17 决策智能

清华大学张崇洁：高效协作的多智能体强化学习前沿研究进展

整理：智源社区 窦勇强

在第二届北京智源大会“决策智能”专题论坛上，来自清华大学交叉信息研究院的张崇洁助理教授以“*Efficient Collaborative Multi-Agent Reinforcement Learning*”为题进行了演讲报告。

张崇洁，于 2011 年在美国麻省大学阿默斯特分校获计算机科学博士学位，而后在美国麻省理工学院从事博士后研究。目前的研究专注于人工智能、深度强化学习、多智能体系统、以及机器人学，担任清华大学交叉信息科学院助理教授，博士生导师，机器智能研究组主任。

过去几年人工智能得到了很大的发展，机器学习特别是深度学习方面在实际问题上的应用使人工智能受到了极大的关注。然而随着人工智能应用的不断广泛化和复杂化，使得研究者对人工智能提出更高的需求。研究趋势也从简单的模式识别到更加复杂的智能决策与控制，从单研究智能体的问题，过渡到解决多智能体的问题。

在本次演讲中，张崇洁系统讲述了高效协作的多智能体强化学习研究的前沿进展。他抽丝剥茧般回顾了当前多智能体学习存在的挑战，通过引入通信和角色的方式逐步解决挑战达到最佳性能水平的研究历程。此外，张崇洁通过理论分析工作展望了对未来多智能体强化学习研究的趋势性看法，见解独到，相信会给大家带来很多启迪。

演讲正文:

一、多智能体强化学习简介 [1]

研究者通常把具有感知和决策的能力的个体称为智能体 (agent)。智能体基于它的感知，可以做出相应的决策以及行动来改变周围的环境，多个智能体可以通过协作式的行为实现一个整体的目标。例如，在机器人集群控制中，每一个机器人就可以看作一个智能体；在一个风力发电场，每一个风机就可以看作一个智能体。

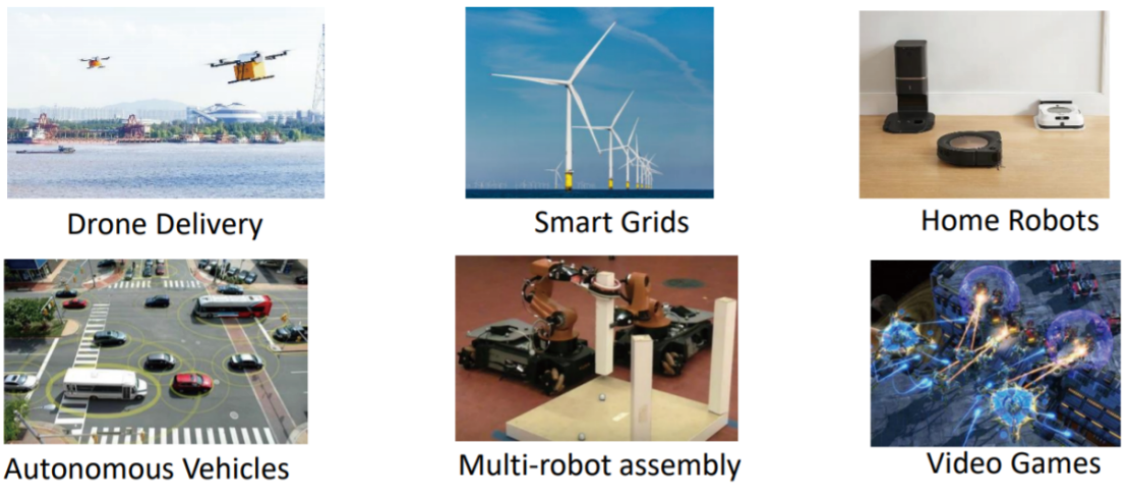


图 1：人工智能愈加复杂的应用场景

多智能体学习的问题可以分为三类：协作式多智能体，对抗式多智能体，以及混合式多智能体。其中，协作多智能体是一群智能体通过协同合作，来共同来优化整体目标的行为。

在大多数协作式多智能体问题中，环境往往是部分可观察的 (partially observable)：每个智能体只能观察环境的部分信息，而且环境的变化会存在一些随机性。这样一类复杂的多智能体协作决策问题，可以用一个较为通用的模型来刻画——部分可观察的 Markov 决策过程（Dec–POMDP）。

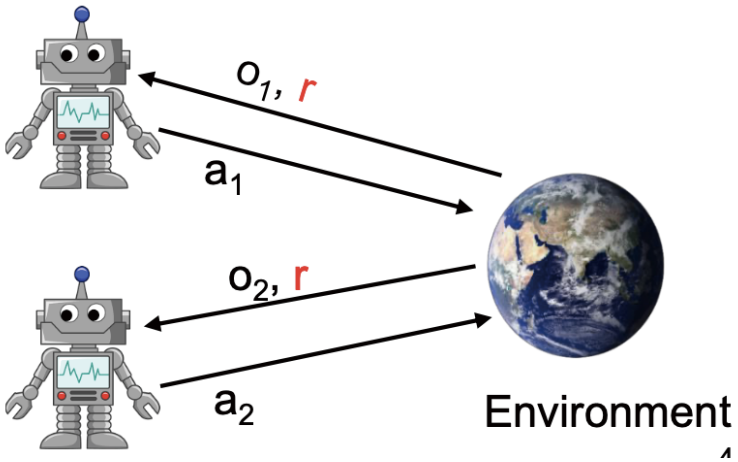


图 2：协作式多智能体模型 Dec–POMDP

Dec–POMDP 决策过程是非常通用的，它可以刻画大部分在不确定环境中多智能体决策的问题。从一种简单的角度来看，可将它视为把单个智能体的 Markov 决策过程过渡到多智能体的环境中。模型的运作方式如图 2 所示，在这个环境有两个机器人，在每一时刻，每个机器人都会根据它当前的感知输入选择某一个动作，执行这个动作之后将会改变环境中某一部分的状态。在这之后机器人通过进一步观察环境，得到新的观测。尽管每个机器人可能会有不同的观测信息，但它们会得到同一个反馈信号。因为这里整体的假设是协作式多智能体的范围，所以这个反馈称为联合的报酬 (joint reward)。在这类协作式多智能体问题中，研究者希望找到一组决策策略，使得智能体根据这个决策策略来执行它们的行动的时候，可以收获最大化的期望累计报酬。

- Objective: to find policies for agents to maximize expected cumulative rewards
- A local policy π_i for each agent i: mapping its observation-action history τ_i to its action
  - State is unknown, so beneficial to remember the history
- Joint policy $\boldsymbol{\pi} = < \pi_1, \ldots, \pi_n >$
- Value function: $Q_{tot}^{\boldsymbol{\pi}}(\boldsymbol{\tau}, \boldsymbol{a}) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, \boldsymbol{a}_0 = \boldsymbol{a}, \boldsymbol{\pi}\right]$
- Policy $\boldsymbol{\pi}(\boldsymbol{\tau}) = \text{argmax}_{\boldsymbol{a}}\, Q_{tot}^{\boldsymbol{\pi}}(\boldsymbol{\tau}, \boldsymbol{a})$

图 3：Dec–POMDP 形式化定义

这里寻找的决策策略，是指在前文定义的分布式 – 部分可观察的马尔可夫决策过程 (Dec–POMDP) 中，寻求一个映射关系，对每一个智能体把它的局部观察的历史映射到一个动作 (action) 上。而在 Dec–POMDP 的定义下，全局的环境状态是不可直接观测的。智能体往往需要记住一些历史的信息来辅助今后更好的决策。决策策略又称“联合策略”，所谓的联合策略是智能体策略的集合。为了更好的描述和解决多智能体决策的问题，研究者定义了一个值函数 (Q value function) 来量化任务中的执行目标，这个值函数是折扣的未来累计期望收益和 (Discounted future cumulative reward)。对给定一个任务如果能够学习出相应的值函数的话，那么智能体的最优联合策略也就相应得到。最优策略，可以直接从值函数中推导出来，即相对于行为变量 a (action) 取参数最大化 (argmax) 的结果。因此，求解协作式多智能体的 Dec–POMDP 问题，有两种途径：①直接学习一个最优的策略来最大化智能体的值函数（累积回报）；②学习出智能体的行为值函数，从值函数中推导出一个最优的策略。

在许多实际问题中，环境往往是非常复杂的，同时无法预先知道环境的模型。在这种情形下，愈加能够体现出强化学习的优势——不断试错：当环境不可知时，可以采用智能体与环境不断交互和探索的方式，在这个过程中建立关于环境的模型，进而进行动作的规划；也可以在环境学习奖励规则，在交互中直接学习智能体的策略。每个人的生活中都有强化学习的影子，在完成一个新的目标时，我们无法预先知道每一步该怎么走，而往往会有走完一步之后发现是好还是坏的反馈。人类学习的过程就是根据这种反馈信号来反思自己做出的行为，进而更好地面对未来的类似的场景。当后面遇到类似的情形时，一般会有两种做法：分析之前发生的类似经历，选择之前反馈最好的那个行为进行执行；同时可以尝试之前没做过的行为，如果新的动作比我们已知期望的回报要好，那今后就会要多做一些，如果比我们期望差，那么今后就会少做类似的行为。强化学习将这种试错 + 探索的方式，通过在贪婪选择目前已知的最优策略的算法中添加随机性来实现人类中学习的思想。

多智能体强化学习是在单智能体的基础上扩展到多个智能体的试错学习：在单智能体学习中，为单个智能体学习一个策略；多智能体学习则会选一组策略（也成为联合策略）为多个智能体服务。同时，在多智能体中一般有智能体之间是相互交互影响的而非完全独立的约束条件。在协作式多智能体中，智能体间的交互可以通过报酬来反映，每一个智能体的动作都会影响整体的报酬。那么，多智能体强化学习如何应用到多智能协作问题？又如何去学习最优的联合策略或者值函数呢？

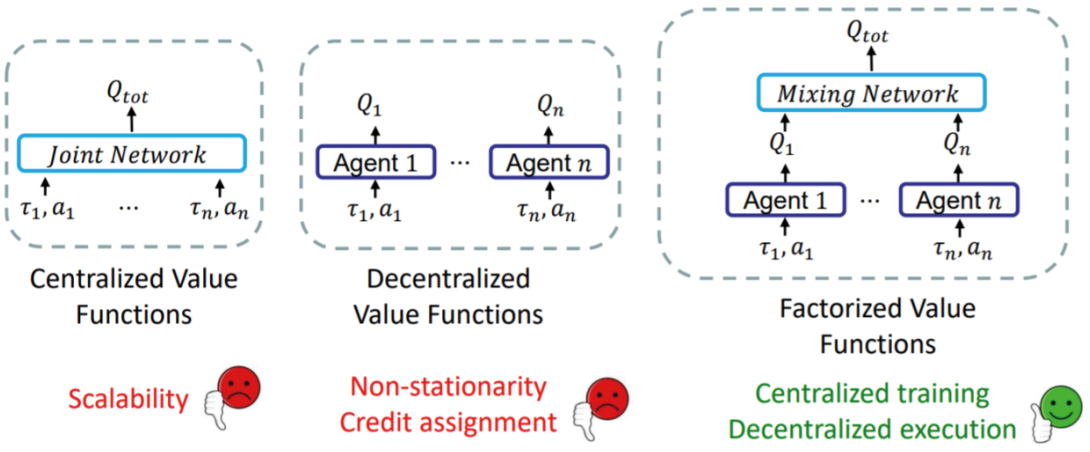


图 4：多智能体强化学习的三种范式

第一种方式是中心化的训练方法，将所有的智能体观测以及动作作为输入，建立一个联合的神经网络来输出联合值函数的函数值。中心化的方式存在的问题是： 不具有扩展性。在训练中观测空间是指数级增长的，即使神经网络表达形式特别强，可以学到这种映射关系，在执行过程中也会遇到通信的困难，要实时的去收集所有智能体的观测，做出决定之后再分配给所有智能体相应的决策动作。另外一种方式是分布式学习，每一个智能体有自己的网络，来实现学习的可扩展性。但分布式学习的问题在于当智能体在一个环境中共同学习的时候，环境成为非稳态的 (non-stationary)，不具有收敛性和最优性。另一个困难是不能很好地分配智能体获得的联合奖励，即信度分配 (credit assignment)。 因此，更好的方法是将这两种方法合并，称为可分解价值函数学习方法： 每个智能体都有其自己的行为价值网络或策略网络，并通过混合网络输出联合的实际价值。在这种情况下，每个智能体有分解的值函数也有共同的值函数。在执行过程中，每个只用仅仅通过个体的值函数或策略网络进行决策。 因此，它具有很好的可扩展性的同时解决了一些分布式学习问题，介于中心化训练和分布式学习两种方式之间的一种较好的学习范式。在具体的实现过程中，通过时间差分学习 (TD-learning)，将环境的反馈信息进行反向传播 (back propagation) 影响和更新每一个智能体的值函数，以解决环境的非稳态 (non-stationarity) 和信度分配的问题 (credit assignment) 的问题。

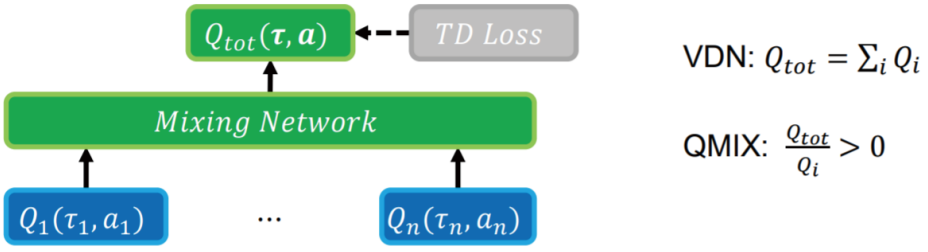


图 5：可分解的值函数学习范式（CTDE）

在中心化训练 – 分布式学习的范式之下，许多研究者提出一些结构约束假设的来实例化这种范式，其核心思想在于价值函数混合网络 (Mixing Network) 的确定。2017 年，领域研究者提出了价值分解网络 Value Decomposition Network (VDN)[2]，该方法通过简单求和的方式将智能体局部价值函数整合为联合价值函数。这种方式简单有效，但是限制了对于联合价值函数 Q_{tot} 的表达能力。

