{\sum_{j=1}^k P_j(t)H_j(t)L_i + \sigma^2}, k \leq U \quad (2)$$

式中: k 表示同时对链路 i 施加干扰的干扰机数量; σ^2 表示环境噪声方差; L_i 和 L_j 分别表示通信信号和干扰信号的路径损耗,由自由空间传播损耗得

$$L = 32.5 + 20\lg(f) + 20\lg(r) \quad (3)$$

式中: f 为信号中心频率; r 为信号传播距离。

战场环境下无法准确获得通信方接收机处的信干比,难以直接对分配方案的干扰效果进行评估。而根据通信 TCP/IP 协议,干扰方在释放干扰信号后可通过对环境侦察获取确认帧/非确认帧信息(ACK/NACK),统计侦收到的 NACK 数据包可得到通信方传输信息的误包率(packet error rate, R_{PE}),进而根据下式计算出符号错误率(symbol error rate, R_{SE})^[17]

$$R_{SE} = 1 - (1 - R_{PE})^{1/H} \quad (4)$$

式中: H 是校验比特数。

可将组网通信中所有链路在 t 时刻的总符号错误率表示为

$$R_{SE_{total}} = \sum_{i=1}^M R_{SE_i}(t) \quad (5)$$

结合每条通信链路的相对重要性,干扰资源受限条件下的干扰资源分配问题就可转化为优化问题,如(6)式所示

$$\max \mathbf{W} \cdot R_{SE_{total}} = \max \sum_{i=1}^M \omega_i \cdot R_{SE_i}(t) \quad (6)$$

$$R_{SE_i}(t) \geq \tau_0, \forall i \in M \quad (7)$$

式中, τ_0 表示干扰方设定的最小阈值。(6) ~ (7) 式表示干扰方案需在使得每条通信链路误符号率都至少达到设置阈值 τ_0 的基础上最大化加权的总符号错误率。

约束条件如(8) 式所示

$$\text{s.t. C1: } \sum_{i=1}^M x_i^n \leq U, \forall n \in N$$

$$\text{C2: } x_i^n \in \{0, 1\}, \forall n \in N, \forall i \in M \quad (8)$$

$$\text{C3: } 0 \leq \sum_{i=1}^M P_i^n \leq P_{\max}, \forall n \in N$$

C1 和 C2 表示每个干扰机至多只能干扰 U 条通信链路,其中 x_i^n 是二进制指示变量, $x_i^n = 1$ 表示分配第 n 台干扰机干扰第 i 条通信链路; C3 表示每个干扰机可输出的总干扰功率有限,其中 P_i^n 表示第 n 台干扰机干扰第 i 条通信链路所分配的干扰信号功率。

1.2 决策模型构建

强化学习方法通过建立马尔科夫决策过程 (Markov decision process, MDP) 求解问题,本场景中干扰机执行当前状态的干扰方案后,环境会转移到新的状态,而新的状态只取决于当前状态和干扰方案,与过去状态和干扰方案无关。因此本文研究的干扰资源分配问题满足马尔科夫时间无后效性,可建模为马尔科夫决策过程,马尔科夫决策过程包含智能体 Agent、状态空间 \mathbf{S} 、动作空间 \mathbf{A} 、奖励函数 \mathbf{R} 和折现因子 γ 等元素。本文中 MDP 定义如下:

智能体 Agent: 干扰方通过智能引擎制定干扰方案,而智能引擎可指引侦察机进行侦察并引导各干扰机进行协同干扰,故智能引擎可视为 MDP 中的智能体。

状态空间 \mathbf{S} : 环境状态 $\mathbf{S}(t)$ 表示当前时刻干扰资源的分配分案和干扰方案的干扰效果, $\mathbf{S}(t)$ 是由干扰资源分配矩阵 $\mathbf{X}(t)$ 和干扰效果评估矩阵 $\mathbf{E}(t)$ 构成的 $(N+1)$ 行 M 列矩阵,即

$$\mathbf{S}(t) = \begin{bmatrix} \mathbf{X}(t) \\ \mathbf{E}(t) \end{bmatrix}^{(N+1) \times M} \quad (9)$$

其中干扰资源分配矩阵表示为

$$\mathbf{X}(t) = [x_1(t), x_2(t), \dots, x_N(t)]^T \quad (10)$$

式中: $x_i(t) = [c_{i1}(t) \ c_{i2}(t) \ \dots \ c_{iM}(t)]$, $1 \leq i \leq N$ 表示单个干扰机的干扰目标;元素 $c_{ij}(t) \in \{0, 1\}$, 当 $c_{ij}(t) = 1$ 表示第 i 个干扰机对第 j 条通信链路进行干扰,反之 $c_{ij}(t) = 0$ 则表示未干扰。

干扰效果评估矩阵表示为

$$\mathbf{E}(t) = [\tau_1(t) \ \tau_2(t) \ \dots \ \tau_M(t)] \quad (11)$$

式中, $\tau_j(t) \in \{0, 1\}$, $1 \leq j \leq M$ 。 $\tau_j(t) = 1$ 表示干扰方评估得出的第 j 条通信链路误符号率达到预设值即 $R_{SE_j}(t) \geq \tau_0$, 反之 $\tau_j(t) = 0$ 表示 $R_{SE_j}(t) < \tau_0$ 。

