arXiv:2304.02005v1 [cs.AI] 4 Apr 2023

风险感知分布式多智能体强化学习

Abdullah Al Maruf[1], Luyao Niu[1], Bhaskar Ramasubramanian[2],Andrew Clark[3],Radha Poovendran[1]

摘要——**自主网络系统和信息物理系统需要在未知环境中进行决策、学习和控制。这类决策可能受到多种因素影响，包括建模误差、成本变化以及概率分布尾部事件的影响。尽管多智能体强化学习（MARL）提供了一种框架，通过与环境反复交互并最小化平均成本来学习行为，但它不足以应对上述挑战。本文提出一种分布式多智能体强化学习方法，通过学习风险感知动作来解决未知环境中的决策问题。我们使用条件风险价值（CVaR）刻画待最小化的成本函数，并定义贝尔曼算子来刻画与给定状态-动作对相关联的价值函数。我们证明该算子满足压缩性质，并且会收敛到最优价值函数。随后，我们提出一种名为*CVaR QD*学习 的分布式多智能体强化学习算法，并证明各个智能体的价值函数会达成共识。我们指出了*CVaR QD*学习算法在实现过程中面临的若干挑战，并提出相应的解决方案。我们通过仿真评估*CVaR QD*学习 算法，并展示风险参数对达成共识时价值函数的影响。**

I. 引言

针对复杂网络与信息物理系统的目标满足性进行推理，通常需要求解一个序贯决策问题。运行环境表示为马尔可夫决策过程（MDP） [1],，系统中任意两个状态之间的转移都是基于决策者或智能体所采取动作的概率结果。在动态且不确定的环境中，强化学习 [2]和最优控制 [3] 框架已被用于求解序贯决策问题，即通过确定动作来最小化累积成本。风险中性决策方案通过最小化期望成本或平均成本来确定动作；此类技术已应用于机器人学、移动网络和博弈等领域 [4]–[10]。

尽管风险中性解在计算上易于处理，但研究表明，它们在刻画对代价变化、建模误差以及概率分布尾部效应的敏感性方面存在局限 [11]–[13]。为了通过学习风险感知 动作来解决序贯决策问题，我们采用条件风险价值（*CVaR*） [14]–[17]来刻画马尔可夫决策过程的目标函数。CVaR对应于在代价取值足够大这一事件条件下的平均代价，并且其应用具有充分的理论依据 [14], [18]。优化基于CVaR的代价，可确保动作对罕见但后果严重的结果保持敏感 [19]。然而，与 [18],不同，我们假设智能体的运行环境未知。随后，智能体通过与环境反复交互，学习如何最小化所揭示的代价。针对这种情形，我们提出一种基于CVaR的经典Q-learning算法变体 [2], ，并证明其收敛性。

当多个决策者共享同一环境时，每个智能体都会与环境及其他智能体交互。因此，从任一单个智能体的视角来看，智能体行为的演变已被证明具有非平稳性 [20]–[22]。关于在多智能体强化学习中融入风险敏感性的研究十分有限。最近， [23], [24] 的作者提出了一种利用条件风险价值在合作式多智能体强化学习中学习风险敏感策略的框架。上述研究提出的算法采用集中式训练与分散式执行（CTDE）范式 [25] 来学习行为。采用集中式训练与分散式执行的智能体在训练期间可以利用其他智能体的观测和动作信息来辅助自身学习，但在测试时必须独立做出决策 [26], [27]。

与上述研究不同，本文设计了一种分布式风险感知多智能体强化学习算法。我们的方法受到 QD-learning [28], 的启发：在每一步中，单个更新规则都会综合环境揭示的成本，以及来自图中相邻智能体的信息；该图描述了智能体之间的通信。我们证明了在智能体优化基于条件风险价值的代价时，其价值函数能够达成共识。实验还表明，随着智能体的风险感知程度提高，共识状态下的价值函数幅度也会增大（对应于更高的已产生代价）；当目标是最小化累积成本时，这一观察符合直觉。我们的具体贡献如下：

- 我们定义一个贝尔曼算子，用以刻画基于条件风险价值的状态-动作价值函数。
- 我们证明，贝尔曼算子是一个压缩映射，且该算子的不动点就是最优风险感知价值函数。
- 我们提出了一种名为*CVaR QD*学习的风险感知分布式多智能体强化学习算法，并证明基于条件风险价值的各智能体价值函数能够达成共识。

[1]网络安全实验室，电气与计算机工程系，华盛顿大学，美国华盛顿州西雅图市，邮编 98195。 {maruf3e, luyaoniu, rp3}@uw.edu[2]电气与计算机工程系，西华盛顿大学，美国华盛顿州贝灵厄姆市，邮编 98225。 ramasub@wwu.edu[3]电气与系统工程系，圣路易斯华盛顿大学，美国密苏里州圣路易斯市，邮编 63130。andrewclark@wustl.edu

- 我们开展了实验评估，以验证*CVaR QD*学习 算法，并表明随着智能体的风险感知程度提高，共识状态下的价值函数幅值也会增大。

本文其余部分安排如下：第二节介绍强化学习和风险感知决策所需的预备知识。第三节构造贝尔曼算子，并证明其为压缩映射。第四节介绍基于CVaR的QD学习及相关分析结果；第五节介绍*CVaR QD*学习算法，并说明如何克服算法实现中的挑战。第六节展示实验评估结果，第七节总结全文。

II. 设置与问题表述

本节介绍本文所考虑的马尔可夫博弈设置，并提供强化学习以及决策中所用风险准则的必要预备知识。随后，我们将对本文要解决的问题进行形式化表述。

A. 设置

我们考虑一个包含 N 个智能体的系统。智能体间通信由无向图 $\mathcal{G} = (\mathcal{V}, \mathcal{E})$描述，其中 $\mathcal{V} = \{1, \cdots, N\}$ 是顶点（或节点）集合，$\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ 是连接顶点对的边集合。图 $\mathcal{G}$ 中的节点和边分别对应智能体以及智能体之间的通信链路。我们假设 $\mathcal{G}$ 是简单的（任意两个顶点之间没有自环或多条边）且连通的（每对节点之间都存在一条路径）。智能体 n的邻居集合记为 $\mathcal{N}(n)$。该图可由 $N \times N$ 拉普拉斯矩阵 L 描述，其元素为 $L_{ij} = -1$ （若$(i, j) \in \mathcal{E}$），否则为零；而 $L_{ii} = |\mathcal{N}(i)|$ 等于节点 i的度数。由于 $\mathcal{G}$ 是连通的， L的特征值可按 $0 = \lambda_1(L) < \lambda_2(L) \leq \cdots \leq \lambda_N(L)$ [29]的顺序排列。

我们在此考虑的多智能体设定与 [28] 的设定类似。我们假设每个智能体都能完全观测系统状态。然而，每个智能体收到的成本/奖励都是局部的，远程控制器可能无法获取。例如，这种多智能体设定可以类似于建筑物中空间分布的温度传感器（智能体） [28]。远程控制器可以访问所有传感器读数，但无法得知建筑物不同房间的期望温度。控制器的一个可能目标是，最小化传感器测得的温度与其对应位置的期望温度之间的平均平方偏差。