在此基础之上，研究者将 VDN 的较强假设进行了松弛，提出 QMIX[3]，它只假设总体的报酬对于个人报酬的偏导是大于零的。这样每个人提高自己的报酬就可以提高整体的报酬，以这样的方式实现 CTDE 范式。这一系列工作中，研究者提出了不同的假设来设计联合价值网络 (Mixing Network) (图 5 绿色部分) 进行中心化训练。同时在分布式执行过程中，把联合价值网络去掉，智能体通过自己的局部价值网络进行决策 (图 5 蓝色部分)，从值函数中推导得到相应的策略，来进行分布式执行。目前基于这种值分解的方法，在多智能体强化学习上取得一些前沿的一些结果，例如在星际争霸 2 的微操作游戏任务中，能学到很多有意思的策略，如放风筝策略，一个狂热者打两个海军陆战队，传统上海军陆战队是打不过的，通过不停“拉仇恨”的方法，就可以打败比自己强大的狂热者。

图 6：智能体学习到的放风筝战斗策略

目前值函数分解方法在一些复杂的问题上还存在着局限性，张崇洁给出了三点重要的方面。其一，前述方法不能很好地处理不确定性。因为当智能体学到自己的策略时，所有智能体进行完全分布的执行，随着环境的不确定，比如说状态不确定性等等，智能体间的协作则会变得不协调（miscoordination）。随着时间的推移，由于不能很好处理中环境的不确定性，这种不协调会得到积累进而导致较大的问题。此外，由于智能体间网络参数的共享（共享子网络或策略网络），目前的方式不能解决比较复杂的问题，智能体的行为趋向于单一化。即使参数共享的方式能够解决这样复杂的问题，也会导致庞大的网络的参数。此外，一些复杂的问题往往需要多样性异构智能体才能够解决。第三点，对这目前这些方法理论的分析，何时有效的理解有待深入。

针对上述三点挑战，张崇洁带领的团队近期进行了三方面的工作来解决。第一方面，通过通信优化的方式来解决不确定性的问题； 第二方面，通过这种角色涌现的方式来解决智能体参数动态共享的学习过程； 第三方面，关于线性值函数分解的理论分析工作。

二、引入最小化通信的多智能体协作 [4]

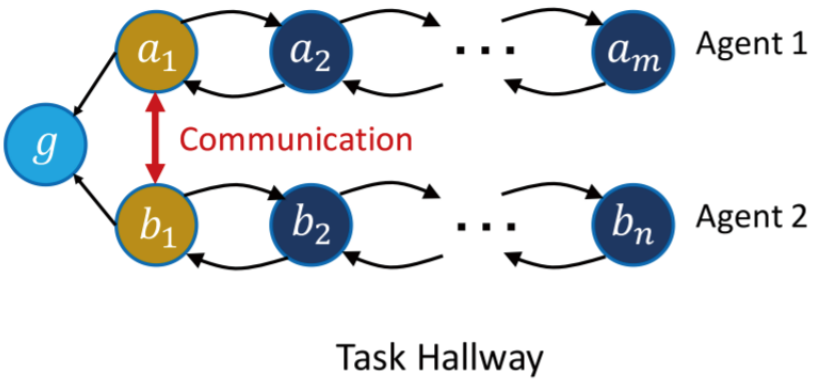


图 7：完全值函数分解存在的缺陷

图 7 中展示了完全值函数分解方法存在的缺陷，在该例中，有两个智能体分别位于长度不同的通道中，其中 a1 是智能体 1 所在通道出口的位置，b1 是智能体 2 的通道出口位置。当两个智能体同时到达通道出口目标 g 的时候，会收到一个报酬。如果两个智能体在不同的时间点到达目标 g，则不会收获报酬。假设初始状态是随机的，每个智能体初始位于不同的位置。由于智能体部分可观察的特性，即使中心化的训练方法表达能力非常强，最终两个智能体也只会以很小的概率同时到达收获报酬，因为智能体在执行的时候行为是完全确定的，向左走或者向右走，而互相不知道对方的初始位置，并且智能体间没有通信的交流，导致了之前的研究工作无法完成这类任务。如果考虑加入智能体间的通信交流，只要任意一个智能体快到达目的地之前，就是告诉其他智能体这个信息。并且等待接收对方同样到达目的地之前的信息，最后同时执行到达目标 g 来解决这个问题。

通信的加入可以解决此问题，但是这种通信不是长期需要的，比如说智能体 1 在位置时，无需进行通信向另一个智能体通报自己的位置。只有在离目标 g 最近的位置处进行通信的协调就可以，所以张崇洁团队提出在允许通信的同时需要最小化通信量。该方法称为近似可分解的值函数（NDQ）算法（图 8），该算法解决智能体通信的时间、内容以及通信对象的问题。

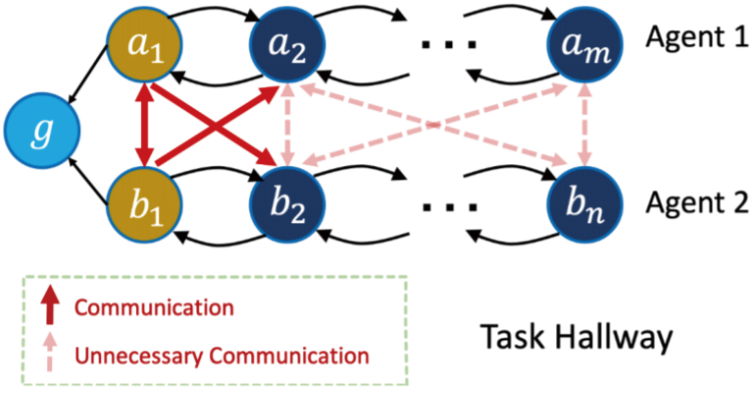


图 8：近似可分解的值函数方法 NDQ 解决的问题图解

NDQ 的框架如图 9 所示，传统值函数分解的结构下，每个智能体有自己独立的函数空间，基于智能体的局部观测历史进行优化。而 NDQ 则再此基础上进行改进，允许智能体之间可以相互发信息，智能体的策略不仅仅基于局部观察，还基于它收到的通信信息。

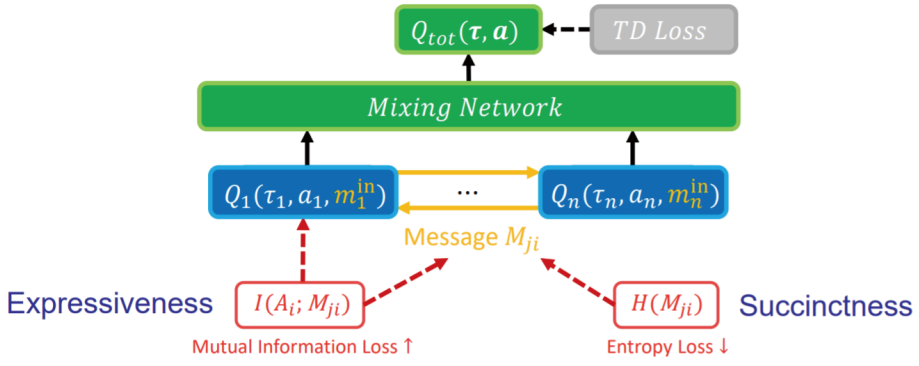


图 9：NDQ 算法框架结构图

然而，如果每个时间步，所有智能体之间都互相通信，则会变成了中心化的过程，不具有扩展性且浪费通信资源。因此，在该工作中，张崇洁团队加入两个约束条件来优化通信：①首先是通信消息的表达性约束，即希望这个智能体发送方到接收方之间发送的消息一定是对于接收方有帮助的，该约束通过最大化通信内容与智能体决策之间的互信息来实现。②信息简洁性约束，即尽最大努力缩小和减少信息的熵 (Entropy)，如果一条信息没有用，数值上的表现是 Entropy 很小，也就代表发送该条信息是没有必要的。因此通过两个约束，一是最大化互信息，二是最小化信息熵，保证了智能体之间的通信的简洁和有效。在实际实现中，无法直接优化这两个目标函数，所以通过借鉴变分推断中的思想，推导出了相应的变分下界来支持实际优化，有关具体推导过程感兴趣的读者可以参考 [4] 中附录。

- By deriving variational lower bounds, we get the equivalent communication loss:

$$L_c = \mathbb{E}_{M_j^{in},\mathrm{T}_i}[\mathcal{CE}[p(A_j|\mathrm{T})\|q_\xi(A_j|\mathrm{T}_j, M_j^{in})] + \beta D_{KL}[p(M_{ij}|\mathrm{T}_i)\|\mathcal{N}(0,1)]]$$

- Overall loss:

$$L = L_{TD} + L_c$$

图 10：NDQ 推导的优化目标函数

如图 10 所示，最终 NDQ 算法通过两个目标函数的结合进行端到端训练：第一个是训练混合网络的强化学习中常用的时间差分损失 TD loss，第二个是对于信息的通讯约束损失 communication loss。

在简单和复杂的任务中，NDQ 都表现出了非常优异的效果。首先，是传统完全值分解方法不能够解决的 Hallway 任务，NDQ 能够很好的学到最优策略。如图 16 所示，初始 t=1 时智能体在左 1 图中黄色的两个位置，此时由于它们没有到达通道出口，智能体间无需通信，NDQ 算法学习到了这个特征，因此此时没有信息的交流。这种不需要通信的状态持续到智能体 2 到达出口 b1 的时候，会向智能体 1 发送消息告知它已经到达出口 b1 的信息。最终当智能体 1 也到达对应的出口 a1 时，同样发送信息给智能体 2。因此 NDQ 算法在这个任务中学到了最优的策略。为了验证 NDQ 算法在复杂环境上的有效性，他在星际争霸 II 微操作管理的任务上同样进行了实验，本文主要显示六个实验结果（测试基准 Benchmark 是由牛津大学的团队提出，感兴趣的读者可以访问网站 https://sites.google.com/view/ndq 查看更多的实验结果。）

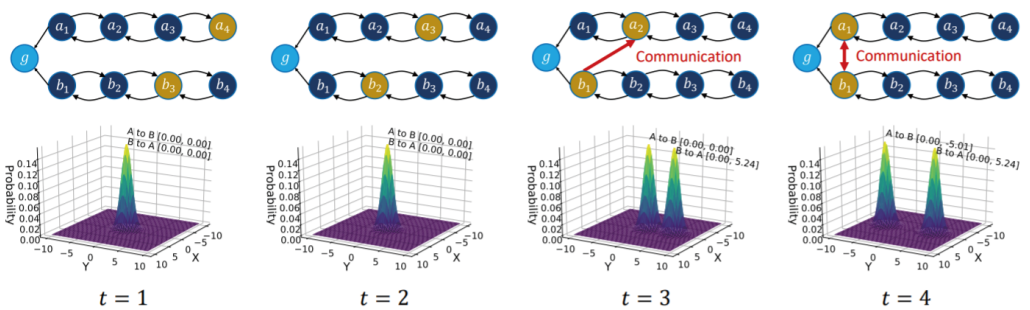


图 11：Hallway 任务中 NDQ 算法的表现结果

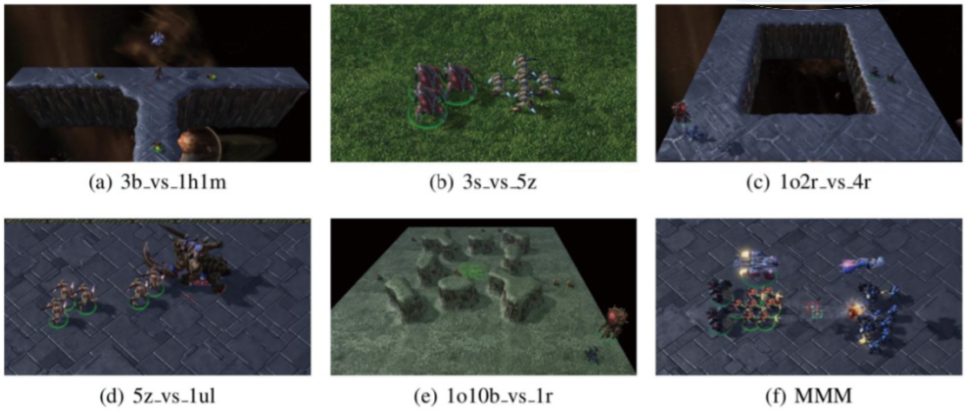


图 12：星际争霸 II 微操作管理实验 (https://sites.google.com/view/ndq)

实验的基线系统（baseline):

- QMIX[3]，完全值分解方法的代表实例。
- TacMAC[5]，基于注意力机制的通信的算法。
- 自行实现的 QMIX 与 TacMAC 的结合算法，测试 NDQ 的表现情况。

实验结果分析：TacMAC 是一个中心化的学习方法，它几乎在所有的复杂环境下都不能学习到好的策略；QMIX 在一些复杂环境中的表现不是很好； 当为 QMIX 增加通信的 Tarmac 算法时 , 性能表现有提高，但是与所提出的 NDQ 相比还是有一定的差距。基于注意力机制 Tarmac 算法的通信学习方法是一种软约束的学习方法，而 NDQ 则是显式地进行优化学习，能够很大程度提高和解决通信的优化问题。当对于通信进行剪切 (communication cut) 时，即将 80% 的通信信息进行丢弃时，接近可分解的值函数 NDQ 方法几乎没有受到影响，而对于基于注意力机制的 TacMAC 方法，则不能鲁棒应对信息丢失的环境的影响。