动作空间 \mathbf{A} : 每个干扰机在时刻 t 可至多选择干扰 U 条通信链路,并在对应信道上分别施加总功率不超过 P_{\max} 的干扰信号,故令干扰方的干扰策略即干扰动作为

$$\mathbf{A}(t) = [a_1(t) \ a_2(t) \ \dots \ a_N(t)]^T \quad (12)$$

式中, $a_i(t) = [p_{i1}(t) \ p_{i2}(t) \ \dots \ p_{iM}(t)]$, $1 \leq i \leq N$ 表示第 i 个干扰机的干扰资源分配情况,其中 $0 \leq p_{ij}(t) \leq P_{\max}$, $1 \leq j \leq M$ 。 $p_{ij}(t) = 0$ 表示第 i 个干扰机未干扰第 j 条链路,否则表示第 i 个干扰机干扰第 j 条链路且干扰信号功率为 $p_{ij}(t)$, 且满足

$$\sum_{j=1}^M p_{ij} \leq P_{\max}, \forall i \in N \quad (13)$$

$$\sum_{j=1}^M \text{sign}(p_{ij}) \leq U, \forall i \in N \quad (14)$$

式中, sign 为符号函数。

奖励函数 \mathbf{R} : 强化学习中奖励函数机制的作用是告诉智能体当前行为相对而言的优劣程度,故奖励函数可引导算法的优化方向。在通信对抗的干扰资源分配问题中,干扰方的目的是在达到期望符号错误率的前提下使得干扰功率尽可能小,避免功率过大而暴露干扰机位置,因此将奖赏函数定义为

$$R(s_t, a_t) = \sum_{i=1}^M w_i \cdot$$

$$[(1 + \text{sign}(R_{SE_i}(t) - \tau_0))/P_i(t)] \quad (15)$$

式中: w_i 为第 i 条通信链路的相对重要性系数; sign 为符号函数; $R_{SE_i}(t)$ 为第 i 条链路的符号错误率; τ_0 为设置的符号错误率门限值; $P_i(t)$ 为对第 i 条链路的总干扰功率。

干扰资源分配优化问题的目标是要最大化分配方案的干扰效能,在强化学习模型中即最大化干扰方一段时间内获得的累积奖励

$$R = \max E \left[\sum_{t=0}^{T-1} \gamma^t R(s_t, a_t) \right] \quad (16)$$

式中, $\gamma \in [0,1]$ 为折现因子。

2 基于最大策略熵深度强化学习的资源分配算法

本文在分配干扰机于干扰链路的同时还涉及对不同通信链路干扰功率的分配,此时资源分配问题是非凸的 NP-hard 问题^[18]。NP-hard 问题的主流解决思路是求其次优解,运算复杂度高,特别是当待决策变量处于连续区间时求解困难,本文采用最大熵深度强化学习思想解决该问题。

2.1 最大策略熵

深度强化学习作为不需要先验信息的机器学习方法,采用试错方式进行学习,即控制智能体不断与环境交互,在所处环境状态下根据当前学到的策略采取动作,采取的动作会改变环境状态,并根据环境给出的反馈修正策略。在感知-决策-反馈-修正的过程中,智能体不断学习并优化行动策略,最终可获得当前环境下较好的执行策略。

传统深度强化学习模型的训练目标为寻找最优策略 π^* 使得累积奖励期望最大,即

$$\pi^* = \arg \max_{\pi} E_{(s_t, a_t) \sim \rho_{\pi}} \left[\sum_t r(s_t, a_t) \right] \quad (17)$$

式中: ρ_{π} 为策略 π 形成的状态-动作轨迹分布; s_t, a_t 和 r 分别是第 t 步时的状态、动作和即时奖励; E 表示数学期望运算。

在递归求解最佳策略 π^* 时采用的 Q 函数贝尔曼迭代公式为

$$Q(s_t, a_t) = r_t + \gamma E_{s_{t+1}} Q(s_{t+1}, a_{t+1}) \quad (18)$$

式中, s_{t+1}, a_{t+1} 是环境状态转移之后的状态和动作, γ 是折现因子。

文献[19]首次提出策略熵的概念,策略熵即策略分布熵,当策略熵较大时意味着策略的随机性较强,在未知环境中的探索能力较强,而足够的探索可实现对环境模型的充分学习避免陷入局部最优。

在深度强化学习模型中加入策略熵后,目标函数变为

$$\pi^* = \arg \max_{\pi} E_{(s_t, a_t) \sim \rho_{\pi}} \left[\sum_t r(s_t, a_t) + \alpha H(\pi(\cdot | s_t)) \right] \quad (19)$$