在这种设置下，环境中智能体的行为可以用马尔可夫博弈 $M := (\mathcal{S}, \mathcal{A}, \{c_1, \cdots c_N\}, P, \gamma)$ 来描述，其中假设 $\mathcal{S}$ 和 $\mathcal{A}$ 是有限的状态空间和动作空间。当系统处于状态 s ，且控制器采取的动作是 a 时，智能体 n 会产生有界且确定性的局部成本 $c_n(s, a) \in [-C_{max}, C_{max}]$。我们强调，在我们的设定中，状态 $s \in \mathcal{S}$ 和 $a \in \mathcal{A}$ 是全局的（即所有智能体共享），而个体成本 $c_n(s, a)$ 则是各智能体的局部成本。

$P(s'|s, a)$ 给出在状态 $s \in \mathcal{S}$ 下采取动作 $a \in \mathcal{A}$ 后转移到状态 $s' \in \mathcal{S}$ 的概率，而 $\gamma \in [0, 1)$ 是折扣因子。不过，与 [28]，不同，我们假设成本是确定性的，以便推导理论结果。

M 的轨迹是由状态和动作交替组成的序列（s_0，a_0，s_1，a_1，.....）。截至时间 k 的历史，记作 $h_k \in H_k$，对应于截至时间 k 的轨迹，即（s_0，a_0，s_1，a_1，.....，s_k）。形式上，给定 $H_0 = S$，我们递归地将截至时间 $k \geq 1$ 的可能历史集合定义为 $H_k = H_{k-1} \times \mathcal{A} \times \mathcal{S}$。时间 k 的策略是一个映射 $\Pi_{H,\,k} \colon H_k \to A$，并且我们定义 $\Pi_H := \lim_{k\to\infty} \Pi_{H,\,k}$ 为所有依赖历史的策略的集合。当策略 $\mu(s_k)$ 仅依赖于当前状态 s_k 时，称其为马尔可夫策略。

令 $c(s_k, a_k) = \frac{1}{N}\sum_{n=1}^{N} c_n(s_k, a_k)$ 表示在时刻 k 观测到的所有智能体的平均成本。直到时刻 k 的折扣平均成本定义为 $C_{0,\,k}(s_t\ a_t) := \sum_{t=0}^{k} \gamma^t c \frac{1}{N}\sum_{t=0}^{k} \gamma^t \sum_{n=1}^{N} c_n\ s_t\ a_t$ (.) (.)。因此，无限时域上的折扣平均成本为 $C_{0,\infty} = \lim_{k\to\infty} C_{0,k}$。

在强化学习（RL）中，转移概率P 是未知的，智能体通过与环境 [2] 反复交互来获知代价。其目标是学习一种使期望累积折扣代价最小化的策略。一种广泛使用的策略学习方法是 Q-learning算法 [30]。利用状态-动作价值函数 $Q^{\pi}(s, a) := \mathbb{E}_{\pi}[C_{0,\infty}|s_0 = s, a_0 = a, \pi]$，Q-learning算法旨在找到与最优策略 π^* 相对应的最优价值 $Q^*(s, a)$，使得对于所有 $(s, a) \in \mathcal{S} \times \mathcal{A}$ 以及任意策略 π，均有 $Q^*(s, a) \leq Q^{\pi}(s, a)$。

B. QD-Learning

QD-learning 是 Q-learning 的多智能体分布式变体。当未知环境中的智能体行为还受到一个底层通信图的影响时，QD-learning $\mathcal{G}$首次在 [28]中提出。在 QD-Learning算法[28], 的时间步 k，每个智能体 n 为所有状态-动作对（s，a）$\in \mathcal{S} \times \mathcal{A}$维护一个状态-动作价值函数序列 $\{Q_{n,k}(s, a)\} \in \mathbb{R}^{|\mathcal{S}\times\mathcal{A}|}$。序列 $\{Q_{n,k}(s, a)\}$按照以下规则 [28]进行更新：

$$\begin{aligned} Q_{n,k+1}(s_k, a_k) &= (1 - \alpha_k) Q_{n,k}(s_k, a_k) \\ &\quad + \alpha_k \big(c_n(s_k, a_k) + \gamma \min_{a' \in \mathcal{A}} Q_{n,k}(s_{k+1}, a')\big) \\ &\quad + \beta_k \sum_{l \in \mathcal{N}(n)} \big(Q_{n,k}(s_k, a_k) - Q_{l,k}(s_k, a_k)\big), \end{aligned} \tag{1}$$

其中，权重序列 $\{\alpha_k\}$和 $\{\beta_k\}$ 定义如下

$$\alpha_k = \frac{a}{(k+1)^{\tau_1}}, \tag{2}$$

$$\beta_k = \frac{b}{(k+1)^{\tau_2}}, \tag{3}$$

其中 a 和 b 为正常数。方程（2）和（3）保证了方程（1）中创新项和共识项的激励具有持续性；即 $\sum_k \alpha_k = \infty$ 和 $\sum_k \beta_k = \infty$。序列 $\{\alpha_k\}$ 和 $\{\beta_k\}$ 进一步满足 $\sum_k \alpha_k^2 < \infty$、$\sum_k \beta_k^2 < \infty$ 和 $\frac{\beta_k}{\alpha_k} \to \infty$，其中 $k \to \infty$。方程（2）和（3）中的常数 a 和 b 也经过选择，便得

对于所有 k，$(I_N - \beta_k L - \alpha_k I_N)$ 均为半正定矩阵，其中L 是图 $\mathcal{G}$ 的拉普拉斯矩阵， I_N 表示 $\mathbb{R}^{N\times N}$中的单位矩阵。确保其半正定的一种方法是选择 a 和 b，使得 $a + Nb \leq 1$。关于权重序列的选择，详细讨论请参见 [28] 。当权重序列选择得当时， [28]的作者证明，序列 $\{Q_{n,k}(s, a)\}$ 对于所有智能体渐近达到共识，且共识时达到的值等于最优动作价值函数 $Q^*(s, a)$。

C. 条件风险价值（CVaR）

设 Z 是概率空间 $(\Omega\ \mathcal{F}, P)$ 上的有界随机变量，其累积分布为 $F(z) = P(Z \leq z)$。置信水平 $y \in (0, 1]$ 下的风险价值（VaR）为 $VaR_y(Z) := min\{z|F(z) \geq 1 - y\}$ [14]。置信水平为 y 时的条件风险价值（CVaR）定义为 $CVaR_y(Z) := \mathbb{E}[Z|Z \geq VaR_y(Z)]$ [14], , 表示在尾部分布的 y 分位数条件下 Z 的期望值。注意，当 $y = 1$ 时，$CVaR_y(Z) = \mathbb{E}(Z)$，并且当 $y \to 0$ 时， $CVaR_y(Z) \to max\{Z\}$ 。随机变量 Z 的 CVaR 可解释为在扰动分布 ξP 下 Z 的最坏情况期望，如 [18], [31] 的以下结果所示。

命题1 (CVaR对偶形式 [18], [31])。 令$\mathbb{E}_\xi[Z]$ 表示 Z和 $\mathcal{U}_{CVaR}\ y\ P = \{\xi \quad \xi\ \omega \in [0, \frac{1}{y}] \int_{\omega\in} \xi\ \omega\ P\ \omega\ d\omega = (,) :: 1\}$ $(),\Omega\,()()$的 ξ加权期望。则，

$$CVaR_y(Z) = \max_{\xi\in\mathcal{U}_{CVaR}(y,P)} \mathbb{E}_\xi[Z].$$

上述对偶表示与CVaR度量的一致性一起，用于以递归方式推导 $CVaR_y(Z)$的分解 [18], [32]。在我们的设定中，随机变量 Z 是一个成本序列。

命题2 (CVaR分解 [18])。 对于 $k \geq 0$，令 $Z = \left(Z_{k+1}、\ Z_{k+2}、\ \cdots\right)$ 表示从时刻 $k+1$ 起的成本序列。则策略 π 下的条件$CVaR$满足

$$CVaR_y(Z|h_k, \pi) = \max_{\xi}\ \mathbb{E}[\xi(s_{k+1})CVaR_{y\xi(s_{k+1})}(Z|h_{k+1}, \pi)|h_k, \pi],$$