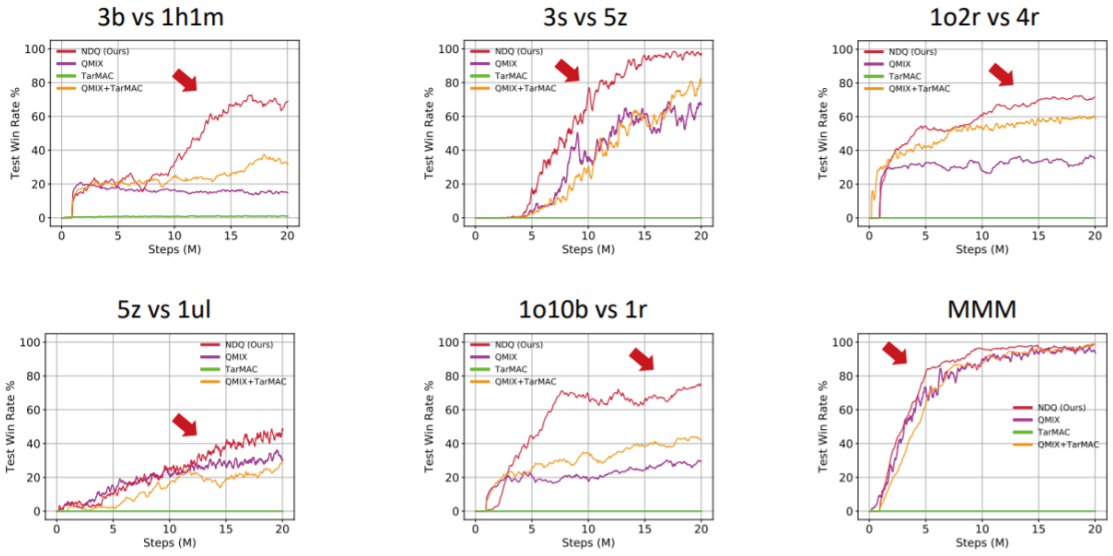


图 13：在星际争霸 II 中无信息丢弃的条件下的实验结果

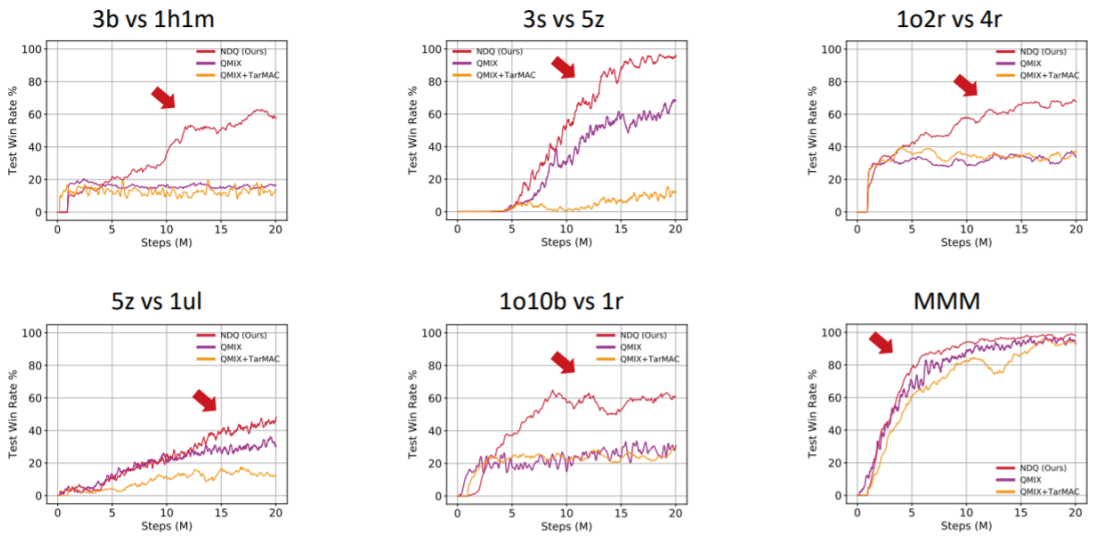


图 14：在星际争霸 II 中 80% 信息丢弃的条件下的实验结果

三、基于角色的多智能体强化学习 (Role-based)[6]

NDQ 算法通过通信的方式解决多智能体在执行过程中不确定性，本节的工作则是介绍通过如何通过基于角色的方式，来加速多智能体的学习。为什么需要动态的共享学习呢？原因是在很多复杂问题中，需要智能体有多样化的行为，或者甚至智能体本身必须为异构，即不能共享参数，因为不同功能的智能体不适合采用一个网络来表达。

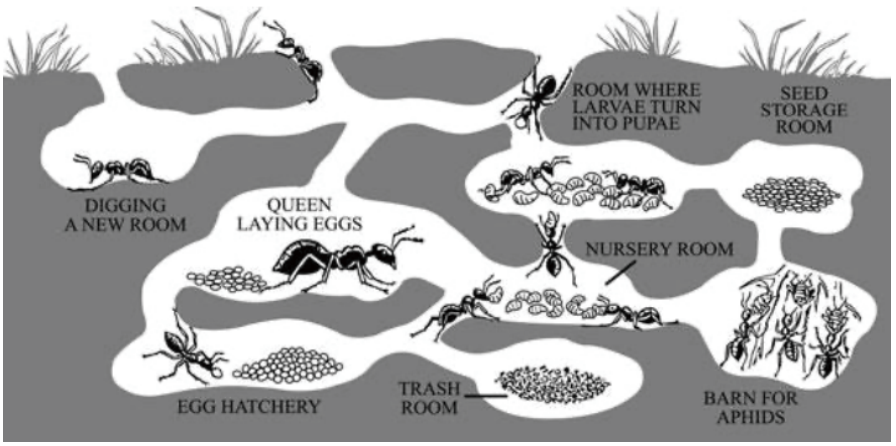


图 15：蚁群中不同蚂蚁有不同的角色

在现实生活或者自然界中有很多这样的情况，比如蚁群中不同蚂蚁有不同的角色，每个角色由不同的蚂蚁来执行——挖洞、寻找食物、搬运食物、清理垃圾、以及专门下蛋的蚂蚁 Queen。如果用一个网络来学上述所有行为，则需要通过全局搜索的方式学习一个很大的网络。学得智能体的行为多样性则会受限于网络的大小，过小的网络不能解决复杂的问题。与此同时，另一种方式是为每一个智能体学习一个独立的价值网络或者策略网络，这种方式也存在缺陷，因为智能体间或多或少可以通过共享个体学习到的一些知识来加速整体的学习。特别是在一些大型系统中，并不是每一个智能体都完全不同。因此，他的团队提出一种基于角色的多智能体强化学习

方法。他的基本思想是，如果智能体在任务中承担类似的角色，就分配类似的决策策略，从而共享智能体间学习的经验。而类似的决策策略意味着智能体在执行类似的子任务，它们将展现类似的行为。

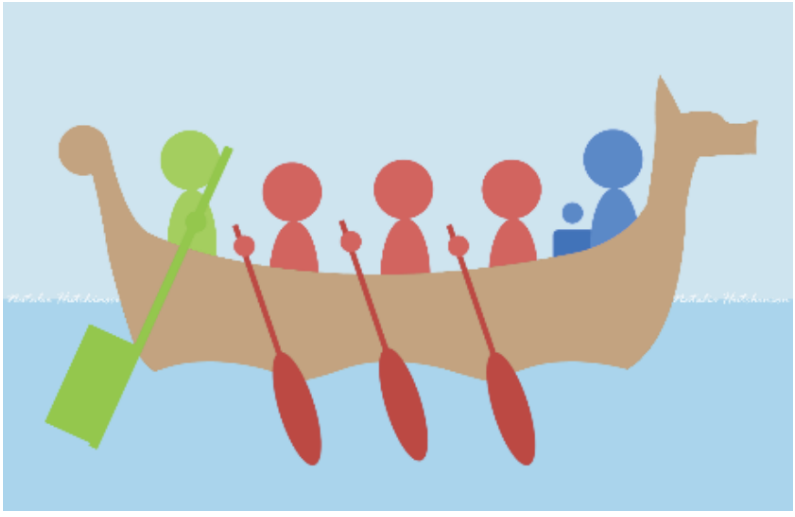


图 16：划龙舟，基于角色涌现的多智能体强化学习

可以从一个简单的示例理解动态共享参数的原因，以划龙舟为例（图 16），其中有三种角色，掌舵者，船员以及协调敲鼓者。显然，划船的船员策略高度一致，所以可以分享他们的学习经验。而掌舵者其他人策略是很不一样的，所以不一定需要有经验的分享。划龙舟的角色划分例子引入了人类的先验知识，但是解决很多问题之前，无法预先获知需要的角色，也无法划分相应的角色给不同的智能体，张崇洁的团队针对这个问题给出了一种巧妙的解决方案 ROMA（Multi-Agent Reinforcement Learning with Emerging Roles）。首先由于这些角色的非预定义性，设计的算法要能够自动学习出所需要的角色。具体的做法是通过推理的方式，根据智能体的行为推理它之前相应的角色是什么。每一个智能体的策略由他的角色所决定。如果智能体角色相同，那么它们的策略也相同； 如果角色不同，那它们的策略也会不同，这样就达到动态共享的一种方式。当然，智能体的角色不是一成不变的，一个智能体可以根据环境的不同来动态改变它的角色。

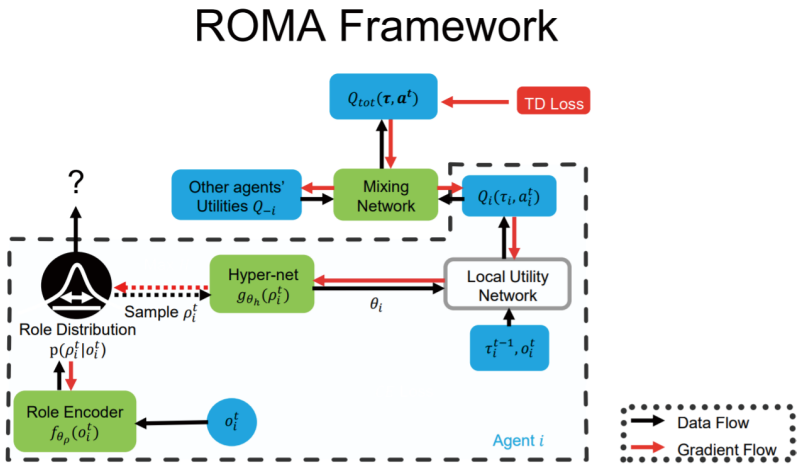


图 17：基于角色的多智能体学习框架 ROMA

ROMA 的框架如图 17 所示，在执行过程中，智能体 i 通过一个编码器把它的观测编码到一个随机的隐空间中，然后在这个角色的隐空间中进行采样，采样出当前智能体需要的一个角色。通过 Role Decoder 解码器（一种超网络 hyper network），解码输出智能体的局部效用函数的参数。该效用函数的输入为历史信息和当前智能体的观测，输出为该智能体的值函数 Q value，每一个角色都对应着智能体的每一种策略。训练过程中，每一个智能体把他们的局部值函数 Q value 输入到一个混合网络中（Mixing Network），该混合网络输出全局联合价值（Total Q value），这样可以通过时间差分 TD 的方式进行训练。

- We propose two regularizers to enable roles learning
- Identifiable by its behaviors (trajectory, role, observation)
  - Maximizing mutual information $I(\tau_i; \rho_i | o_i)$
- Capable of clustering agents
  - Either agents have similar roles;
  - Or they have different behaviors, which are characterized by the local observation-action history.

图 18：角色学习中的两个正则器 (Regularizer)

但是简单按照上述的做法很难学到较优的策略，为了保证学得的角色在空间上是有意义的，需要施加一定的约束条件。具体包括两个方面，一是角色跟它的行为的相对应性，一个角色对应某一类的行为，通过最大化角色在隐空间中的参数以及角色的经历（trajectory）二者的互信息实现；二是角色可以区分不同智能体的行为，如果智能体的行为类似，应该将它们的角色在隐空间中聚在一起，需要分化聚类的功能，不同智能体要么承担相同的角色，要么具有不同的行为。

- To formalize this idea
  - Introduce a learnable dissimilarity model d_ϕ
  - For agents, i and j, seek to maximize $I(\tau_j; \rho_i | o_j) + d_\phi(\tau_i, \tau_j)$
  - Minimizing the number of roles by minimizing $\|D_\phi\|_{2,0}$, the number of non-zero elements in $D_\phi = (d_{ij})$, where $d_{ij} = d_\phi(\tau_i, \tau_j)$
- The clustering loss:

$$\mathcal{L}_D(\theta_\rho, \xi, \phi) = \mathbb{E}_{(\boldsymbol{\tau}^{t-1}, \boldsymbol{o}^t) \sim D, \boldsymbol{\rho}^t \sim \rho(\cdot | \boldsymbol{o}^t)} \left[\left\| D_\phi^t \right\|_F - \sum_{i \neq j} \min\{ q_\xi(\rho_i^t | \tau_j^{t-1}, o_i^t) + d_\phi(\tau_i^{t-1}, \tau_j^{t-1}), U \} \right]$$

图 19：角色可聚类的损失函数

因此，整个损失函数由三方面组成，一是训练混合网络的 TD Loss，二是可区分角色的 Identifiability loss，三是不同智能体聚类的 Clusterability loss。

$$\mathcal{L}(\theta) = \underbrace{\mathcal{L}_{TD}(\theta)}_{\text{TD Loss}} + \lambda_I \underbrace{\mathcal{L}_I(\theta_\rho, \xi)}_{\text{Identifiability}} + \lambda_D \underbrace{\mathcal{L}_D(\theta_\rho, \xi, \phi)}_{\text{Clusterability}}$$

图 20：ROMA 整体的优化目标

实验结果表明，在具有挑战性的星际争霸 2 微操作管理任务中，ROMA 在越复杂的环境下表现地越突出，例如，在 27 vs. 30 个海军陆战队的场景中，智能体需要具有非常好的微操作才能以少数赢多数。在比赛初期，智能体大致采用相同的角色分布，而后期会根据血量的多少来承担不同的角色。

智能体的角色是动态的，在初期智能体会根据所处的位置选取角色，比如站在前、中、后的智能体将会具有不同的角色。角色相近的智能体随着打斗的过程中血量的变化承担新的角色，例如血多的向前冲，血少的向后退。