式中, $\sum_t r(s_t, a_t)$ 表示某时段内的累积奖励,在最佳干扰资源分配问题中,即表示累积干扰效能。 H 为状态 s_t 下策略分布的熵, α 为熵系数,且

$$H(\pi(\cdot | s_t)) = - \log(\pi_{\varphi}(a_{t+1} | s_{t+1})) \quad (20)$$

(9)式表示学习最佳策略过程中不仅要最大化累积奖励期望,还要最大化策略熵。

故可将(18)式写为

$$Q(s_t, a_t) = r_t + \gamma E_{s_{t+1}} (Q(s_{t+1}, a_{t+1}) - \alpha \log(\pi_{\varphi}(a_{t+1} | s_{t+1}))) \quad (21)$$

式中, π_{φ} 为从分布 Φ 采样出的策略。

文献[20]证明了策略分布与玻尔兹曼能量分布有相同的形式即正比于 Q 函数的指数形式,可通过 Kullback-Leibler (KL) 散度约束来更新策略

$$\pi_{\text{new}} = \operatorname{argmin}_{D_{\text{KL}}} \cdot \left(\pi_{\varphi}(\cdot | s_t) \iint \frac{\exp\left(\frac{1}{\alpha} Q^{\text{old}}(s_t, \cdot)\right)}{Z^{\text{old}}(s_t)} \right) \quad (22)$$

式中, $D_{\text{KL}}(\cdot)$ 表示 KL 散度约束; $Q^{\text{old}}(s_t, \cdot)$ 表示原策略下的 Q 函数; $Z^{\text{old}}(s_t)$ 表示原策略的对数配分函数。

2.2 算法框架

为提升模型在高维决策空间的泛化能力,采用深度神经网络表示 Q 函数和策略函数即评估网络和策略网络,核心思想是利用策略网络输出于干扰方案,利用评估网络对干扰方案优劣程度进行评判,并在价值误差函数中加入策略熵项,通过梯度下降方法优化策略网络和评估网络,当误差函数收敛后策略网络输出的干扰方案即为最终资源分配方案。基于最大策略熵深度强化学习的资源分配算法基本框架如图 2 所示。

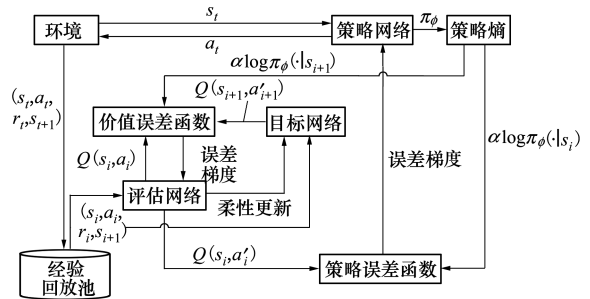


图 2 基于最大熵深度强化学习的资源分配方法基本框架

在图 2 中, π_{φ} 为策略网络输出的策略, a'_t 为在状态 s_t 根据策略 π_{φ} 采样出的动作, a'_{t+1} 为在状态 s_{t+1} 根据策略 π_{φ} 采样出的动作, $Q(s_t, a'_t)$ 是在状态 s_t 选择 a'_t 的价值, $\alpha \log \pi_{\varphi}(\cdot | s_t)$ 为在状态 s_t 时策略 π_{φ} 的熵。经验回放池存储过去交互得到的经验样本,可在训练阶段从回放池中采样样本用于训练神

经网络。

借鉴 DQN 算法中设置目标网络提升网络训练稳定性,本文算法亦采用了与评估网络结构完全相同的目标网络^[21],用目标网络的输出与即时奖励 r 之和作为评估网络训练的标签。

此外,为了解决 Q 函数对 Q 值过高估计会使学到的策略偏差较大,本文算法中评估网络和目标网络均采用剪枝孪生网络结构^[22]即设置 2 个相同结构的神经网络分别表示 Q 函数,2 个网络输入完全相同,每次将孪生网络中输出较小的 Q 值输入至价值误差函数中,如(23)式所示

$$y = r(s_t, a_t) + \gamma \min_{i=1,2} Q_{\theta_i}(s_{t+1}, a_{t+1}) \quad (23)$$

定义 Q 函数的价值误差为

$$J_Q(\theta_m) = E_{(s_t, a_t, s_{t+1}) \sim D, a_{t+1} \sim \pi_{\varphi}^d} \cdot \left[\frac{1}{2} \left(Q_{\theta_m}(s_t, a_t) - \left(r(s_t, a_t) + \gamma Q_{\theta_{\min}}(s_{t+1}, a_{t+1}) \right) \right)^2 \right] \quad (24)$$

for $m = 1, 2$
式中: D 为经验回放池, θ_m 为评估网络参数, φ 为策略网络参数, $\bar{\theta}_m$ 为目标网络参数, $\bar{\theta}_{\min}$ 为孪生目标网络中 Q 值较小网络对应的参数。

使用梯度下降更新评估网络参数 θ_m

$$\hat{\nabla}_{\theta} J_Q(\theta_m) = \nabla_{\theta} Q_{\theta}(s_t, a_t) \cdot \left[\left(Q_{\theta}(s_t, a_t) - \left(r(s_t, a_t) + \gamma Q_{\bar{\theta}_{\min}}(s_{t+1}, a_{t+1}) \right) \right)^2 \right] \quad (25)$$