其中 $\xi \in \mathcal{U}_{CVaR}(y, P(\cdot|s_t, a_t))$，且期望是相对于 s_{k+1}而言的。

D. 问题表述

本文的目标是以分布式方式学习适用于多智能体强化学习的风险感知策略。我们使用条件风险价值作为风险敏感性的度量，并旨在学习一种策略 $\pi \in \Pi_H$ ，使其能够最小化风险敏感的折扣成本。我们交替使用置信水平和风险参数 来指代参数$y \in (0, 1]$。解决这一问题的挑战在于，转移概率未知，而且在特定状态下执行动作所产生的成本，只有通过与环境反复交互才会向智能体揭示。我们将要解决的问题正式表述如下：

$$\min_{\pi\in\Pi_H} CVaR_y\big(C_{0,\infty}|s_0, \pi\big). \tag{4}$$

为了解决式（4）中给出的问题，我们将首先提出一种用于Q学习的基于CVaR的贝尔曼算子，并证明该算子存在一个与最优解相对应的不动点。随后，我们将设计一种基于CVaR的QD-learning算法，并证明各智能体的价值函数能够达成共识。通过仿真，我们还将验证该算法的收敛性。

III. 贝尔曼算子的构造

本节定义一个贝尔曼算子，用于计算基于CVaR的强化学习中的最优状态-动作价值函数。与其他Q学习变体（例如 [13], [30], [33], [34], ）所采用的技术类似，我们通过证明该算子满足压缩性质来确立其收敛性。与此尤其相关的是文献 [18]中提出的基于CVaR的动态规划结果，该结果证明了马尔可夫决策过程的状态价值函数的收敛性（其转移概率和成本结构均为先验已知）。

利用命题2，我们首先为状态-动作对 (s, a) 增加一个额外的“连续状态” $y \in \mathcal{Y} = (0, 1]$ ，它表示 CVaR 的置信水平。然后，对于给定的策略 π 和 CVaR置信水平 $y \in \mathcal{Y}$，我们将增广状态-动作价值函数定义为：

$$Q^\pi(s, a, y) := CVaR_y(C_{0,\infty}|s_0 = s, a_0 = a, \pi). \tag{5}$$

为了建立$Q^\pi(s, a, y)$ 的动态规划刻画，我们在增广状态-动作价值函数空间上定义一个贝尔曼算子。考虑$CVaR$贝尔曼算子$\mathbf{T}$： $\mathcal{S} \times \mathcal{A} \times \mathcal{Y} \to \mathcal{S} \times \mathcal{A} \times \mathcal{Y}$ 和 $\xi \in \mathcal{U}_{CVaR}(y, P(\cdot|s, a))$。接着，我们定义

$$\mathbf{T}[Q](s, a, y) := c(s, a) + \gamma \min_{a'\in\mathcal{A}} \Big[\max_{\xi} \sum_{s'\in\mathcal{S}} \xi(s') P(s'|s, a) Q(s', a', y\xi(s'))\Big]. \tag{6}$$

我们的第一个结果形式化说明，式（6）中定义的贝尔曼算子满足压缩性质。

引理1（压缩）。设 $Q^1(s, a, y)$ 和 $Q^2(s, a, y)$ 是式（5）中针对相同的 (s, a, y) 定义的两个增广状态-动作价值函数。则式（6）中定义的贝尔曼算子 T 在 sup-范数下是一个压缩映射。也就是说， $||T[Q^1](s, a, y) - T[Q^2](s, a, y)||_\infty \leq \gamma||Q^1(s, a, y) - Q^2(s, a, y)||_\infty$。

证明。 我们的证明采用了与 [18] 类似的论证。然而，与 [18], 不同的是，我们设定中的环境未知；因此需要围绕状态-动作价值函数 [2], 进行推理，而不是使用 [18] 中的状态价值函数。

由于对于所有 $\xi \in \mathcal{U}_{CVaR}(y, P(\cdot|s, a))$，$\xi(s')P(s'|s, a') \geq 0$，因此只要对所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$、$y \in \mathcal{Y}$，$Q^1(s, a, y) \leq Q^2(s, a, y)$，就有 $\mathrm{T}[Q^1](s, a, y) \leq \mathrm{T}[Q^2](s, a, y)$。此外，由于

对于所有 $\xi \in \mathcal{U}_{CVaR}(y, P(\cdot|s,a))$，$\sum_{s' \in \mathcal{S}} \xi(s') P(s'|s,a') = 1$，我们有 $\mathbf{T}[Q + c](s,a,y) = c\gamma + \mathbf{T}[Q](s,a,y)$，其中 c 是一个常数。因此，我们已经证明 $\mathbf{T}[Q](s,a,y)$ 具有单调性和常数平移性质。现在，根据上确界范数的定义，对于所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$、$y \in \mathcal{Y}$，我们有 $|Q^1(s,a,y) - Q^2(s,a,y)| \leq ||Q^1(s,a,y) - Q^2(s,a,y)||_\infty$，这等价于

$$Q^2(s,a,y) - ||Q^1(s,a,y) - Q^2(s,a,y)||_\infty \leq Q^1(s,a,y) \leq Q^2(s,a,y) + ||Q^1(s,a,y) - Q^2(s,a,y)||_\infty. \quad (7)$$

将贝尔曼算子应用于式（7）中的各项，并利用单调性和常数平移性质，对于 $c = \pm||Q^1(s,\ a,\ y) - Q^2(s,\ a,\ y)||_\infty$，对所有 $s \in \mathcal{S},\ a \in \mathcal{A},\ y \in \mathcal{Y}$，可得

$$\mathbf{T}[Q^2](s,a,y) - \gamma||Q^1(s,a,y) - Q^2(s,a,y)||_\infty \leq \mathbf{T}[Q]^1(s,a,y) \leq \mathbf{T}[Q]^2(s,a,y) + \gamma||Q^1(s,a,y) - Q^2(s,a,y)||_\infty. \quad (8)$$

这等价于 $|\mathbf{T}[Q^1](s,\ a,\ y) - \mathbf{T}[Q^2](s,\ a,\ y)| \leq \gamma||Q^1(s,\ a,\ y) - Q^2(s,\ a,\ y)||_\infty \ \forall s \in \mathcal{S},\ a \in \mathcal{A},\ y \in \mathcal{Y}$，由此完成证明。□

引理1使我们能够证明，反复应用 CVaR 贝尔曼算子 $\mathbf{T}$ 时，其将收敛到不动点 $Q^f(s,\ a,\ y)$。设 $Q^*(s,\ a,\ y)$ 为关于 CVaR置信水平 y 的最优状态-动作价值函数，即

$$Q^*(s,a,y) = \min_{\pi \in \Pi_H} CVaR_y(\mathcal{C}_{0,\infty}|x_0 = s, a_0 = a, \pi). \quad (9)$$