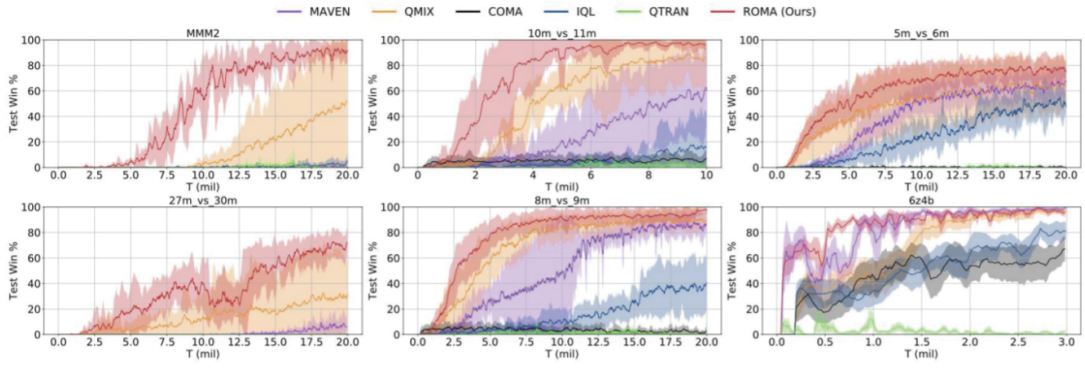


图 21 ROMA 在星际争霸 II 微操作管理挑战上的实验结果（https://sites.google.com/view/romarl）

图 21 和 22 展示了 ROMA 在星际争霸 II 微操作管理挑战上的实验结果，学习阶段初期时，智能体在探索不同的角色，随着学习越来越深入，角色功能的分化也会越来越好。因此角色和学习的过程是一个相互反馈的交互过程。上面是同构的情况，在异构的情况下也是相同，能够学出不同的角色，智能体的功能不同，角色也会不同。

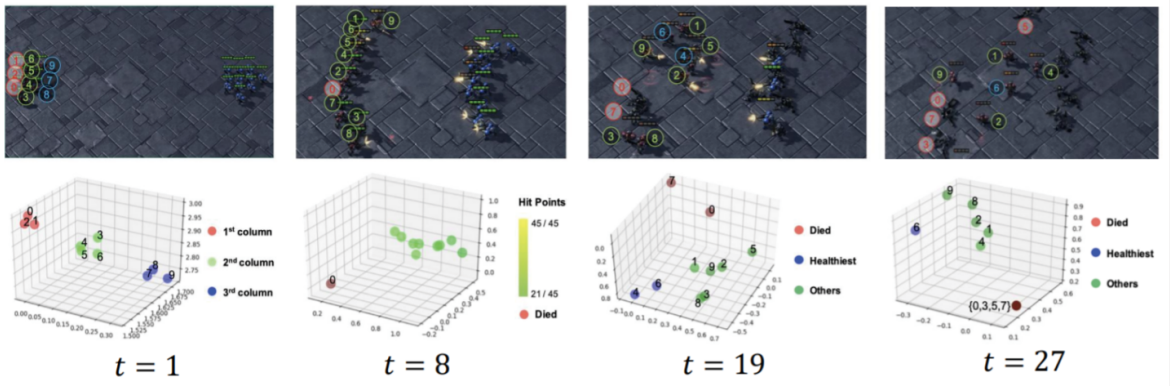
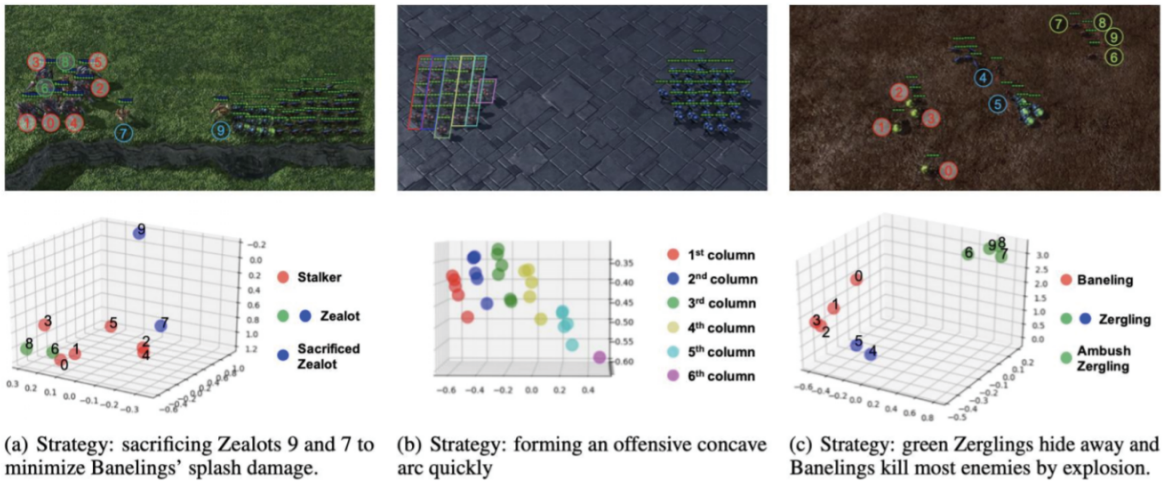


图 22：动态角色学习图解



(a) Strategy: sacrificing Zealots 9 and 7 to minimize Banelings' splash damage.

(b) Strategy: forming an offensive concave arc quickly

(c) Strategy: green Zerglings hide away and Banelings kill most enemies by explosion.

图 23：几个 ROMA 角色学习中令人兴奋的示例

总结前两项工作在多智能体强化学习研究中的位置，如图 24 所示，协作式多智能体主要由两个方面刻画。

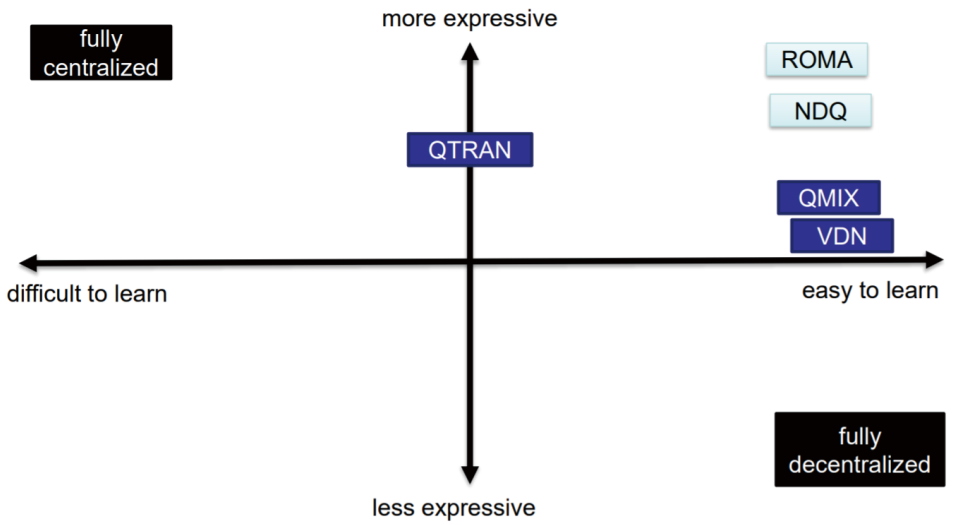


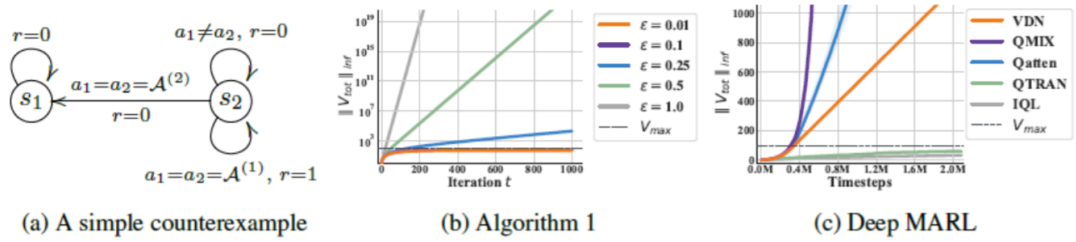
图 24：协作式多智能体研究工作总览

一个是方法学习的难易度；另一个是方法学习得到的效果，即表达最优策略的效果。显然完全中心化的学习方式（Fully centralized），具有很强的表达能力，但同时较难学习。而完全分布式的学习方式（Fully decentralized），具有较弱的表达能力，但是较为容易学习。在它们中间值函数分解（Value factorization）的方法，具有折中的表达能力以及中等的学习难度，从表现上讲相较于完全分布式的学习方式要好得多。上述介绍的接近可分解的值函数算法 NDQ 通过增加通信的方式具有更强的表达能力，而学习的难度相较于值函数分解的算法（如 QMIX）也相差无几。同时，基于角色的多智能体学习 ROMA 算法则有更多的多样性。

四、线性值函数分解的理论分析工作以及 Offline RL[7]

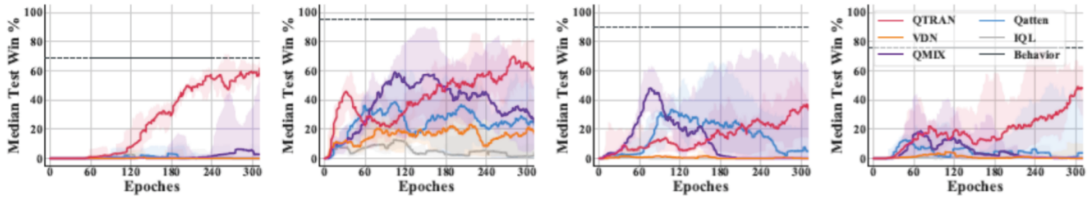
最后一节介绍张崇洁团队在线性值函数分解方面的理论分析工作，在该工作中意图解决的问题是：为什么多

智能体学习值函数的分解的算法（如 VDN,QMIX）能够取得不错的效果？在该工作中，他们设计了一个 Multi-agent Fitted Q-iteration 的理论框架，将单智能中研究函数近似（Function Approximation）下算法的收敛性以及最优性的表现的 Fitted Q-iteration 框架扩展到了多智能体领域（Multi-agent fitted Q-iteration）。从中推导出了闭式的更新规则（closed-form update rule），即 Empirical Bellman error minimization，基于这个闭式解，他们发现简单的线性值函数分解方法（linear value factorization），例如 VDN 中的局部值函数求和得到联合值函数，基于这样简单的结构假设以及混合网络的训练，它们隐式地实现了一个非常好的信度分配（credit assignment），一种联合报酬分配的机制，称为反事实的报酬分配。具体来说，单个智能体自己的报酬，等价于假想做随机的动作得到的整体的报酬，与做最优的策略行为得到的报酬之间的差值，将这个作为行为值函数 Q value，是一种非常好的方式。在现实生活中，如公司评价一个员工的贡献也是类似的做法，由此解释了值函数分解方法在多智能体学习的任务中具有较好表现的原因，同时该工作也证明了在智能体 on-policy 训练过程中的局部收敛性。



(a) A simple counterexample　(b) Algorithm 1　(c) Deep MARL

图 25：反例的构造与收敛性实验



(a) 5m_vs_6m　(b) 2s3z　(c) 1c3s5z　(d) 2c_vs_64zg

图 26：值函数分解方法及分布式训练方法 IQL 在离线训练中的实验

尽管有特定条件下局部收敛性的证明（图 25），但是目前的工作缺乏全局的收敛性保证。此外，而且在采用不同的策略得到的数据训练（Off-policy）时，线性值函数分解算法表现的不尽如人意，无法从示例中进行学习（learn from demonstrations）。如在图 25c 的离线训练实验中，当前许多最好的算法的价值函数都无法收敛，会发散。在复杂的环境下（图 26），如果 data 先通过一个好的策略收集起来（实验采用 QMIX 收集），用收集的数据来进行训练的方法也没有效果，即使采用 QMIX 的得到数据训练同类 QMIX 智能体，也无法得到很好的表现，这与常识中单智能体 Q-learning 是一个离线策略训练的观念相违背，非常值得进一步的探索。

五、结语

多智能体强化学习是一个具有很大前景的研究领域，本次演讲中，张崇洁从协作式多智能体的角度分享了前沿的研究工作，包括通过值分解的方法与通讯优化方式结合实现多智能体合作，有效地解决多智能体任务中的不确定性问题；通过基于角色学习的方法，通过动态共享智能体的参数提升可扩展性，使得在非常困难的任务中

相对于最好的方法也有极大的性能提升。谈到领域未来的发展方向，他指出在环境允许的条件下，支持分层的学习具有较大的前景。同时，针对线性值函数分解算法的理论分析工作，表明了当前算法对于离线训练(off-policy training)需要有较大的关注。离线训练不仅具有较高的学习效率，同时可以利用已有的示例数据，如自动驾驶中的经验数据，来进行策略的学习。未来离线多智能体强化学习将会成为重要的组成部分。

参考资料

[1] J. N. Foerster, "Deep Multi-Agent Reinforcement Learning," p. 205.

[2] P. Sunehag et al., "Value-Decomposition Networks For Cooperative Multi-Agent Learning," arXiv:1706.05296 [cs], Jun. 2017, Accessed: May 29, 2020. [Online]. Available: http://arxiv.org/abs/1706.05296.

[3] T. Rashid, M. Samvelyan, C. S. de Witt, G. Farquhar, J. Foerster, and S. Whiteson, "QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning," arXiv:1803.11485 [cs, stat], Jun. 2018, Accessed: Mar. 13, 2020. [Online]. Available: http://arxiv.org/abs/1803.11485.

[4] T. Wang, J. Wang, C. Zheng, and C. Zhang, "Learning Nearly Decomposable Value Functions Via Communication Minimization," arXiv:1910.05366 [cs, stat], Oct. 2019, Accessed: May 23, 2020. [Online]. Available: http://arxiv.org/abs/1910.05366.

[5] A. Das et al., "TarMAC: Targeted Multi-Agent Communication," arXiv:1810.11187 [cs, stat], Feb. 2020, Accessed: May 25, 2020. [Online]. Available: http://arxiv.org/abs/1810.11187.

[6] T. Wang, H. Dong, V. Lesser, and C. Zhang, "ROMA: Multi-Agent Reinforcement Learning with Emergent Roles," arXiv:2003.08039 [cs], Mar. 2020, Accessed: May 08, 2020. [Online]. Available: http://arxiv.org/abs/2003.08039.

[7] J. Wang, Z. Ren, B. Han, and C. Zhang, "Towards Understanding Linear Value Decomposition in Cooperative Multi-Agent Q-Learning," arXiv:2006.00587 [cs, stat], May 2020, Accessed: Jun. 15, 2020. [Online]. Available: http://arxiv.org/abs/2006.00587.