式中, ∇ 为梯度算子。

在更新目标网络参数时,为减小波动性本文借鉴 DDPG 算法中^[23]的柔性更新方式更新目标网络参数 $\bar{\theta}_m$

$$J_{\pi}(\varphi) = E_{s_t \sim D, a_t \sim \pi_{\varphi}} [\alpha \log(\pi_{\varphi}(a_t | s_t)) - Q_{\theta}(s_t, a_t)] \quad (26)$$

由于从策略分布采样得出动作的过程无法进行链式求导,为计算策略梯度使用自编码器中重参数化技巧^[24],如图 3 所示。

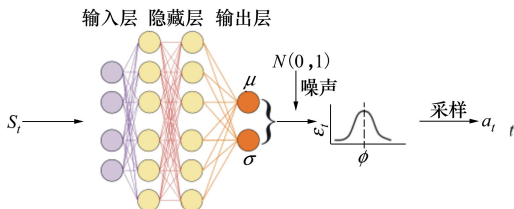


图 3 策略网络重参数化

图 3 中,不直接从均值和协方差构成的高斯分布中采样,而是先从标准正态分布里采样出噪声,然后把噪声值乘以策略网络输出的协方差再加上均值即可反向求导。动作 a_t 可表示为

$$a_t = f_{\varphi}^{\mu}(s_t) + \varepsilon_t \odot f_{\varphi}^{\sigma}(s_t) \quad (27)$$

式中: f_{φ}^{μ} 是策略网络输出策略分布的均值; f_{φ}^{σ} 是输出策略分布的方差; ε_t 是从标准正态分布中采样出的噪声值。

重参数之后,便可对策略网络进行反向传播和梯度下降更新

$$\hat{\nabla}_{\phi} J_{\pi}(\phi) = \nabla_{\phi} \alpha \log(\pi_{\phi}(a_t | s_t)) + \left[\begin{array}{l} \nabla_{a_t} \alpha \log(\pi_{\phi}(a_t | s_t)) \\ - \nabla_{a_t} Q_{\theta_{\min}}(s_t, a_t) \end{array} \right] \nabla_{\phi} f_{\phi}(o_t, s_t) \quad (28)$$

为了有效平衡在未知环境中的探索和利用,本文中熵系数 α 在学习过程可自适应更新,在初始阶段由于对环境模型不够了解,可调小熵系数增加策略的探索性以避免陷入局部最优;在经验积累到一定阶段,对学到的策略有足够信心时,可调大熵系数,增加对当前所学知识的利用程度。本文通过计算(29)式梯度并反向传播,可在不同策略熵状态时自适应更新熵系数

$$J(\alpha) = E_{a_t \sim \pi_{\varphi}} [-\alpha \log \pi_{\varphi}(a_t | s_t) - \alpha H'] \quad (29)$$

式中, H' 设置为动作的维度大小。

2.3 算法流程

结合建立的马尔科夫决策过程模型,提出基于最大策略熵深度强化学习的干扰资源分配方法如下。

算法 基于最大策略熵深度强化学习的干扰资源分配方法

步骤 1 建立干扰策略网络 π , 网络参数为 φ ; 建立干扰方案效果评估孪生网络 Q_1 和 Q_2 , 网络参数分别为 θ_1 和 θ_2 , 随机初始化上述网络参数;

步骤 2 建立干扰方案效果评估目标孪生网络 \bar{Q}_1 和 \bar{Q}_2 , 网络参数分别 $\bar{\theta}_1$ 和 $\bar{\theta}_2$, 对目标网络参数进行赋值: $\bar{\theta}_1 \leftarrow \theta_1, \bar{\theta}_2 \leftarrow \theta_2$;

步骤 3 设置经验回放池 D ;

步骤 4 While 连续 x 轮训练的奖励平均值变化幅度小于 δ , 执行:

for 每一时隙 t : 根据环境状态 s_t , 对干扰策略网络输出的策略分布进行采样, 得到干扰方案 $a_t \sim \pi_{\varphi}(a_t | s_t)$;

在环境中执行干扰方案 a_t , 得到下一时隙的环

境状态 s_{i+1} , 并计算得到环境奖励值 $r(a_i, s_i)$;

将状态转移, 干扰方案及奖励值存入经验回放池 D 中:

$$D \leftarrow D \cup \{(s_i, a_i, r(s_i, a_i), s_{i+1})\}$$

end for

for 每一次训练:

从经验回放池中采样小批次样本:

$$B = \{\dots, (s_i, a_i, r(s_i, a_i), s_{i+1}), \dots\} \text{ Length} = \text{batch_size}$$

计算干扰方案目标价值:

$$y(r(a_i, s_i), s_{i+1}) = r(a_i, s_i) +$$

$$\gamma [\min_{j=1,2} \bar{Q}_j(s_i, \bar{a}') - \alpha \log \pi_\varphi(\bar{a}' | s_{i+1})], \bar{a}' \sim \pi_\varphi(\cdot | s_{i+1})$$