现在我们推导一个中间结果，以便证明 CVaR 贝尔曼算子不动点的最优性，即 $Q^f(s,\ a,\ y) = Q^*(s,\ a,\ y)$。为此，我们用 $\mathbf{T}^k$ 表示贝尔曼算子 $\mathbf{T}$ 应用 k 次，其中 $k \in \mathbb{N}$为某个值。

引理 2。设对于所有 $s \in \mathcal{S}, a \in \mathcal{A}, y \in \mathcal{Y}$，$Q_k(s,a,y) := T^k[Q_0](s,a,y)$，其中 $k \in \mathbb{N}$，且 Q_0 是对于所有 $s \in \mathcal{S}$，$a \in \mathcal{A}, y \in \mathcal{Y}$，$Q(s,a,y)$ 的任意初始值。那么 $Q_k(s,a,y) = \min_{\pi \in \Pi_k} CVaR_y(\mathcal{C}_{0,k-1} + \gamma^k Q_0|s_0 = s, a_0 = a, \pi)$。

证明。我们使用归纳法证明该结果。对于基础情形，即 $k = 1$，我们有

$$\begin{aligned}
&Q_1(s,a,y) = \mathbf{T}[Q_0](s,a,y) = c(s,a) + \gamma \min_{a' \in \mathcal{A}} \Big[\max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \sum_{s' \in \mathcal{S}} \xi(s') P(s'|s,a) Q_0(s', a', y\xi(s'))\Big] \\
&= c(s,a) + \min_{a' \in \mathcal{A}} \Big[\gamma \max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \mathbb{E}_{\xi(s')}[Q_0 |s_0 = s', a_0 = a', \pi]\Big] \\
&= \min_{a' \in \mathcal{A}} \Big[c(s,a) + \gamma \max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \mathbb{E}_{\xi(s')}[CVaR_{y\xi(s')}(Q_0|s', a')|s_0 = s, a_0 = a, \pi)]\Big] \\
&= \min_{a' \in \mathcal{A}} \Big[c(s,a) + \gamma CVaR_y(Q_0|s_0 = s, a_0 = a, \pi)\Big] \\
&= \min_{\pi \in \pi_1} CVaR_y(\mathcal{C}_{0,0} + \gamma Q_0|s_0 = s, a_0 = a, \pi).
\end{aligned}$$

由此，基础情形得证。现在假设该结果对 $k = i$ 成立。接下来，对于 $k = i + 1$，我们有

$$\begin{aligned}
&Q_{i+1}(s,a,y) = \mathbf{T}^{i+1}[Q_0](s,a,y) = \mathbf{T}[Q_i](s,a,y) \\
&= c(s,a) + \gamma \min_{a' \in \mathcal{A}} \Big[\max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \sum_{s' \in \mathcal{S}} \xi(s') P(s'|s,a) Q_i(s', a', y\xi(s'))\Big] \\
&= c(s,a) + \gamma \min_{a' \in \mathcal{A}} \Big[\max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \sum_{s' \in \mathcal{S}} \xi(s') P(s'|s,a) \min_{\pi \in \Pi_i} CVaR_{y\xi(s')}(\mathcal{C}_{0,i-1} + \gamma^i Q_0|s_0 = s', a_0 = a', \pi)\Big] \\
&= \min_{a' \in \mathcal{A}} \Big[c(s,a) + \max_{\xi \in \mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \mathbb{E}_{\xi(s')}[\min_{\pi \in \Pi_i} CVaR_{y\xi(s')}(\mathcal{C}_{1,i} + \gamma^{i+1} Q_0|s', a')|s_0 = s, a_0 = a, \pi)]\Big] \\
&= \min_{\pi \in \pi_{i+1}} CVaR_y(\mathcal{C}_{0,i} + \gamma^{i+1} Q_0|s_0 = s, a_0 = a, \pi). \quad (10)
\end{aligned}$$

因此，通过归纳法，证明完成。□

本节的主要结果证明，贝尔曼算子 $\mathbf{T}[Q](s,\ a,\ y)$ 的不动点是最优价值函数 $Q^*(s,\ a,\ y)$。

定理 1。当 $k \to \infty$, $Q_k(s,a,y)$ 收敛到唯一的不动点 $\mathbf{T}[Q^*](s,a,y) = Q^*(s,a,y)$ 时，对所有 $s \in \mathcal{S}, a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$ 均成立。

证明。证明由引理1和引理 2 得出。由于 CVaR 贝尔曼算子 $\mathbf{T}[Q](s,\ a,\ y)$ 具有压缩性质，因此我们有 $\lim_{k \to \infty} Q_k(s,\ a,\ y)$ 收敛到唯一的不动点 $\mathbf{T}[Q^f](s,\ a,\ y) = Q^f(s,\ a,\ y)$。根据引理 2，并利用 $\gamma < 1$ 这一事实，我们可以写出

$$\begin{aligned}
&Q^f(s,a,y) \\
&= \lim_{k \to \infty} Q_k(s,a,y) \\
&= \min_{\pi \in \Pi_H} CVaR_y(\lim_{k \to \infty} \mathcal{C}_{0,k-1} + \gamma^k Q_0|x_0 = s, a_0 = a, \pi) \\
&= \min_{\pi \in \Pi_H} CVaR_y(\mathcal{C}_{0,\infty}|x_0 = s, a_0 = a, \pi) \\
&= Q^*(s,a,y).
\end{aligned}$$

至此，证明完毕。□

我们注意到，最优价值函数 $Q^*(s,a,y)$ 可由一个平稳（确定性）马尔可夫策略 $\mu^*(s,y)$ 实现，其中 μ^* 是从当前状态 s 和 CVaR 置信水平 y 到动作 a 的映射，即 $\mu^* : (\mathcal{S}, \mathcal{Y}) \to \mathcal{A}$。我们通过反复应用 CVaR 贝尔曼算子得到不动点 $Q^*(s,a,y)$，然后在 $Q^*(s,a,y)$ 上确定一个贪心策略 μ^*，使得 $\mu^*(s,y) = \text{argmin}_a Q^*(s,a,y)$ [28]。接着，根据式 (9) 中 $Q^*(s,a,y)$ 的定义可知，贪心策略 μ^* 确实能实现最优性。因此，即使原问题 (4) 是在历史依赖策略集合上定义的，寻找最优策略的问题仍然是可处理的。

我们利用定理 1 的洞见，指导分布式多智能体强化学习算法的设计，其中智能体旨在优化风险感知目标。该目标以基于 CVaR 的折扣累积成本形式表示（见式 (5)）。我们的*CVaR QD*学习 算法的更新规则参考了 [28] 中提出的 QD-learning 方案。不过，优化基于 CVaR 的价值函数（而非 [28] 中基于期望的函数）需要我们构建额外的数学结构，下面将对此进行说明。

在 *CVaR QD*学习 更新的每次迭代中，智能体的增广状态-动作价值函数演化为以下各项的加权和：(i) 上一次迭代的增广状态-动作价值函数；(ii) 在给定状态下采取特定动作后观测到的代价所产生的创新 项；以及 (iii) 与智能体及其邻居 $\mathcal{N}(\cdot)$ 在智能体间通信图 $\mathcal{G}$中的增广状态-动作价值之差相对应的共识 项。具体而言，每个智能体 n 的序列 $\{Q_{n,k}(s,a,y)\}$按以下方程演化：

$$\begin{aligned} Q_{n,k+1}(s_k,a_k,y_k) &= (1-\alpha_k)Q_{n,k}(s_k,a_k,y_k) \\ &+ \alpha_k\Big(c_n(s_k,a_k) + \gamma \min_{a'\in\mathcal{A}}\big[\max_{\xi\in\mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \\ &\qquad \xi(s_{k+1})Q_{n,k}(s_{k+1},a',y_k\xi(s_{k+1}))\big]\Big) \\ &- \beta_k \sum_{l\in\mathcal{N}(n)}\big(Q_{n,k}(s_k,a_k,y_k) - Q_{l,k}(s_k,a_k,y_k)\big), \end{aligned} \tag{11}$$