清华交叉信息学院吴翼：多智能体强化学习中的课程学习、演化与复杂性涌现

整理：智源社区 熊宇轩

作为 ACM–ICPC 领域的传奇人物之一，昔日的「姚班」少年吴翼在加州大学伯克利分校取得博士学位之后，加入了 OpenAI 从事通用人工智能研究。近年来，他发表了以 MADDPG 为代表的一系列高水平研究成果。在本届智源大会上，吴翼博士带来了以「多智能体强化学习中的课程学习、演化与复杂性涌现」(Curriculum,Evolution and Emergent Complexity) 为题的主题演讲。吴翼博士从哲学的终极命题「我们从哪里来?」出发，介绍了涌现出复杂群群体行为的条件，并结合 OpenAI 近期完成的捉迷藏游戏项目进行了详细的说明。

下面是演讲正文，智源社区编辑做了一定的编辑整理。

一、物种进化的启示

本次演讲，我主要针对人工智能研究领域最重要的任务之一——「构建通用人工智能」进行讨论。为了实现这一最终的目标，我们已经在深度学习、强化学习、规划 / 推理、搜索等方面取得了许多令人备受鼓舞的进展。这些伟大的技术用到了与机器学习和优化方法有关的诸多概念。在本次演讲中，吴翼博士主要从受人类自身启发的新型人工智能研究的角度进行了深入探讨。

图 1：智能的涌现——我们来自何方?

该领域的研究要从人类智能的漫长历史说起。我们美丽的地球母亲拥有着神奇的生态系统，它最终孕育了人类的生命。在现代科学技术的帮助下，我们逐渐开始构建和人类有相似行为的机器。然而，在我们创造一类新型智能体之前，我们需要回答一个问题：我们 (智慧生命) 从哪里来?

「我们从哪里来?」这是一个经久不衰的问题。200 年前，一位名叫「查尔斯・达尔文」的年轻人也对这一问题产生了兴趣。1831 年，为了探寻该问题的答案，达尔文开始了他的环球航行。大约 4–5 年后，达尔文抵达了一处被称为「加拉帕戈斯」的群岛。如今，这里已经是一个自然保护区，成为了各种生物栖息的乐园。

在此后的数月中，达尔文环绕整个群岛进行了深入的考察，并受此启发撰写出了名垂后世的巨著《物种起源》。

在达尔文的环岛旅行中，最著名的故事莫过于「达尔文雀」的演化。他注意到，尽管所有的达尔文雀都起源于大陆上，但是分布在加拉帕戈斯群岛中不同小岛上的达尔文雀演化出了形状不同的喙，这是由于它们吃的食物不同。

除了鸟类之外，我们还可以在加拉帕戈斯群岛上发现许多有趣的现象，这些物种在数百万年间涌现出来的过程会令人感到震惊。例如，那里有生活在陆地上或者生活在水中的不同种类的蜥蜴，有生活在赤道上的企鹅，还有不会飞的鸟（鸬鹚）。这正是大自然的神奇之处，它们有着出人意料的极为多样的能力和行为。

因此，加拉帕戈斯之旅巩固了达尔文脑海中伟大的理论——「进化论」。达尔文在它的巨著「物种起源」中详细介绍了「进化论」，他认为这种物种的多样性并不是刻意设计的，而是进化（自然选择）的结果。有机生命会努力适应环境的变化，而只有那些最适合的种群能够在自然选择的过程中幸存下来。这些物种本身也成为了环境变化的一部分，使得各种不同的物种能够以合作、竞争等形式协同演化。人类作为地球上唯一真正具有「智能」的生物，能够得以诞生，本身也是生命的奇迹。

那么，我们能从物种进化的过程中得到怎样的启示呢？首先，生物体循序渐进地从简单形式演化到复杂形式；其次，各物种作为环境的一部分进行交互、合作、竞争，并协同进化； 第三，地球是一个复杂系统，进化的过程存在极大的复杂性，并且会产生意想不到的结果； 最后，进化发生在大量个体组成的种群中，在一个种群中会涌现出群体行为。

简而言之，对于强化学习来说，我们可以从物种进化的过程中得到以下启示：（1）我们需要构建一个足够复杂的仿真环境；（2）我们应该让智能体在该仿真环境下协同演化。

在上述要点的启发下，我们的工作主要涉及两个方面：（1）在符合规律的物理环境下根据简单的规则涌现出复杂性；（2）涌现出群体行为，以及如何实现这种行为。吴翼博士在本次演讲中仅仅介绍了第一个部分的工作。

二、捉迷藏游戏中的智能涌现

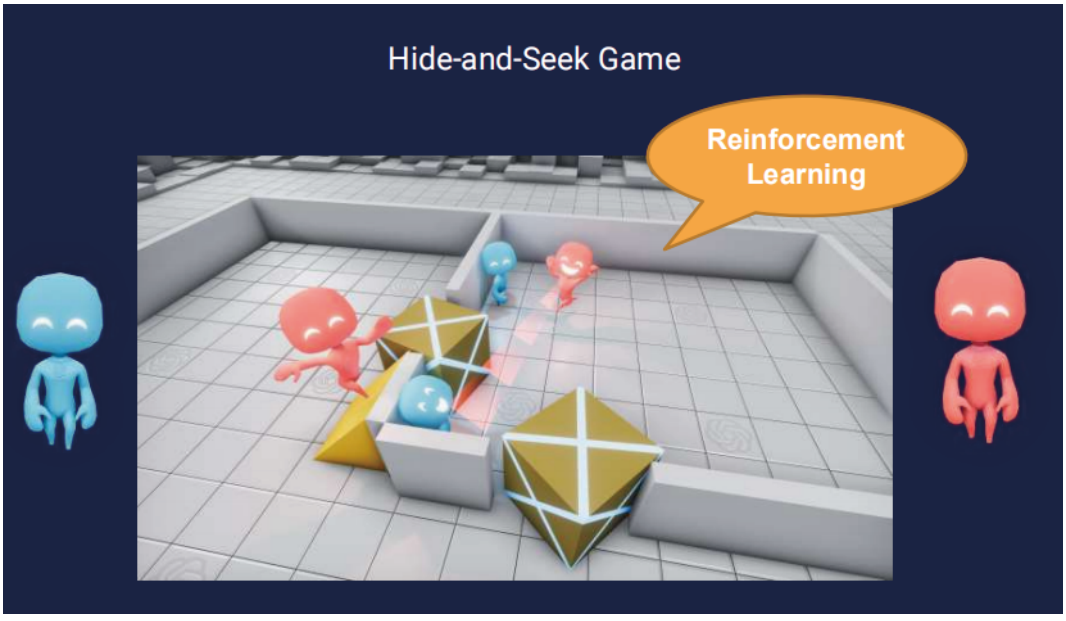


图 2：捉迷藏游戏的仿真环境

如图 2 所示，在捉迷藏游戏的仿真环境中，我们设置一些仿真物种（可爱的小智能体），还有道具物体、墙，这些智能体会在该环境中玩捉迷藏游戏。红色的智能体（seeker）需要找到蓝色的智能体（hider），hider 则需要躲起来而不被 seeker 找到。当红色的智能体找到 hider 时，seeker 会得到奖励，而 hider 则会由于被找到而受到惩罚。因此，通过执行强化学习让这些智能体能够协同学习，从而优化他们的奖励函数。

为了模拟自然环境下的演化过程，我们创建了数以千计的捉迷藏游戏，让智能体在仿真的物理世界中并行运行游戏。仿真演化实验的技术细节如下:

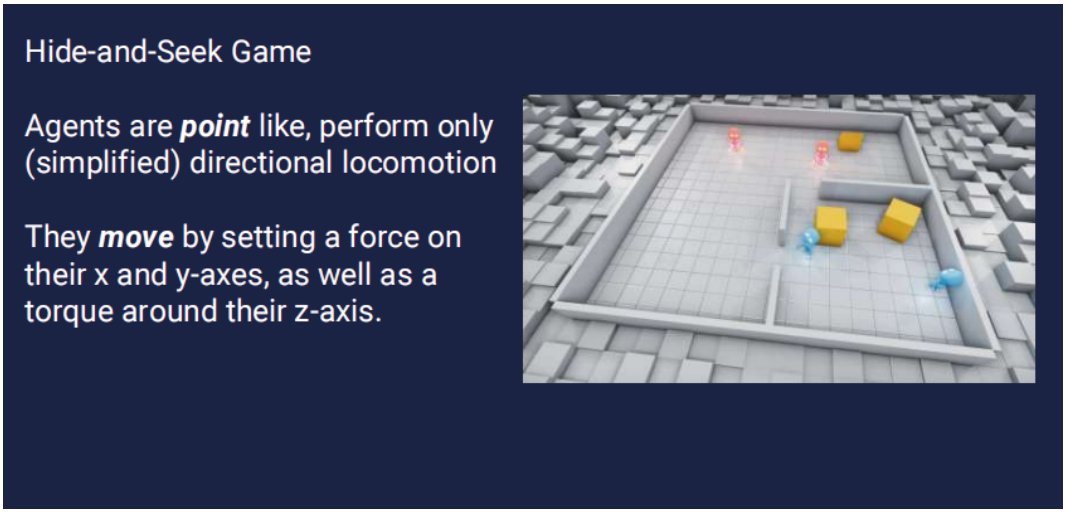


图 3：捉迷藏游戏的技术细节

为了简化控制过程，我们将智能体视为一个个点，它们只执行简化的定向运动。通过对智能体施加一个 x 和 y 轴上的力，以及 z 轴上的扭矩（让智能体旋转），使智能体移动。

图 4：智能体的视线

智能体可以看见正面的圆锥形视线（如图 4 中 seeker 正面的红色区域）中的物体和智能体，它只能接收这些可见物体和智能体的信息。

图 5：智能体的感知能力

智能体也可以通过类似于激光雷达的传感器感知到它们与物体、墙，以及其它周围的智能体之间的距离。因此，它们可以通过感知门和墙壁在该环境中进行导航。

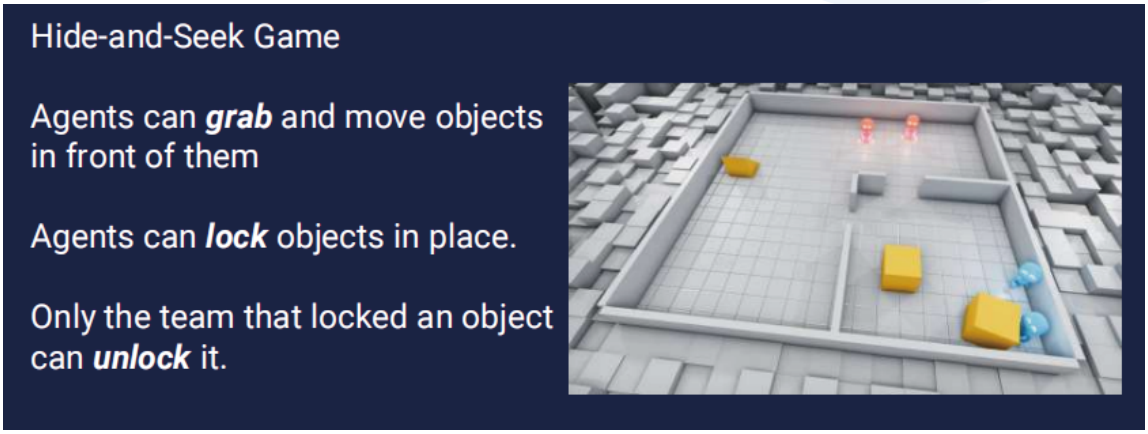


图 6：智能体的特殊行为

除了移动之外，智能体还能完成两种特殊的行为：（1）抓取（2）上锁 / 解锁。当智能体执行抓取行为时，它们可以移动面前的物体，并一直持有这些物体。当智能体执行上锁动作时，它们可以将面前的物体锁定在某个地方。请注意，我们进行的是有组别的锁定动作，如果 seeker 将某物体上锁，那么只有 seeker 能够将该物体解锁；反之，如果 hider 将某物体上锁，那么只有 hider 能够将该物体解锁。

此外，我们还设计了一些额外的细节。例如，设计了一种基于团队的奖励函数，来自同一个团队的智能体会得到同样的奖励。如果某 seeker 发现了任意的 hider，则红队奖励函数 +1，反之则 −1；如果某 hider 没有被发现，则蓝队奖励函数 +1，反之则 −1。

同时，我们还为 hider 设计了一个准备阶段，在此期间 seeker 不能移动，所有的智能体也不会得到奖励。在训练过程中，我们希望在随机环境下，通过并行的近端策略优化算法进行训练。我们将在并行的环境中随机初始化物体的位置、墙的结构，以及智能体的数量。在这里，我们也使用了参数共享机制，每一个智能体会共享相同的策略权值。我们向策略的输入中引入了智能体的类别信息，每个智能体知道它们是 seeker 还是 hider。在明确了这些技术细节之后，我们接下来将讨论涌现出的行为。