利用梯度下降更新干扰方案价值评估网络参数 θ_1 和 θ_2 :

$$\nabla_{\theta_j} \frac{1}{|B|} \sum_{(s_i, a_i, r_i, s_{i+1}) \in B} (Q_{\theta_j}(s_i, a_i) - y(r(a_i, s_i), s_{i+1}))^2, \text{ for } j = 1, 2$$

$$\theta_j \leftarrow \theta_j - \nabla_{\theta_j} J_Q(\theta_j), \text{ for } j = 1, 2$$

利用梯度下降更新干扰策略网络参数 φ :

$$\nabla_\varphi \frac{1}{|B|} \sum_{s_i \in B} (\min_{j=1,2} Q_{\theta_j}(s_i, \bar{a}_\varphi(s_i)) - \alpha \log \pi_\varphi(\bar{a}_\varphi(s_i) | s_i))^2, \bar{a}_\varphi(s_i) \sim \pi_\varphi(\cdot | s_i)$$

$$\varphi \leftarrow \varphi - \nabla_\varphi J_\pi(\varphi)$$

利用梯度下降更新温度熵系数 α :

$$\alpha \leftarrow \alpha - \nabla_\alpha J(\alpha)$$

采用柔性更新方式更新干扰方案价值目标网络参数 $\bar{\theta}_1$ 和 $\bar{\theta}_2$:

$$\bar{\theta}_j \leftarrow \tau \cdot \bar{\theta}_j + (1 - \tau) \cdot \theta_j, \text{ for } j = 1, 2$$

end for

end while

算法流程图如图 4 所示。

为使输出动作连续且限制在规定范围内, 神经网络的激活函数采用 tanh 函数, 输出动作可表示为

$$a'_i = \tanh(f_\varphi^\mu(s_i) + \varepsilon_i \odot f_\varphi^\sigma(s_i)) \quad (30)$$

为抵消 tanh 函数对原高斯分布的影响, 需对原策略分布进行修正

$$\log \pi'(a' | s) = \log \pi(a | s) - \sum_{i=1}^D \log(1 - \tanh^2(a_i)) \quad (31)$$

式中: $\pi'(a' | s)$ 为修正后的策略分布; a_i 为经验回放池 D 中存放的第 i 个动作。

tanh 函数输出范围为 $[-1, 1]$, 将输出的动作值进行线性映射之后即可投影至真实的干扰功率

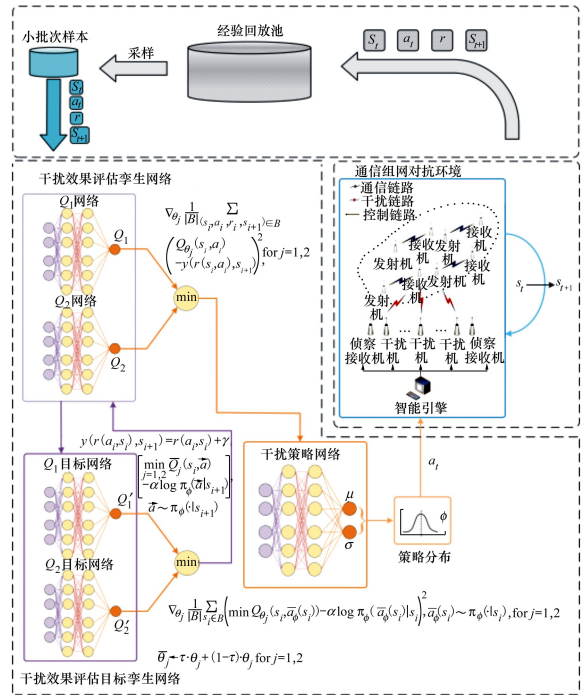


图 4 基于最大熵深度强化学习的干扰资源分配方法框图

范围。

3 仿真与分析

假设通信方使用 8 条通信链路进行组网通信, 各通信链路的相对重要性指数为 $W = [0.522\ 8, 0.295\ 2, 0.419\ 9, 0.673\ 4, 0.526\ 7, 0.697\ 0, 0.570\ 6, 0.517\ 4]$ 。干扰方有 5 台干扰机, 每台干扰机可至多同时干扰 2 条通信链路, 干扰机部署位置距离通信方 300 km, 其他实验及模型参数如表 1 所示。

表 1 实验及模型参数

| 参数 | 取值 |
|---------------------------|--------------------|
| 通信发射机发射功率 P_t /dBm | 55 |
| 信道带宽 B /kHz | 180 |
| 中心频率 f /MHz | 225 |
| 干扰机最大发射功率 P_{\max} /dBm | 70 |
| 环境噪声方差 σ^2 /mW | 10^{-11} |
| 误符号率门限值 τ_0 | 0.05 |
| 通信链路增益 H_i /dB | 5 |
| 干扰信号传播距离 r_j /km | 300 |
| 各链路通信距离 r_i /km | 60, 110, 120, 100, |
| | 90, 85, 75, 80 |
| 干扰链路增益 H_j /dB | 8 |

续表 1

| 参数 | 取值 |
|------------------|-------|
| 柔性更新系数 τ | 0.005 |
| 训练提前终止的阈值 x | 10 |
| 变化幅度 $\delta/\%$ | 5 |
| 每回合交互次数 T | 1 000 |
| 经验回放池容量 D | 106 |
| 批次样本大小 B | 256 |
| 折现因子 γ | [0,1] |
| 熵系数初始值 α | 1 |
| 训练回合数 E | 500 |