其中，权重序列 $\{\alpha_k\}$ 和 $\{\beta_k\}$ 由 (2) 和 (3) 给出，其中 $\tau_1 \in (\frac{1}{2}, 1]$ 和 $\tau_2 \in (0, \tau_1 - \frac{1}{2})$。与此处类似，权重序列满足 $\sum_k \alpha_k = \infty$ $\sum_k \beta_k =$, , $\sum_k \alpha_k^2 < \infty$, $\sum_k \beta_k^2 < \infty$ 和 $\frac{\beta_k}{\alpha_k} \to \infty$，当 $k \to \infty$时。方程 (2) 和 (3) 中的常数 a 和 b 还满足：对于所有 k，$(I_N - \beta_k L - \alpha_k I_N)$ 均为半正定。

由于代价 $\{c_n(s, a)\}$ 和参数 $\{\xi(s)\}$有界，且为所有智能体 n选取的初始增广状态-动作价值函数 $\{Q_{n,0}(s, a, y)\}$ 均有界，并且对所有 $s \in \mathcal{S}$，$a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$，我们可以证明 $\{Q_{n,k}(s_k, a_k, y_k)\}$ 逐路径有界，即 $P(\sup_k ||Q_{n,k}||_\infty < \infty) = 1$。接下来的结果将证明，各智能体的增广状态-动作价值函数渐近地达到共识。

定理 2 (条件风险价值 QD学习的共识)。 对于公式 (11) 中的条件风险价值 *QD*学习更新，在所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$ 下，每个智能体都会渐近地达到共识。也就是说，

$$P(lim_{k\to\infty}||Q_{n,k}(s,a,y) - \bar{Q}_k(s,a,y)|| = 0) = 1, \tag{12}$$

其中

$$\bar{Q}_k(s,a,y) = \frac{1}{N}\sum_{n=1}^{N} Q_{n,k}(s,a,y). \tag{13}$$

证明。 对于智能体 n，我们可以将式 (11) 中的更新写为：

$$z_{k+1} = (I_N - \beta_k L - \alpha_k I_N)z_k + \alpha_k(U_k + J_k) \tag{14}$$

其中， L 是图 $\mathcal{G}$ 的拉普拉斯矩阵，且

$$\begin{aligned} z_k &:= [Q_{1,k}(s_k = s, a_k = a, y_k = a)\cdots \\ &\qquad Q_{n,k}(s_k = s, a_k = a, y_k = y)]^T, \\ U_k &:= \gamma \min_{a'\in\mathcal{A}}\big[\max_{\xi\in\mathcal{U}_{CVaR}(y,P(\cdot|s,a))} \xi(s_{k+1} = s') \\ &\qquad Q_{n,k}(s_{k+1} = s', a', y_k\xi(s_{k+1}))\big], \\ J_k &:= c_n(s_k = s, a_k = a). \end{aligned}$$

令 $\hat{z}_k = z_k - \frac{1}{N}1_N^T z_k 1_N$，其中 1_N是 $\mathbb{R}^N$中的一个向量，其所有元素均为 1。于是，公式 (14) 可写为

$$\hat{z}_{k+1} = (I_N - \beta_k L - \alpha_k I_N)\hat{z}_k + \alpha_k(\hat{U}_k + \hat{J}_k) \tag{15}$$

其中 $\hat{U}_k = U_k - \frac{1}{N}1_N^T U_k 1_N$和 $\hat{J}_k = U_k - \frac{1}{N}1_N^T J_k 1_N$。然后，按照 [28]中的论证，并应用 [28], 中的引理 4.2，我们可以写出 $k \geq k_0$的表达式

$$||(I_N - \beta_k L - \alpha_k I_N)\hat{z}_k|| \leq (1 - c_2\ r_k)||\hat{z}_k|| \tag{16}$$

其中 $c_2 \in (0, 1)$、$0 \leq r_k \leq 1$ 和 $k_0 \in \mathbb{N}$。结合公式 (15) 和 (16)，我们可以写出 $k \geq k_0$的表达式

$$||\hat{z}_{k+1}|| \leq (1 - c_2\ r_k)||\hat{z}_k|| + \alpha_k(||\hat{U}_k|| + ||\hat{J}_k||). \tag{17}$$

由于对所有智能体 n 以及所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$，$\{Q_{n,k}(s, a, y)\}$、$\{\xi(s)\}$ 和 $\{c_n(s, a)\}$ 均有界，因此我们有 $\{||U_k||\}$ 和 $\{||J_k||\}$ 有界。利用[28] 中的引理 4.1，我们可以得出结论：对所有 $\tau \in (0\ \tau_1 - \tau_2 - \frac{1}{2})$，$P(z_k \to (,),$ 当 $0 = 1 \Rightarrow P(z_k \to \frac{1}{N}1_N^T z_k 1_N - = 1 \Rightarrow P(Q_{n\,k}\ s\ a\ y \to$ $k \to \infty$时，$P((k+1)^\tau \hat{z}_k \to 0) = 1$。因此，对所有 n 以及所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$和 $y \in \mathcal{Y}$，当 $k \to \infty$时，我们有 $(\hat{}) () (, (, ,) Q_n(s, a, y)) = 1$。由此恢复公式 (12)。 □

为了有效求解式 (11) 中关于 ξ 的最大化问题，我们现在证明 CVaR QD 学习更新能够保持 $\{y\ Q_{n,k}(s, a, y)\}$ 的凹性。我们注意到，这一问题仅存在于基于 CVaR 的更新中，而在 [28] 中基于期望的 QD 学习更新中并不存在。接下来的结果将对此见解加以形式化。

定理 3. 假设 $\{y\ Q_{n,k}(s, a, y)\}$ 对所有智能体 n 以及所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$，关于 y 都是凹的。那么，$\{y\ Q_{n+1,k}(s, a, y)\}$ 对所有智能体 n以及所有 $s \in \mathcal{S}$、$a \in \mathcal{A}$ 和 $y \in \mathcal{Y}$，关于 y 也都是凹的。

Proof. Let, $y_1, y_2 \in \mathcal{Y}$, $\lambda \in [0, 1]$ and $y_\lambda = (1-\lambda)y_1 + \lambda y_2$.