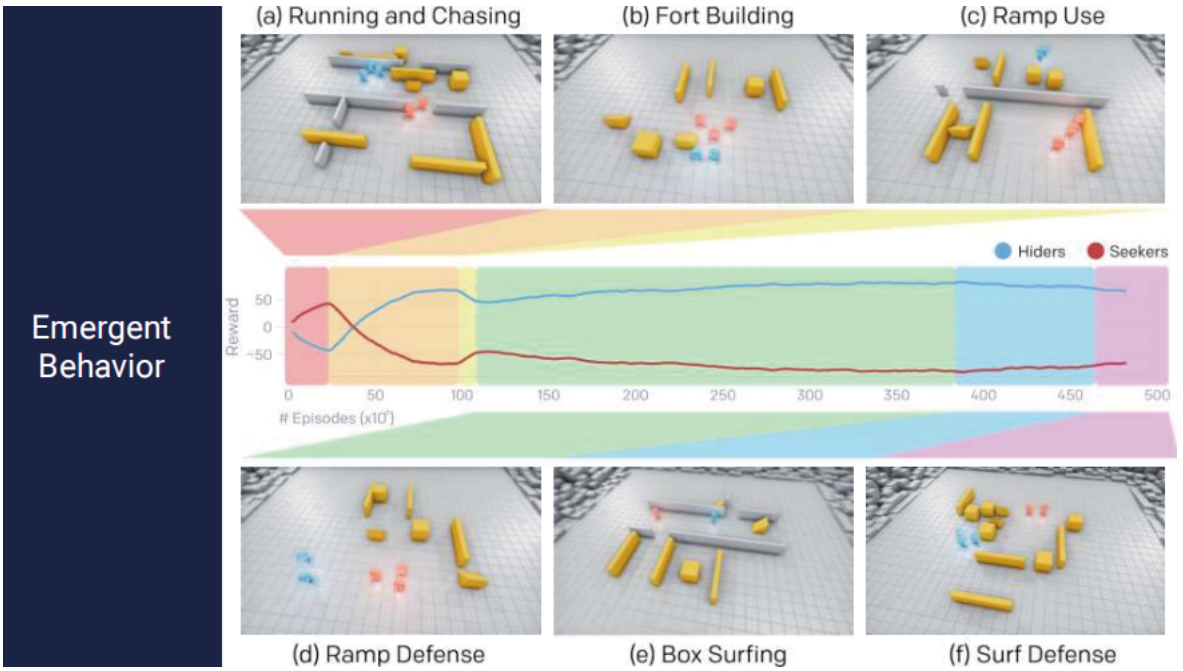


图 7：涌现出的行为——逃跑和追赶、构建堡垒、使用扶梯、防御扶梯攻击、箱子冲浪

如图 7 所示，我们一共总结出了 6 种涌现出的行为阶段，我们也展示出了各个智能体的奖励函数曲线:

阶段 1「逃跑和追赶」: 在训练初期，智能体基本上学习到的行为是「逃跑和追赶」，seeker 试图追赶上 hider，而 hider 试图逃跑。

阶段 2「构建堡垒」: 在本阶段中，hider 学会了在准备阶段使用环境中的箱子在它们自己周围搭建一个堡垒，这样一来，seeker 在准备阶段结束后就无法直接看到 hider。

阶段 3「使用扶梯」: 随着训练继续进行，seeker 很快就发现了一种对策，它们学会了将扶梯移动到 hider 构建的堡垒旁，然后爬上扶梯跳入堡垒，从而发现 hider。

阶段 4「防御扶梯攻击」: 在经过了长时间的训练后，hider 也会针对 seeker 的扶梯攻击找到相应的对策。Hider 学会了在准备阶段构建堡垒之前就锁上所有的扶梯，这样一来 seeker 就无法借助扶梯跳入堡垒了。我们曾经认为这就是群体智能涌现的最终阶段了，然而正如我们在视频中所展示的，在该阶段之后又发生了一些意想不到的现象。

阶段 5「箱子冲浪」: seeker 学会了将箱子移动到被锁定的扶梯旁，然后通过扶梯爬上箱子，站在箱子顶上向堡垒「冲浪」，最终跳入堡垒发现 hider。这是由于我们构建环境中的物理模型的方式所导致的（尤其是抓取机制）——当智能体抓取住某物体后，这二者之间的距离就固定下来了，因此当智能体站到箱子上之后，它可以同时执行抓取和移动两种行为，从而使箱子和智能体一起移动（即「冲浪」）。这并不是一种漏洞，而是该环境的特性，智能体只是学会了利用这种特性。

阶段 6「防御冲浪攻击」: 最后，hider 学会了在准备阶段将包括箱子、扶梯在内的所有物体都锁上，从而防御箱子冲浪攻击。这也正是我们观测到的最后的阶段。

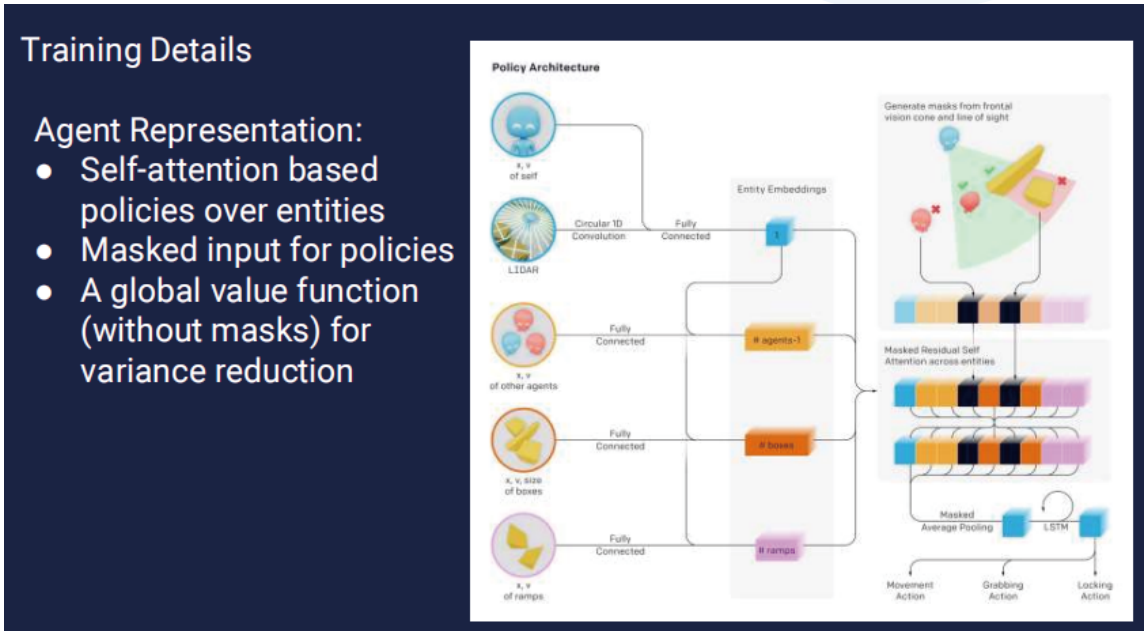


图 8：训练细节

下面，我们将讨论训练的细节。实际上，我们进行了大量的模型简化实验（又称「消融实验」），如果你想了解全面的训练细节，请参阅论文「Emergent Tool Use From Multi-Agent Autocurricula」(https://arxiv.org/abs/1909.07528)。

我们在实验中也观测到两个重要事实。

首先，对于智能体的表征是十分关键的。由于实验中会涉及到随机数量的物体，我们对各个实体使用了基于自注意力机制的策略。接着，我们通过掩模屏蔽了一些策略的输入，从而使智能体只能获取它们可以看见的物体的信息。对于值函数而言，我们使用了一种不带有掩模的全局值函数，从而减小了方差。

图 9：训练细节——训练规模

此外，我们还观察到训练规模也有很大的影响。我们使用了不同的 Batch size 进行模型简化实验，我们发现当 Batch size 较大时，训练速度会快很多。如图 9 所示，蓝色的部分代表我们采样得到的所有用于训练智能体的状态序列，黄色部分是训练时间。显然，随着 Batch size 增大，训练时间会缩短。

除了这六个智能涌现阶段，我们还有很多有趣的实验发现。首先，我们发现智能体可以学习到很棒的协作策略。例如，我们观察到智能体会相互传递持有的箱子，从而更高效地堵住门。在我们的实验中，也出现了一些意想不到的行为，这与地球上物种的自然进化十分相似。例如，在训练的初期，hider 实际上仅仅学会了不断逃向距离 seeker 无限远的地方，而这种行为会阻碍其它有趣的行为的涌现。因此，在最终的环境实现方案中，我们在围绕游戏场地中心的一定范围内设定了一个惩罚区域，这样一来就不会有智能体试图逃向无限远处。

实际上，智能体是非常聪明的，他们学会了利用环境中的一些漏洞。例如，hider 学会了通过非常明智的方式利用环境中的物理机制，这是由于在训练初期物理引擎中的某些设置情况造成的，我们需要在最终的环境实现中修正这些漏洞。同样地，seeker 也会利用一些环境的漏洞。例如，seeker 发现了它们可以利用扶梯爬到某些特定位置，然后飞到空中。这也是由于物理设定造成的，我们需要仔细地修正这些漏洞。

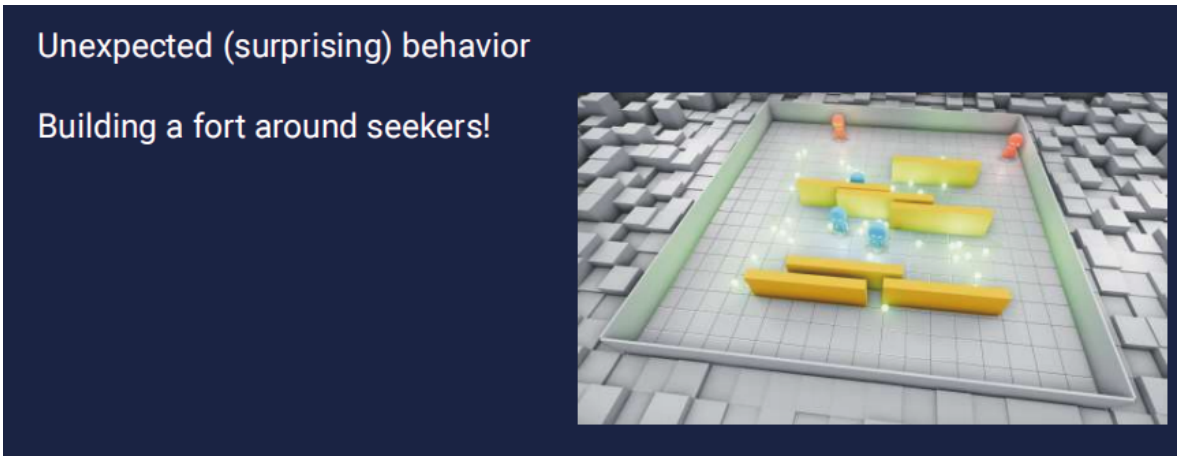


图 10：hider 构建一个堡垒将 seeker 围住

另一个令人意想不到的行为是，由于 hider 可以构建堡垒将它们自己包围起来，那么它们是否能构建堡垒将 seeker 围起来，从而让 seeker 无法出来抓 hider 呢？这是一种非常自然的策略，但是在我们标准的捉迷藏游戏中并没有涌现出来。但是在一些变体中，这种策略确实发生了。

在如图 10 所示的变体游戏中，绿色的发光点是一些奖励金币。Hider 不仅仅要防止它们自己被 seeker 找到，还要保护这些金币不被 seeker 捡到。因此，在这种情况下，仅仅构建一个堡垒将 hider 自己围起来并不足以保护这些金币。而这些 hider 非常聪明，它们找到了正确的应对策略：在准备阶段将 seeker 推到角落，然后使用箱子将这些 seeker 堵在这些角落中。有趣的是，hider 甚至使用了两层箱子加固这个陷阱，这种行为是非常出乎我们的意料的。这种变体是由吴翼博士本人发现的，这也是他本人最喜欢的整个项目中的一段视频。

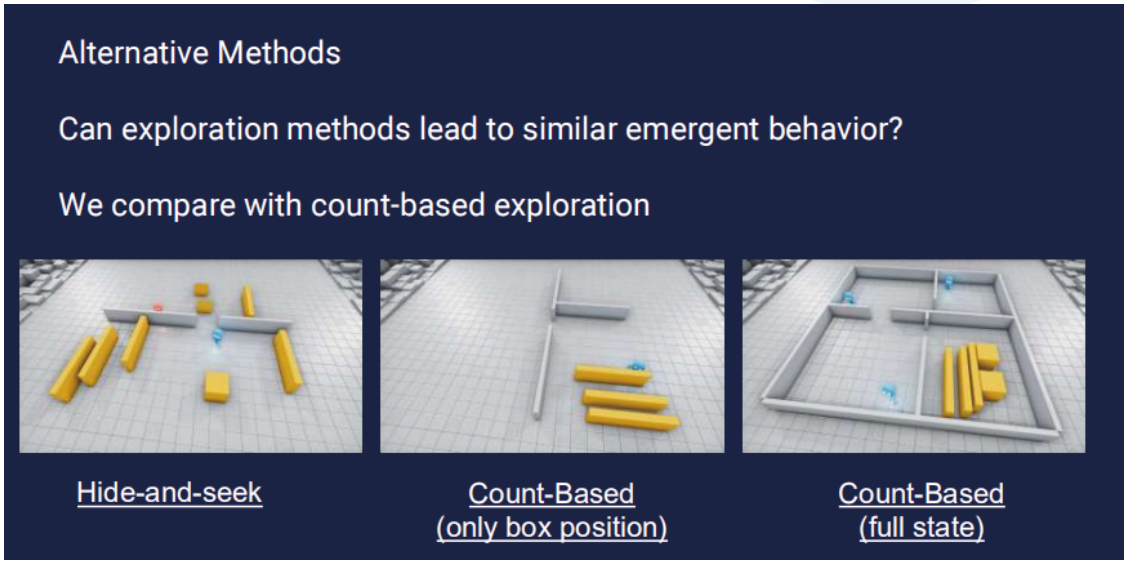


图 11：对比实验

那么，其它的方法也能够得到与我们相似的行为涌现结果吗？为此，我们也和其它方法进行了对比实验。具体而言，我们通过一种常用的「基于计数的探索」(count-based exploration) 方法进行了一系列对比实验。我们的多智能体强化学习框架发现的行为如图 11 最左侧的图所示。我们可以很容易地描述不同行为的强度，这些策略对于人类而言也是可解释的，很容易理解。

然而，在基于技术的探索中，我们可以观察到此时确实涌现出了某些行为。例如，智能体抓取到某物体，然后在环境中移动。但是此时出现的行为是比较低级的，即我们无法看到任何策略性的行为。