本文算法在资源分配过程中构建了策略网络、评估网络和目标网络,各个网络输入输出相互关联,神经网络的性能优劣直接影响算法实用性,而网络性能取决于网络的超参数,如隐藏层结构、优化器等,不同问题的最佳超参数配置一般不同且无法事先获得,加之通过理论方法分析不同参数深度强化学习算法的收敛性较为困难。本文参考文献[16]采用的仿真分析调参方式,此处给出精调后的参数及神经网络结构配置:本文算法选定 2 层隐藏层,神经元数为(256,64)的全连接网络,在上述网络结构基础上采用 Adam 优化器,并选择折现因子为 0.1。

首先分析熵系数对本文算法寻优性能的影响,之后在相同实验环境中将本文算法与基于 DQN^[8]和基于 DDPG^[11]的资源分配方法进行比较。每次实验采用蒙特卡洛方法重复执行 5 000 次,对实验结果取平均值。

图 5a)中,熵系数可随策略优化而自适应变化时,熵系数最终下降至 0 表明已不考虑策略熵的影响,转为充分利用已学到的环境信息。从图 5b)可知此时算法收敛速度更快,干扰效能在 530 回合左右即可收敛至稳定值。而当熵系数固定不变时,由于熵的存在使算法始终保持一定的随机性,干扰效能在训练 1 000 个回合仍不能完全收敛,熵系数自适应变化时获得的总效能提高了 7%。

在相同实验条件下利用 MPEDRL、DDPG、DQN 等算法解决干扰资源分配问题,分别从每回合压制干扰成功率、价值误差函数收敛速度以及获得的干扰总效能等方面进行对比。本文将压制干扰定义如下:当通信网络中所有通信链路的误符号率均高于误符号率门限值时,认为实现了对组网通信的压制干扰。

DQN 算法无法解决连续变化动作的控制问题,

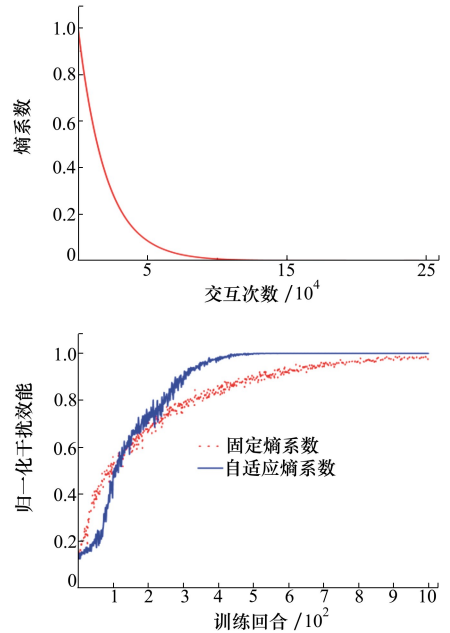


图 5 熵系数对算法性能影响

在本实验中需要将连续变量如干扰功率进行离散化,此处将干扰功率等间隔划分成 $|A|$ 个等级。

图 6a)是 $|A| = 30$ 时每回合压制干扰成功率对比。可以看出 DQN、DDPG 算法在 500 个训练回合内最高压制干扰成功率不超过 85%,而 MPEDRL 算法最终可实现单回合近 95%的压制干扰成功率。

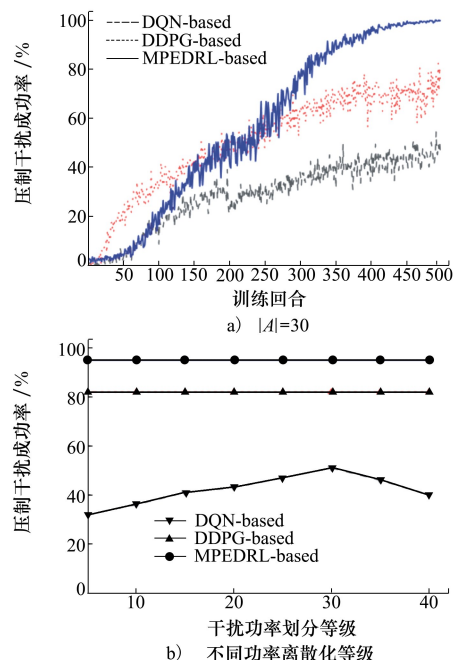


图 6 压制干扰成功率对比

图 6b) 是不同功率划分等级下压制干扰成功率对比。当功率划分等级从 5 增加至 30 时, DQN 算法干扰成功率也在提升。然而进一步增加输出维度并不能改善该算法性能, 当功率划分等级超过 30 时 DQN 算法的成功率慢慢下降至 40%, 这说明巨大的动作空间会导致实际训练比较困难。通过简单地扩大动作空间, 也无法完全消除量化误差。而 DDPG 和 MPEDRL 算法无需离散化动作空间, 性能优于 DQN 算法。DDPG 算法虽适用于连续的动作空间, 但采用确定性策略, 对未知环境的探索不足, 压制干扰成功率低于 MPEDRL 算法。