因此，我们可以写成

$$
\begin{aligned}
&(1-\lambda)y_1 Q_{n,k+1}(s,a,y_1)+\lambda y_2 Q_{n,k+1}(s,a,y_2)\\
&=(1-\alpha_k-|\mathcal{N}(n)|\beta_k)\Big((1-\lambda)y_1 Q_{n,k}(s,a,y_1)\\
&+\lambda y_2 Q_{n,k}(s,a,y_2)\Big)+\beta_k\sum_{l\in\mathcal{N}(n)}\big((1-\lambda)y_1 Q_{l,k}(s,a,y_1)\\
&+\lambda y_2 Q_{l,k}(s,a,y_2)\big)+\alpha_k\Big(((1-\lambda)y_1+\lambda y_2)c_n(s,a)+\\
&\gamma\Big(\min_{a_1'\in\mathcal{A}}\Big[\max_{\xi_1\in\mathcal{U}_{CVaR}(y,P(\cdot|s,a))}\xi_1(s')(1-\lambda)y_1\\
&Q_{n,k}(s',a',y_1\xi_2(s))\Big]+\min_{a_2'\in\mathcal{A}}\Big[\max_{\xi_2\in\mathcal{U}_{CVaR}(y,P(\cdot|s,a))}\xi_2(s')y_2\\
&Q_{n,k}(s',a_2',y_2\xi_2(s))\Big]\Big)\\
&\le(1-\alpha_k-|\mathcal{N}(n)|\beta_k)y_\lambda Q_{n,k}(s,a,y_\lambda)+\beta_k\sum_{l\in\mathcal{N}(n)}(y_\lambda\\
&Q_{l,k}(s,a,y_\lambda))+\alpha_k\big(y_\lambda c_n(s,a)+\gamma\big(\min_{a'\in\mathcal{A}}\Big[\\
&\max_{\substack{\xi_1\in\mathcal{U}_{CVaR}(y_1,P(\cdot|s,a))\\ \xi_2\in\mathcal{U}_{CVaR}(y_2,P(\cdot|s,a))}}\xi_1(s')(1-\lambda)y_1 Q_{n,k}(s',a',y_1\xi_2(s))\\
&+\xi_2(s')y_2 Q_{n,k}(s',a_2',y_2\xi_2(s))\Big]\big)\\
&\le(1-\alpha_k-|\mathcal{N}(n)|\beta_k)y_\lambda Q_{n,k}(s,a,y_\lambda)+\beta_k\sum_{l\in\mathcal{N}(n)}(y_\lambda\\
&Q_{l,k}(s,a,y_\lambda))+\alpha_k\big(y_\lambda c_n(s,a)+\gamma\big(\min_{a'\in\mathcal{A}}\\
&\Big[\max_{\substack{\xi_1\in\mathcal{U}_{CVaR}(y_1,P(\cdot|s,a))\\ \xi_2\in\mathcal{U}_{CVaR}(y_2,P(\cdot|s,a))}}((1-\lambda)y_1\xi_1(s')+\lambda y_2\xi_2(s'))\\
&Q_{n,k}(s',a',(1-\lambda)y_1\xi_1(s')+\lambda y_2\xi_2(s'))\Big]\big)
\end{aligned}
$$

注意，$(1-\alpha_k-|\mathcal{N}(n)|\beta_k)\ge 0$ 和 $\beta_k\ge 0$。在上述推导中，我们用到了 min 运算的凹性、凹函数正线性组合的凹性，以及如下假设：$\{y\ Q_{n,k}(s,a,y)\}$ 对所有智能体 n 以及所有 $s\in\mathcal{S},a\in\mathcal{A}$ 和 $y\in\mathcal{Y}$，关于 y 都是凹的。

我们定义 $\xi:=$ ~~$(1-\lambda)y_1\xi_1(s')+\lambda y_2\xi_2(s')\,y_\lambda$~~。注意，当 $\xi_1\in\mathcal{U}_{CVaR}(y_1,P(\cdot|s,a))$ 和 $\xi_2\in\mathcal{U}_{CVaR}(y_2,P(\cdot|s,a))$ 时，我们可以写成 $\xi\in[0,\frac{1}{y_\lambda}]$，其中 $\sum_{s'\in\mathcal{S}}\xi(s')P(s'|s,a)=1$。因此，我们可以写成

$$
\begin{aligned}
&(1-\lambda)y_1 Q_{n,k+1}(s,a,y_1)+\lambda y_2 Q_{n,k+1}(s,a,y_2)\\
&\le(1-\alpha_k-|\mathcal{N}(n)|\beta_k)y_\lambda Q_{n,k}(s,a,y_\lambda)+\beta_k\sum_{l\in\mathcal{N}(n)}(y_\lambda\\
&Q_{l,k}(s,a,y_\lambda))+\alpha_k\big(y_\lambda c_n(s,a)+\gamma\big(\min_{a'\in\mathcal{A}}\\
&\Big[\max_{\xi\in\mathcal{U}_{CVaR}(y,P(\cdot|s,a))}y_\lambda\xi(s')Q_{n,k}(s',a',y_\lambda\xi(s'))\Big]\big)\\
&=y_\lambda Q_{n,k+1}(s,a,y_\lambda),
\end{aligned}
$$

由此完成证明。 □

V. CVAR QD学习算法

定理2和定理3是设计我们的*CVaR QD*学习 算法的关键组成部分。本节介绍我们的算法，并说明如何克服算法实现中的一些挑战。

首先，Q 值 $Q_{n,k}(s,a,y)$ 中的参数 y 取值范围为连续区间 $\mathcal{Y}=(0,1]$。我们通过以下方式克服这一挑战：按照 [18] 中提出的方法，将 $\mathcal{Y}$ 离散化为足够多的区间 m，并且只考虑每个区间的端点。然后，我们可以改写为 $\mathcal{Y}=[y_1,\cdots,y_m]$，其中 $0<y_1<\cdots<y_m\le 1$。

第二个挑战源于在式（11）的更新中，需要对$\xi(s_{k+1})$ $Q_{n,k}(s_{k+1},a',y_k\xi(s_{k+1}))$在 ξ 上进行最大化。我们的算法通过求解一个修改后的最大化问题来克服这一挑战，

$$\frac{1}{y_k}\max_{\xi(s_{k+1})}\big[(y_k\xi(s_{k+1}))Q_{n,k}(s_{k+1},a',y_k\xi(s_{k+1}))\big].$$

定理 3 中证明的凹性性质使我们得以断定，$\big[(y_k\xi(s_{k+1}))$ $Q_{n,\ k}(s_{k+1},\ a',\ y_k\xi(s_{k+1}))\big]$ 的任何局部最大值点实际上都是全局最大值点。

最后一个挑战是在最大化步骤中确定$\xi(s_{k+1})$的一个可容许取值，因为强化学习中的转移概率 $P(s_{k+1}|s_k,a_k)$ 未知，只有在与环境交互时才会向智能体揭示。我们的实现通过首先为所有 $s'\in\mathcal{S}$ 选择 $\xi(s')=1$ 来应对这一挑战。然后，在 CVaR QD 学习算法的每次迭代中，~~我们以一种使~~ $\bar{\xi}(s'|s,a)\,\hat{}P(s'|s,a)\le 1$ 成立的方式更新上界 $\xi(s'|s,a)$，其中 $\hat{}P(s'|s,a)$ 是对$P(s'|s,a)$的估计。我们采用 Q学习中的一个标准假设：选择一个（随机）基策略 π_0，使每个状态-动作对都被无限次访问， [2], [30] 以计算一个 ~~$\hat{}$~~估计 $P(s'|s,a)$，从而得到界 $\xi(s'|s,a)$。如果 $\hat{}P(s'|s,a)\ne 0$，我们使用 $\bar{\xi}(s'|s,a)=1\hat{}P(s'|s,a)$（否则，我们~~保留初始猜测~~ $\xi(s'|s,a)=1$）。对于 ~~对~~ ~~ξ 的~~最大化，我们将使用任意满足 $\xi(s_{k+1})$ $y_k\in\mathcal{Y}$的 $\xi(s_{k+1})\le\bar{\xi}(s_{k+1}|s_k,a_k)$。