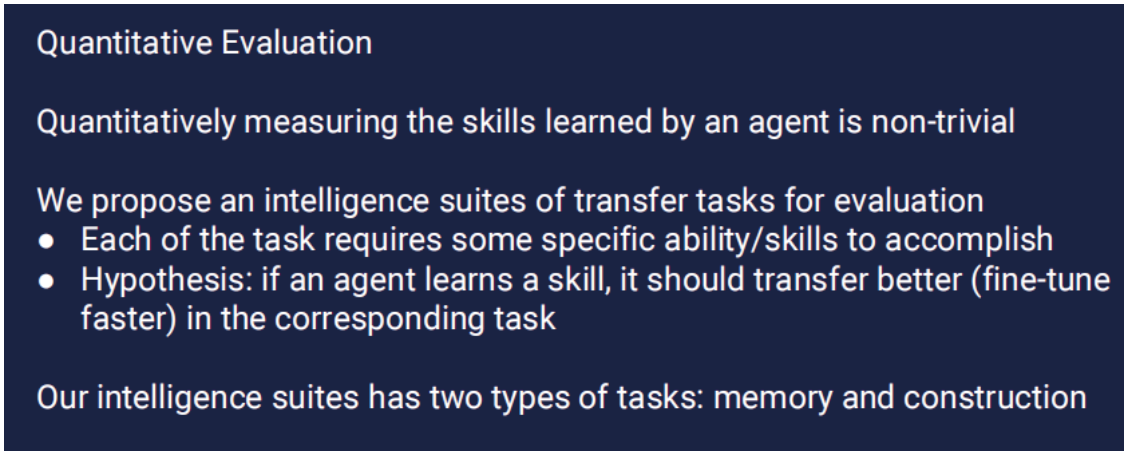


图 12：用于定量评估的迁移学习任务

除了观察涌现出的行为之外，我们执行了一些迁移学习任务，用于定量评估。由于捉迷藏游戏中的智能体学习到了一些复杂的技能（例如，导航、操作物体，建造等），那么我们能将这些技能迁移到其它的环境中吗？

为了实现这一目标，我们提出了一套由 5 个迁移学习任务组成的智能实验，将其用于定量评估。为了完成每项任务，都需要智能体具备某些特定的技能。我们假设，如果某个智能体学会了某种技能，那么它应该在相应的任务中能够更好地迁移这种技能（相对于从头开始训练调优更快）。我们可以把这 5 个迁移学习任务分成两类：(1) 记忆 (2) 构建。

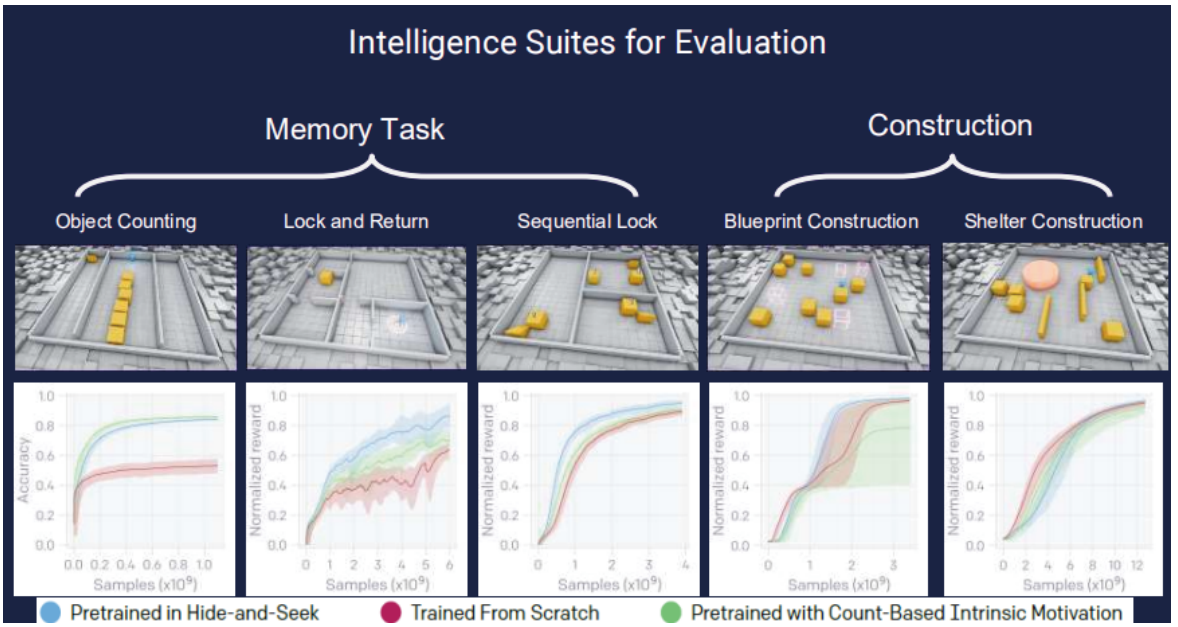


图 13：5 类迁移任务实验

5 类迁移任务如图 13 所示。其中，「物体计数」、「上锁并返回」、「顺序锁定」为记忆任务。为了完成记忆任务，智能体需要记住从环境中发现的关键信息。此外，「蓝图构建」和「遮蔽物构建」为构建任务。为了完成建造任务，智能体需要以一种特殊的方式操纵大量的物体。

任务 1「物体计数」：在该任务中，智能体不能移动。环境中有一些数量随机的箱子，这些箱子会随机向左侧或右侧移动。智能体需要观察这些箱子的移动情况，并且在状态序列 (episode) 的最后回答出有多少箱子移动到了左侧，有多少箱子移动到了右侧。因此，智能体需要记住这些箱子的移动过程。
任务 2「上锁并返回」：在环境中存在一个箱子，智能体需要找到这个箱子，将其上锁，并且自己返回到出生点。因此，智能体需要记住自己的出生点信息，而这种信息并不是输入的一部分。
任务 3「顺序锁定」：这是最具挑战的一类任务，环境中分布着若干箱子，智能体需要以特定的顺序锁定这些箱子，然而智能体事先并不知道这种顺序。因此，智能体首先需要对环境进行探索，然后记住箱子的状态，通过不断地试错最终完成任务。因此这是一种典型的长期收益 (long horizon) 任务。
任务 4「蓝图构建」：在该任务中，给定一个蓝图 (blueprint)，智能体需要将所有箱子移动到蓝图中规定的期望地点才能完成任务。
任务 5「遮蔽物构建」：在环境中有一个大的红色圆柱体，智能体需要用箱子将这个圆柱体包围起来，保护它不被环境所发现。

在实验中，我们在这 5 个迁移任务中执行了近端策略优化 (PPO) 算法，采用不同的初始化策略时的学习曲线如

图 13 下方所示。蓝色的曲线为使用我们的捉迷藏游戏初始化的策略的学习曲线，红色的曲线为从头开始训练的学习曲线，绿色曲线代表受到通过基于计数的方法启发的预训练方法。在所有的迁移任务中，捉迷藏游戏都要优于对比基线或与其性能相当。在记忆任务中，这种优势要更为普遍，尤其是对于那些需要导航技能的任务来说。但是，对于构建任务来说，这种差别就非常小了。我们至今还不明白为什么这种差距如此之小，这也是一个有待研究社区解决的开放性问题。

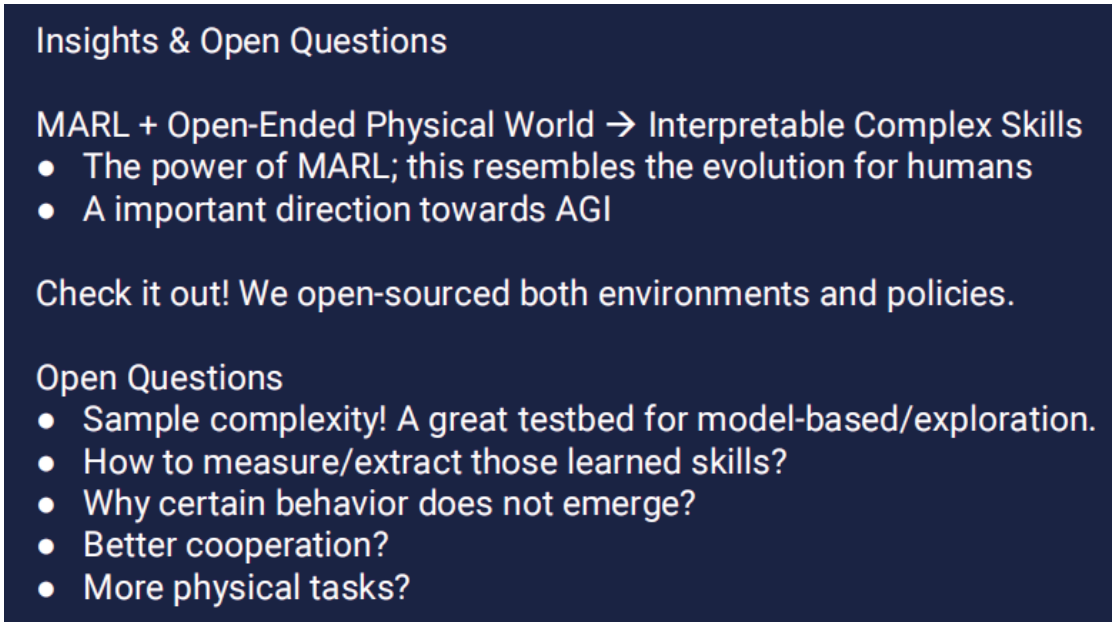


图 14：见解与开放性问题

那么，我们可以从捉迷藏游戏项目中学到什么呢？最重要的一点是，通过将多智能体强化学习与开放式的物理世界相结合，会自然地涌现出一些人类能够理解的复杂技能。这归功于多智能体强化学习的强大能力，而这种训练过程与人类的进化过程非常相似。我们相信，这是通向通用人工智能之路上的重要研究方向。该项目的环境与策略已经开源，感兴趣的读者可以通过以下链接获取相关资源：https://github.com/openai/multi-agent-emergence-environments

作为通向通用人工智能之路的阶梯，本项目也提出了许多开放性问题。例如，「我们是否能够减小采样复杂度」、「如何测量、提取出学习到的技能」、「为什么会涌现出某些行为，而为什么有些行为则没有涌现出来」。我们希望这些见解与开放性问题可以作为一个起点，帮助整个研究社区未来在该领域取得更大的进展。

三、结语

本次演讲的主题为「课程学习、演化与复杂性涌现」，我们认为，在演化机制与复杂环境的共同作用下，我们会发现涌现出了许多复杂的行为，而这些行为甚至对于人类来说也是可解释的。多智能体强化学习为学习复杂的行为提供了一种天然的自主课程，而这也与自然界中的演化过程相类似。在我看来，即使是对于现实生活或理解我们人类本身来说，这也是一个非常重要的研究领域。目前，这是一个非常活跃的使用强化学习技术理解人类社会的研究领域。

然而，「计算」是该研究领域中目前遇到的最大的瓶颈。我们必须承认，在这一项目中，我们使用了大量的计算资源。可能有人会认为，这是一件非常糟糕的事情，他们认为 OpenAI 过于依赖巨大的算力。但是，在我看来，这并不是一件坏事，因为这里的庞大计算量实际上意味着在未来还有巨大的提升空间。我相信，这个项目会成为该领域的一个对比基准，帮助那些激动人心的研究涌现出来。

最后，我想说，实现通用人工智能（AGI）是 OpenAI 的使命，我们对我们所坚信的道路非常有信心。我们希望，通用人工智能有朝一日能够真正得以实现，并造福于整个人类社会！

北京大学卢宗青：多智能体合作中的通信

整理：北京大学 姜杰川

第二届北京智源大会上，北京大学计算机系卢宗青教授做了《通信驱动的合作学习》的报告。

近年来，强化学习在理论和应用层面都有了很大的突破，但是在多智能体的情景下表现却不尽人意。自然界中的生物和人类社会都广泛地采用通信作为合作方式，这启发我们可以将通信引入多智能体合作算法。围绕通信的必要性和影响等方面，卢宗青教授分享了他们团队近年来的一些工作，包括通信如何帮助优化目标的提升和通信在多智能体情景中的应用。

以下为卢宗青教授的演讲正文:

一、通信作为合作方式：从自然界到人类社会

在自然界中存在着广泛的合作现象。例如在 Wood Wide Web 中，树根通过菌丝连在一起。森林中比较高的树，光合作用比较强，因此可以合成更多的养分，它可以通过菌丝把多余的养分传给旁边的小树苗，而一些将要死掉的树也会把养分传送给旁边的树。此外，当有害虫侵蚀植物时，它们通过树根之间的连接传递一些化学信号，提醒旁边的植物有害虫出现，这样这样旁边的树可以提前分泌特定的化学物质，从而抵御害虫。

另一个案例发表在 2013 年 Nature 上，研究者们发现在红海海底，石斑鱼可以与海鳗协同捕鱼。石斑鱼向海鳗摇摇头，然后一起游向珊瑚礁。海鳗身体比较柔软，可以进入珊瑚礁里边，而石斑鱼在外面等待。石斑鱼可以用头给海鳗提醒这个鱼藏匿的位置，从珊瑚礁里边被海鳗赶出来的鱼也会被石斑鱼捕食。

灵长类动物的合作就更高级了。在 Social Moneys 的例子中，猴子们通过不同的声音来提醒族群中其他猴子附近有捕食者。比如一只猴子看到了一条蛇，它就会发出声音提醒大家附近有蛇。

总结这三个例子，植物是通过分享一些化学物质来进行合作，鱼是通过一些特定的行为姿势，猴子是通过不同的声音。对于经过几千年进化的人类来说，我们的合作利用了所有这些方式，除此之外我们还进化出了更高级别的语言。这些合作方式可以概括为 Communication，这也是我今天要讲的的主题——研究如何通过 Communication 来更好地促进智能体之间的合作。

对于 Communication，学界的研究主要有两个方向，一个方向是 Information Sharing，智能体学习分享有助于算法训练和协作决策的表示或者其他特殊信息； 另一个方向是 Grounded Language，主要探究智能体之间所学会的由离散信号构成的语言是否与人类语言具有某种相似的特性。在这个报告中，我主要关注第一方面的 Information Sharing，重点介绍我们最近的一些工作。前两个工作主要是研究通信如何服务于强化学习的优化目标，后两个是研究如何通过简单的通信来使智能体更好地合作。