图 7 是训练过程中的各算法价值误差对比, 对比曲线变化, DDPG 算法价值误差下降最快, 在 50 个训练回合之后误差即可降至 0.1, 但仍存在一定波动性。MPEDRL 算法开始时由于输出的策略随机

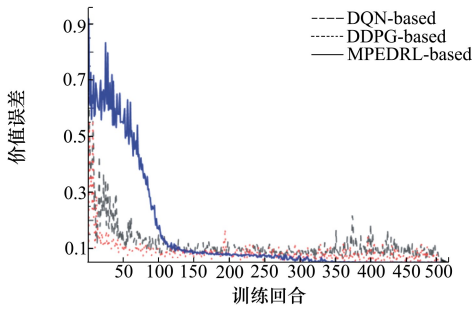


图 7 价值误差曲线对比

性较强, 波动性也较大, 但能迅速收敛, 在 350 回合之后价值误差已下降接近于 0。

图 8 是各资源分配算法的归一化干扰总效能对比。可以看到, 基于 DQN 和基于 DDPG 的资源分配

方法初始学习速度较快, 但训练过程波动性相对较大, 而基于 MPEDRL 的资源分配方法在初始训练阶段对环境的探索性较强, 收敛速度较慢, 但通过充分利用所学知识, 收敛速度迅速提升。图 8 中, MPEDRL 算法在 280 回合之后总干扰效能逐渐超过其他算法, 最后趋于稳定, 最终干扰效能高出 DDPG 算法 15%。

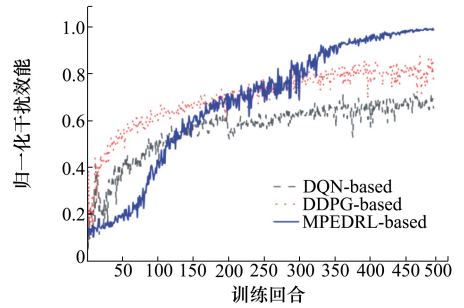


图 8 归一化干扰效能曲线对比

4 结 论

针对通信组网对抗中的干扰资源分配问题, 本文基于最大策略熵深度强化学习提出了一种新的干扰资源分配方法。该方法不需要过多有关通信方的先验信息, 在深度强化学习框架中将干扰方作为智能体, 通过在目标函数中加入策略熵使得智能体在追求获得最大干扰效能的同时期望最大化干扰策略熵, 可获得在未知环境中获得探索和利用的较好平衡, 避免陷入局部最优解。仿真结果表明, 本文算法能够在与外部环境不断交互的过程中学习到高效的干扰资源分配策略, 相较于已有方法收敛速度更快, 学习过程波动性小。

参考文献:

[1] BAO J J, JI L J. Frequency hopping sequences with optimal partial hamming correlation[J]. IEEE Trans on Information Theory, 2016, 62(6): 3768-3783

[2] WANG X J, LEI M J, ZHAO M J, et al. Cooperative anti-jamming strategy and outage probability optimization for multi-hop ad-hoc networks[C]//2017 IEEE 86th Vehicular Technology Conference, 2017: 24-27

[3] SUN J, LI X. Carrier frequency offset synchronization algorithm for short burst communication system[C]//Proceedings of 2016 IEEE 13th International Conference on Signal Processing, 2016: 6-10

[4] 李东生, 高杨, 雍爱霞. 基于改进离散布谷鸟算法的干扰资源分配研究[J]. 电子与信息学报, 2016, 38(4): 899-905
LI Dongsheng, GAO Yang, Yong Aixia. Jamming resource allocation via improved discrete cuckoo search algorithm[J]. Journal of Electronics & Information Technology, 2016, 38(4): 899-905 (in Chinese)

[5] 刘以安, 倪天权, 张秀辉, 等. 模拟退火算法在雷达干扰资源优化分配中的应用[J]. 系统工程与电子技术, 2009, 31(8): 1914-1917