我们的条件风险价值 *QD*学习 算法的步骤详见算法1。

算法1 CVaR *QD*-学习

1: 输入：y, $\{\alpha_k\}$, $\{\beta_k\}$, γ, $\mathcal{S}$, $\mathcal{A}$, $\mathcal{Y}=[y_1,\cdots,y_m]$, N.
2: **初始化**：$\{Q_{n,0}(s,a,y)\}$, $\xi(s'|s,a)=1$, $k=0$.
3: 输出：$\{Q_n(s,a,y)\}$。
4: 循环 每个回合
5: 初始化 $s_0\in\mathcal{S}$
6: **循环** 在 s_k 中执行动作 a_k，并观察下一个状态 s_{k+1}.
7: ~~更新~~ $\bar{\xi}(s_{k+1}|s_k,a_k)$。
8: 每个智能体 n 在本地更新 $\{Q_{n,k+1}(s_k,a_k,y)\}$ 依据公式 (11)，对所有 $y\in\mathcal{Y}$ 使用任意 $\xi(s_{k+1})\le\bar{\xi}(s_{k+1}|s_k,a_k)$，使得 $\xi(s_{k+1})$ $y_k\in\mathcal{Y}$。
9: $k+1\leftarrow k$。
10: **结束循环** 直到 s是终止状态。
11: **结束循环**

由于所有智能体渐近地达成共识（定理 2），因此可以基于任一智能体对增广动作价值函数的评估采用贪心策略，以选择所需策略。因此，对于置信水平 y 下的条件风险价值，可将所需策略选为 $\mu(s,\ y)=lim_{k\to\infty}\mathrm{argmin}_a\,Q_{n,\ k}(s,\ a,\ y)$，其中任意 $n=1,\ \cdots,\ N$ 均成立。

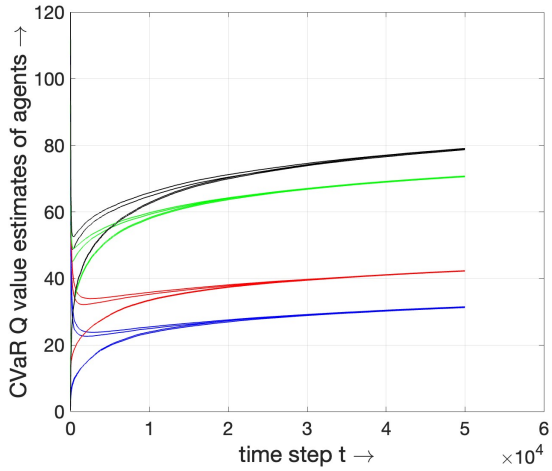


图1：该图展示了在 CVaR 置信水平为 $y = 0.7$ 时，四个不同非邻居智能体的 $CVaR\ Q$–值估计的演化。每个状态-动作对（s， a）用不同颜色表示。我们观察到，遵循算法1时，智能体的 $CVaR\ Q$–值对于所有（s， a）对都达成共识。

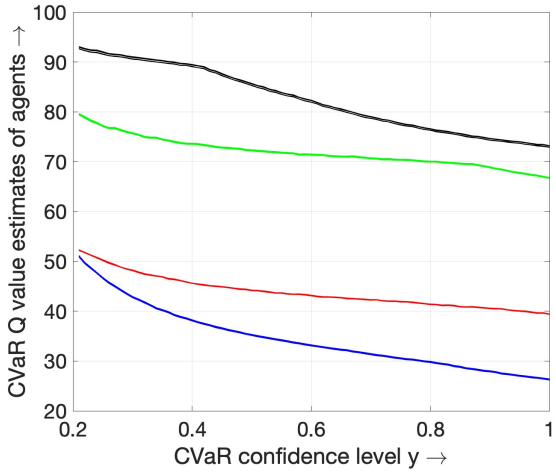


图 2：该图展示了 CVaR 置信水平为 $y \in [0.2, 1]$ 时，四个非邻居智能体在时间步 $t = 50000$ 的 $CVaR\ Q$–值估计变化。每种颜色表示一个 $(s,\ a)$ 对的 $CVaR\ Q$–值。我们观察到，随着智能体的风险敏感性从风险感知型（$y < 1$）转变为风险中性（$y = 1$），$CVaR\ Q$–值会降低。直观地说，这表明在目标为最小化累积成本时，风险感知行为会产生更高的 $(CVaR)\ Q$– 值。

VI. 实验评估

本节开展实验，以评估*CVaR QD*学习 算法（算法 1）的性能。我们首先介绍实验环境，然后报告实验结果。

A. 环境

我们的设置由一个包含 $N = 40$ 个智能体的网络构成。每个智能体与其两个最近邻 [28] 通信。状态空间和动作空间均为二值，因此有 $|\mathcal{S} \times \mathcal{U}| = 4$，以及 8 种不同的转移概率。从区间 $[0, 1]$ 中通过均匀采样随机选取其中四种转移概率；由此确定其余四种转移概率。对于每个智能体 n，代价 c_n（s， a）从均匀分布中选取，该分布对每个状态-动作对（s， a）具有不同的均值。我们设定折扣因子 $\gamma = 0.7$，并将方程（2）和（3）中的参数设为 $\tau_1 = 0.2$、 $\tau_2 = 0.3$、 $a = 0.2$ 和 $b = 0.1$。区间 $(0, 1]$ 被离散为 100 个间距相等的子区间，以量化与 CVaR 相关的置信水平。因此，我们有 $\mathcal{Y} = \{0.01, 0.02, \cdots, 0.99, 1\}$。我们通过实例化单条轨迹 $\{s_t, a_t\}$ 来评估算法1。在每一步 t ，处于状态 s_t 时，动作 a_t 通过均匀采样随机选取；下一个状态 s_{t+1} 由转移概率决定。初始状态 s_0 随机选取。我们还将智能体的 Q–值初始估计值设为不同的数值。

B. 评估结果

我们展示了在遵循我们的*CVaR QD*学习 算法（算法 1）时，四个不同的非邻居智能体的 $CVaR\ Q$–价值估计的演变过程，见图 1，其中，每个状态-动作对（s， a）用不同的颜色表示。所有状态-动作对的 CVaR 置信水平均设为 $y = 0.7$ 。我们观察到，对于所有（s， a）对，智能体渐近地就其 $CVaR\ Q$–价值估计达成共识。收敛速率与该状态-动作对在轨迹中被访问的次数成正比（因为有更多样本可用于更新相应的 $CVaR\ Q$–价值）。

图 2 展示了在置信水平 $y \in [0.2, 1]$ 取不同值时，时间步 $t = 50000$ 的 $CVaR\ Q$– 值估计变化。随着智能体达成共识，我们观察到，对于所有状态-动作对，$CVaR\ Q$– 值会随 y 增大而降低。当 $y = 1$ 时，$CVaR\ Q$– 值最低；也就是说，这种情形与更传统的基于期望的 Q– 值 [2], [28] 相同。直观地说，这一结果表明，随着智能体的风险敏感性从风险中性（$y = 1$）转变为风险感知（$y < 1$），其各自的 $(CVaR)\ Q$– 值会更高（因为其目标是最小化累积成本）。

VII. 结论

本文提出了一种分布式多智能体强化学习（MARL）框架，通过学习风险感知策略进行决策。我们使用条件风险价值（CVaR）来刻画风险敏感成本函数，并引入贝尔曼算子来描述基于CVaR的状态-动作价值函数。从理论上，我们证明了该算子具有压缩映射性质，且会收敛到最优价值。基于这一发现，我们开发了一种名为*CVaR*