- LIU Yian, NI Tianquan, ZHANG Xiuhui, et al. Application of simulated annealing algorithm in optimizing allocation of radar jamming resources[J]. *Systems Engineering and Electronics*, 2009, 31(8): 1914-1917 (in Chinese)
- [6] 袁建国, 南蜀崇, 张芳, 等. 基于人工蜂群算法的多用户 OFDM 自适应资源分配方案[J]. *吉林大学学报*, 2019, 49(2): 624-630
- YUAN Jianguo, NAN Shuchong, ZHANG Fang, et al. Adaptive resource allocation for multi-user OFDM based on bee colony algorithm[J]. *Journal of Jilin University*, 2019, 49(2): 624-630 (in Chinese)
- [7] LUONG N C, HOANG D T, GONG S, et al. Applications of deep reinforcement learning in communications and networking: a survey[J]. *IEEE Communications Surveys & Tutorials*, 2019, 21(4): 3133-3174
- [8] WANG S, LIU H, GOMES P H, et al. Deep reinforcement learning for dynamic multichannel access in wireless networks[J]. *IEEE Trans on Cognitive Communications and Networking*, 2018, 4(2): 257-265
- [9] XU Z, WANG Y, TANG J, et al. A Deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs[C]//2017 IEEE International Conference on Communications, 2017: 1-6
- [10] 廖晓闽, 严少虎, 石嘉, 等. 基于深度强化学习的蜂窝网资源分配算法[J]. *通信学报*, 2019, 40(2): 11-18
- LIAO Xiaomin, YAN Shaohu, SHI Jia, et al. Deep Reinforcement learning based resource allocation algorithms in cellular networks[J]. *Journal on Communications*, 2019, 40(2): 11-18 (in Chinese)
- [11] FAN Meng, CHEN Peng, WU Lenan, et al. Power allocation in multi-user cellular networks: deep reinforcement learning approaches[J]. *IEEE Trans on Wireless Communications*, 2020, 19(10): 6255-6267
- [12] ZHAO D, QIN H, SONG B, et al. A graph convolutional network-based deep reinforcement learning approach for resource allocation in a cognitive radio network[J]. *Sensors*, 2020, 20(18): 5216-5239
- [13] KAUR A, KUMAR K. Energy-efficient resource allocation in cognitive radio networks under cooperative multi-agent model-free reinforcement learning schemes[J]. *IEEE Trans on Network and Service Management*, 2020, 17(3): 1337-1348
- [14] ZHANG H, YANG N, LONG K, et al. Power control based on deep reinforcement learning for spectrum sharing[J]. *IEEE Trans on Wireless Communications*, 2020, 19(6): 4209-4219
- [15] XU Y, YANG C, HUA M, et al. Deep deterministic policy gradient(DDPG)-based resource allocation scheme for NOMA vehicular communications[J]. *IEEE Access*, 2020, 8: 18797-18807
- [16] ZHAO N, LIANG Y, NIYATO D, et al. Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks[J]. *IEEE Trans on Wireless Communications*, 2019, 18(11): 5141-5152
- [17] AMURU S, TEKIN C, SCHAAR M, et al. Jamming bandits-a novel learning method for optimal jamming[J]. *IEEE Trans on Wireless Communications*, 2016, 15(4): 2792-2808
- [18] LUO Z, ZHANG S. Dynamic spectrum management: complexity and duality[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2008, 2(1): 57-73
- [19] HAARNOJA T, TANG H, ABBEEL P, et al. Reinforcement learning with deep energy-based policies[C]//Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 2017: 1352-1361
- [20] HAARNOJA T, ZHOU A, ABBEEL P, et al. Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor[C]//Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 2018: 1861-1870
- [21] MNIHL V, KAVUKCUOGLU I K, SLIVER D, et al. Human-level control through deep reinforcement learning[J]. *Nature*, 2015, 518(7540): 529-540
- [22] FUJIMOTO S, HOOF H, MEGER M. Addressing function approximation error in actor-critic methods[C]//Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 2018: 1587-1596
- [23] LILLICRAP T, HUNT J, PRITZEL A, et al. Continuous control with deep reinforcement learning[C]//Proceedings of the 32th International Conference on Machine Learning, Lille, France, 2015: 2361-2369
- [24] DURK K, SALIMANS T, WELLING M. Variational dropout and the local reparameterization trick[C]//Advances in Neural Information Processing Systems, Montreal, Canada, 2015: 2575-2583

Allocation method of communication interference resource based on deep reinforcement learning of maximum policy entropy

RAO Ning, XU Hua, QI Zisen, SONG Bailin, SHI Yunhao

(College of Information and Navigation, Air Force Engineering University, Xi'an 710077, China)

Abstract: In order to solve the optimization of the interference resource allocation in communication network countermeasures, an interference resource allocation method based on the maximum policy entropy deep reinforcement learning (MPEDRL) was proposed. The method introduced the idea of deep reinforcement learning into the communication countermeasures resource allocation, it could enhance the exploration of the policy and accelerate the convergence to the global optimum with adding the maximum policy entropy criterion and adaptively adjusting the entropy coefficient. The method modeled interference resource allocation as Markov decision process, then established the interference strategy network to output allocation scheme, constructing the interference effect evaluation network of the clipped twin structure for efficiency evaluation, and trained the policy network and the evaluation network with the goal of maximizing the strategy entropy and the cumulative interference efficacy, then decided the optimal interference resource allocation scheme. The simulation results show that the algorithm can effectively solve the resource allocation problem in communication network confrontation, comparing with the existing deep reinforcement learning methods, it has faster learning speed and less fluctuation in the training process, and achieved 15% higher jamming efficacy than DDPG-based method.

Keywords: interference resource allocation; deep reinforcement learning; maximum policy entropy; deep neural network

引用格式: 饶宁, 许华, 齐子森, 等. 基于最大策略熵深度强化学习的通信干扰资源分配方法[J]. 西北工业大学学报, 2021, 39(5): 1077-1086

RAO Ning, XU Hua, QI Zisen, et al. Allocation method of communication interference resource based on deep reinforcement learning of maximum policy entropy[J]. Journal of Northwestern Polytechnical University, 2021, 39(5): 1077-1086 (in Chinese)