QD-Learning 算法，并证明了与各智能体相关的风险感知价值函数能够达成共识。我们介绍了在实现*CVaR QD-Learning* 算法过程中遇到的多项挑战及其解决方案，并通过仿真实验评估了算法性能。我们还展示了风险参数对智能体达成共识时其价值函数的影响。

我们方法的一种可能扩展是研究部分智能体可能具有恶意或遭到破坏的情形。尽管已有一些初步工作研究了这一方向，但这些工作采用的是最小化平均成本准则 [35]。另一个值得探讨的问题是连续状态-动作设置，其中策略由（深度）神经网络参数化。关于 MARL 风险敏感策略设计的初步研究，主要关注集中式训练机制 [23]。在综合分布式风险感知策略的背景下研究韧性解决方案，并开发分布式算法以刻画连续状态-动作空间中的风险感知行为，是一个很有前景的研究方向。

参考文献

[1] M. L. Puterman，马尔可夫决策过程：离散随机动态规划。John Wiley & Sons，2014年。[2] R. S. Sutton 和 A. G. Barto，强化学习：导论。MIT Press，2018年。[3] D. P. Bertsekas，动态规划与最优控制，第*1*卷，第*4*版2017版。Athena Scientific，。[4] R. Hafner 和 M. Riedmiller，“反馈控制中的强化学习”，机器学习，第84卷，页137–169，2011年。[5] V. Mnih 等，“通过深度强化学习实现人类水平的控制”，《自然》，第518卷，第7540期，2015年。[6] D. Silver 等，“利用深度神经网络和树搜索掌握围棋”，《自然》，第529卷，第7587期，2016年。[7] C. Zhang、P. Patras 和 H. Haddadi，“移动与无线网络中的深度学习：综述”，*IEEE*《通信调查与教程》，第21卷，第3期，页2224–2287，2019年。[8] D. Sadigh、S. Sastry、S. A. Seshia 和 A. D. Dragan，“为能够利用对人类动作影响的自动驾驶汽车进行规划。”载于《机器人学：科学与系统》，2016年。[9] Z. Yan 和 Y. Xu，“随机电力系统的数据驱动负荷频率控制：一种采用连续动作搜索的深度强化学习方法”，*IEEE*《电力系统汇刊》，第34卷，2，2018年 no.。[10] C. You、J. Lu、D. Filev 和 P. Tsiotras，“运用强化学习和深度逆向强化学习为自动驾驶车辆进行高级规划”，《机器人与自主系统》，第114卷，页1–18，2019年。[11] R. A. Howard 和 J. E. Matheson，“风险敏感型马尔可夫决策过程”，《管理科学》，第18卷，第7期，页356–369，1972年。[12] A. Nilim 和 L. El Ghaoui，“转移矩阵不确定时马尔可夫决策过程的鲁棒控制”，《运筹学》，第53卷，第5期，页780–798，2005年。[13] V. S. Borkar，“用于风险敏感控制的Q学习”，《运筹学数学》，第27卷，第2期，页294–311，2002年。[14] R. T. Rockafellar 和 S. Uryasev，“一般损失分布的条件风险价值”，《银行与金融杂志》，第26卷，第7期，页1443–1471，2002年。[15] M. Ahmadi、U. Rosolia、M. D. Ingham、R. M. Murray 和 A. D. Ames，“约束风险规避型马尔可夫决策过程”，载于*AAAI*人工智能大会，2021年。[16] M. P. Chapman、R. Bonalli、K. M. Smith、I. Yang、M. Pavone 和 C. J. Tomlin，“使用条件风险价值进行风险敏感安全分析”，*IEEE*《自动控制汇刊》，2021年。[17] L. Lindemann、G. J. Pappas 和 D. V. Dimarogonas，“信号时序逻辑规范下非完整系统的控制屏障函数”，载于*IEEE*决策与控制会议（*CDC*）。IEEE，2020年，页1422–1428。[18] Y. Chow、A. Tamar、S. Mannor 和 M. Pavone，“风险敏感与鲁棒决策：一种条件风险价值优化方法”，神经信息处理系统进展，第28卷，2015年。[19] G. Serraino 和 S. Uryasev，“条件风险价值（CVaR）”，运筹学与管理科学百科全书，页258–266，2013年。[20] M. Tan，“多智能体强化学习：独立智能体与合作智能体”，载于国际机器学习会议论文集，1993年，页330–337。[21] L. Matignon、G. J. Laurent 和 N. Le Fort-Piat，“合作马尔可夫博弈中的独立强化学习者：关于协调问题的综述”，知识工程评论，第27卷，第1期，页1–31，2012年。[22] K. Zhang、Z. Yang 和 T. Bas,ar，“多智能体强化学习：理论与算法精选概述”，强化学习与控制手册，页321–384，2021年。[23] W. Qiu、X. Wang、R. Yu、R. Wang、X. He、B. An、S. Obraztsova 和 Z. Rabinovich，“RMIX：为合作强化学习智能体学习风险敏感策略”，神经信息处理系统进展，第34卷，页23 049–23 062，2021年。[24] J. Zhao、M. Yang、X. Hu、W. Zhou 和 H. Li，“DQMIX：多智能体强化学习的分布视角”，*arXiv*预印本 *arXiv:2202.10134*，2022年。[25] R. Lowe、Y. I. Wu、A. Tamar、J. Harb、O. Pieter Abbeel 和 I. Mordatch，“混合合作竞争环境下的多智能体演员-评论家”，神经信息处理系统进展，第30卷，2017年。[26] J. Foerster、G. Farquhar、T. Afouras、N. Nardelli 和 S. Whiteson，“反事实多智能体策略梯度”，载于*AAAI*人工智能大会论文集，第32卷，第1期，2018年。[27] T. Rashid、M. Samvelyan、C. S. De Witt、G. Farquhar、J. Foerster 和 S. Whiteson，“深度多智能体强化学习的单调价值函数分解”，机器学习研究期刊，第21卷，第1期，页7234–7284，2020年。[28] S. Kar、J. M. Moura 和 H. V. Poor，“QD学习：一种通过共识创新实现多智能体强化学习的协作式分布策略 + 创新”，*IEEE*信号处理汇刊，第61卷，第7期，页1848–1862，2013年。[29] F. R. Chung，谱图论。American Mathematical Soc.，1997年，第92卷。[30] C. J. Watkins 和 P. Dayan，“Q学习”，机器学习，第8卷，页279–292，1992年。[31] P. Artzner、F. Delbaen、J.-M. Eber 和 D. Heath，“一致风险度量”，数学金融，第9卷，第3期，页203–228，1999年。[32] G. C. Pflug 和 A. Pichler，“一致风险泛函的时间一致决策与时间分解”，《运筹学数学》，第41卷，第2期，页682–699，2016年。[33] V. S. Borkar 和 S. Chandak，“前景理论Q学习”，系统与控制快报，第156卷，页105009，2021年。[34] B. Ramasubramanian、L. Niu、A. Clark 和 R. Poovendran，“ 超越期望的强化学习 ”，载于 *IEEE* 决策与控制会议（*CDC*）。IEEE，2021年，页1528–1535。[35] Y. Xie、S. Mou 和 S. Sundaram，“迈向多智能体QD学习的韧性”，载于*IEEE*决策与控制会议。IEEE，2021年，页1250–1255。