二、通信服务于强化学习优化的目标

通信的直观理解，是每个智能体都把自己的一些信息分享给其他的智能体。但是这样存在一些问题，首先是通

信代价比较大。另外简单的分享信息未必可以得到更好的效果，我们真正需要的是“必要的通信”，即能够对奖励带来增益的通信，这也是我们提出 ATOC 的初衷。

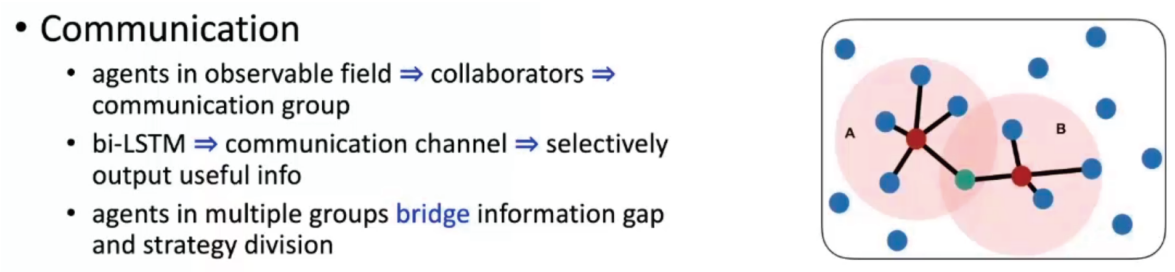


图 1： ATOC

ATOC 中的通信采用比较传统的设定，智能体可以与视野内所有智能体进行通信。通信的具体形式是，通过一个双向 LSTM 对信息进行整合。而何时与视野内智能体发起通信是通过一个门控机制来控制的。这个门控机制的学习采用的是反事实推断的思路，利用 Critic 计算通信或不通信所做出决策的 Q 值之差，差值越高说明通信对于奖励的增益越大，也就意味着越必要。将这个差值作为监督的信号训练门控机制，可以衡量通信在什么时候是有必要的。

图 3 中展示的是 Cooperative Navigation 实验中通信情况随时间的变化。绿色的虚线表示通信通道，方形智能体表示通信的发起者。可以看到随着智能体占据更多的 Landmark，通信会越来越少。最右边的图显示的是在四个智能体构成的一个通信组，黑色和黄色箭头分别代表有无通信时的动作。可以看到，加入通信之后，智能体会趋向于合作的动作。

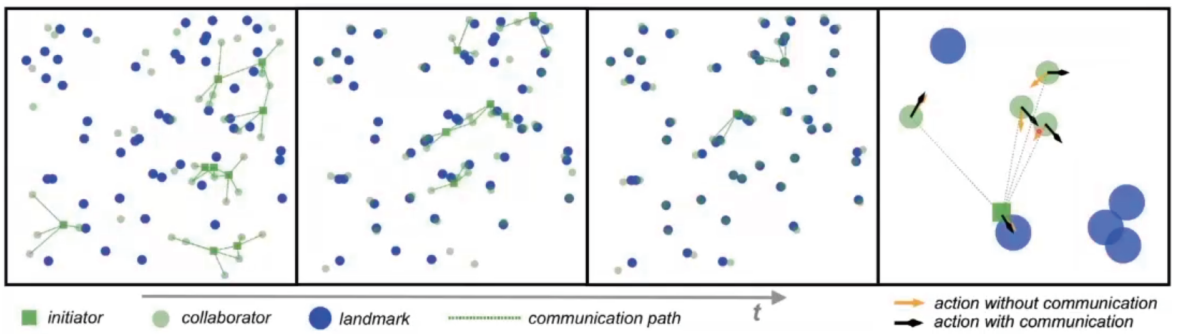


图 2: Cooperative Navigation

ATOC 以及之前的通信方法 CommNet、TarMAC、IC3Net，都是某种形式的 All to All、One to All 或者 One to Many 通信机制。但对于真正的网络通信或者人类对话，大部分情况下都是一对一通信。如何在中心训练分布执行的学习范式上进行一对一的通信，是我们接下来的第二个研究“I2C”的主要出发点。I2C 利用的是 Request–Reply 通信机制。智能体决定与视野内哪个智能体进行通信，并发送一个 Request，相应智能体的信息被传送给发出 Request 的智能体。

- **Learning Prior Network via Causal Inference**
  - Intuitively, an agent likely communicates with others who are potentially imposing influence on itself

$$\mathcal{I}_i^j = D_{\mathrm{KL}}(P(a_i|\boldsymbol{a}_{-i}, \boldsymbol{o}) \| P(a_i|\boldsymbol{a}_{-ij}, \boldsymbol{o}))$$

  - Joint action-value function can be exploited:

$$P(a_i|\boldsymbol{a}_{-i}, \boldsymbol{o}) = \frac{\exp(\lambda Q(a_i, \boldsymbol{a}_{-i}, \boldsymbol{o}))}{\sum_{a_i'} \exp(\lambda Q(a_i', \boldsymbol{a}_{-i}, \boldsymbol{o}))}$$

$$P(a_i|\boldsymbol{a}_{-ij}, \boldsymbol{o}) = \sum_{a_j} P(a_i, a_j|\boldsymbol{a}_{-ij}, \boldsymbol{o}) = \sum_{a_j} \frac{\exp(\lambda Q(a_i, a_j, \boldsymbol{a}_{-ij}, \boldsymbol{o}))}{\sum_{a_i', a_j'} \exp(\lambda Q(a_i', a_j', \boldsymbol{a}_{-ij}, \boldsymbol{o}))}$$

  - I2C can also serve as a component to reduce communication

$$\mathcal{I}_i^j = D_{\mathrm{KL}}(P(a_i|\boldsymbol{a}_{-i}, \boldsymbol{o}) \| P(a_i|\boldsymbol{a}_{-i}, \boldsymbol{m}_i, \boldsymbol{o}))$$

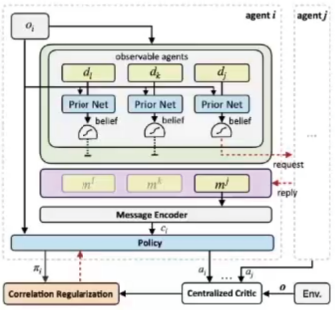
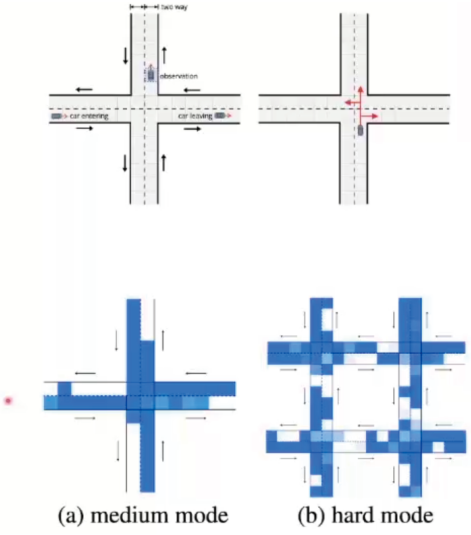


图 3: I2C

I2C 通过 Joint Action–Value Function 推断通信何时发生。在单个智能体中，通信对于奖励的提升容易衡量，但是这在 Joint Action–Value Function 中并不容易。我们的想法是，一个智能体应该选择与对自己影响比较大的智能体通信。对另一个智能体的动作考虑或不考虑这两种情况，我们将它们分布的 KL 散度作为智能体之间的影响。这两个分布通过 Joint Action–Value Function 计算得到，如图 4 中所示。计算所得的影响作为监督信号学习一个 Prior Network，用于在执行过程中决定通信与否。通过近似我们可以去除其他智能体动作的依赖，使得 Prior Network 能够独立工作，不依赖其他智能体信息。



	MEDIUM	HARD
I2C+TARMAC	**97.92%**	**92.17%**
TARMAC	97.60%	89.24%
IC3NET	78.08%	40.47%
NO COMMUNICATION	79.19%	48.10%

+ Reduced communication with better performance 👍

图 4: Traffic junction

在 Traffic Junction 场景中，对比 TarMAC 和 IC3NET，I2C 取得了性能上的提升，另外 I2C 也有助于减少通信量。左下角的图显示的是通信的 Overhead，颜色越深通信越大，可以很直观地看到通信量显著减少。

三、通信在智能体合作中的应用实例

通信中如何衡量其他智能体信息的重要性？我们希望，通过智能体信息的编码计算出智能体之间的某种关系，并利用这种关系来衡量来自其他智能体信息的重要性。

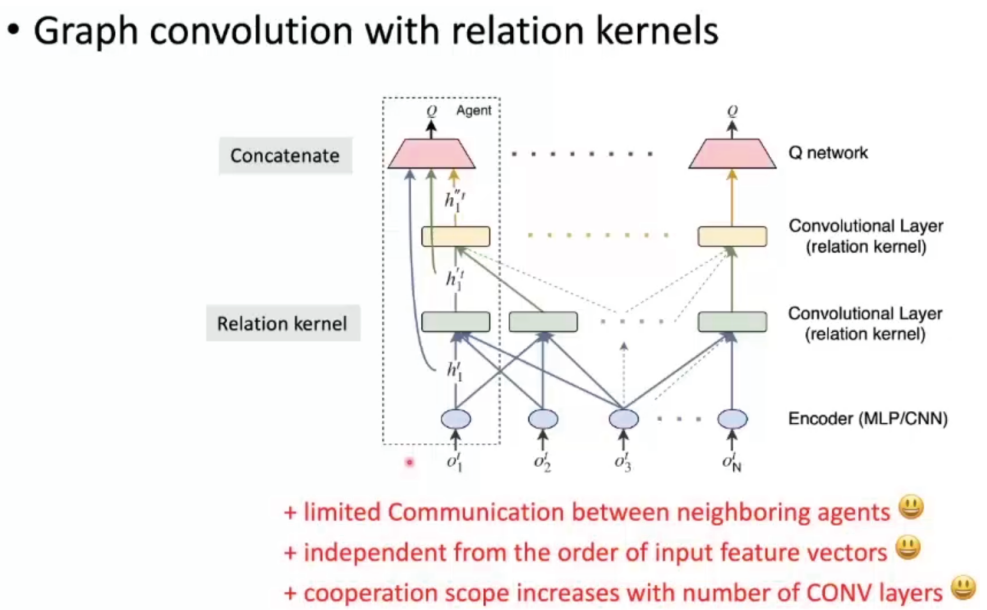


图 5: DGN

DGN 采用图卷积架构，并采用多头注意力机制作为卷积核，处理来自不同智能体的通信信息。每个智能体只与周围的邻居进行通信，通过多头注意力机制把信息整合在一起，发给下一卷积层。在下一层同样接收其他一跳内智能体的通信信息。这样随着卷积层的增加，所收集到的信息覆盖面积就会越来越大，从而使得智能体通过一跳内的通信机制获得更广泛的信息。

另外考虑在一些场景中，智能体高动态的变化，我们希望在这种情况下合作具有持续性。我们对前后两个时间步所计算的权重分布施加一个正则，促进权重在连续两个状态上尽量保持一致，从而提到合作的持续性。

最后一个研究中，我们探讨的是公平。公平对于我们社会有帮助的，但是我们想要的肯定不是绝对的公平，因为绝对的公平对系统性能没有帮助。在多智能体的系统中，我们想要的是公平和效率的权衡，这是多智能体中的经典问题。

• Fair-efficient reward

$$\hat{r}_t^i = \frac{\bar{u}_t/c}{\epsilon + \left|u_t^i/\bar{u}_t - 1\right|}$$

$\bar{u}_t/c$: the resource utilization of the system, encouraging to improve *efficiency*

$\left|u_t^i/\bar{u}_t - 1\right|$: the agent's utility deviation from the *average*.

Proposition 1. The optimal fair-efficient policy set π∗ is *Pareto efficient* in infinite-horizon sequential decision-making.
Proposition 2. The optimal fair-efficient policy set π∗ achieves *equal allocation* when the resources are fully occupied.

图 6: Fair–efficient reward

FEN 探讨的是如何通过简单的通信，来同时学习公平和效率。这是一个多目标优化问题，如果简单把两个目标融合在一起，这两个目标不是独立的，通过传统强化学习难以得到比较好的结果。

我们提出如图所示的公平 – 效率奖励 (Fair–Efficient Reward) 作为这个问题的优化目标，这一目标同时考虑了效率与公平。但是直接优化这一目标也比较困难，因此我们提出一种分层结构。上层的控制器的优化目标是 Fair–Efficient 奖励，控制器选择下层一个子策略执行策略。其中一个子策略的优化目标是环境给出的奖励，而对于其他子策略，我们提出一种基于互信息的奖励，使得不同的子策略行为不同，以提供给控制器多种选项。FEN 是分布式训练的，智能体之间只需要通过简单的通信获得平均效用即可。

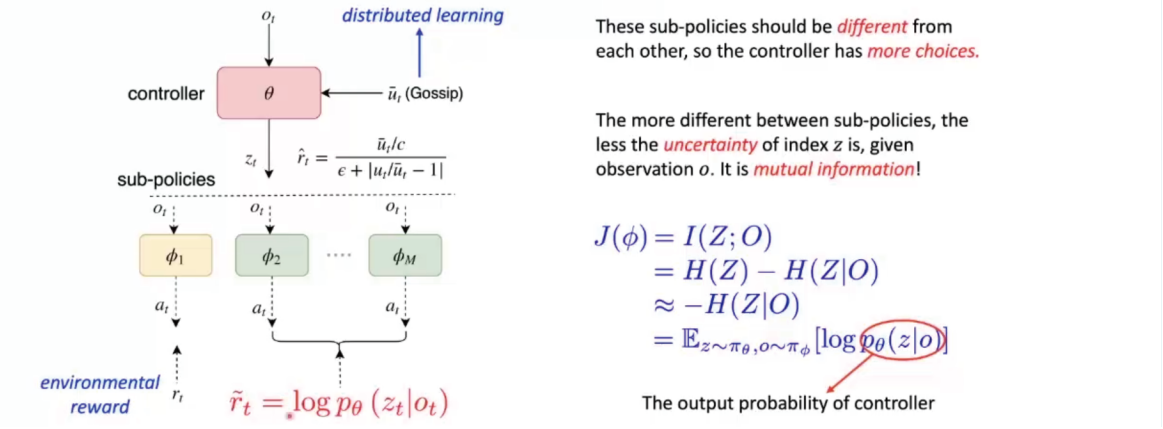
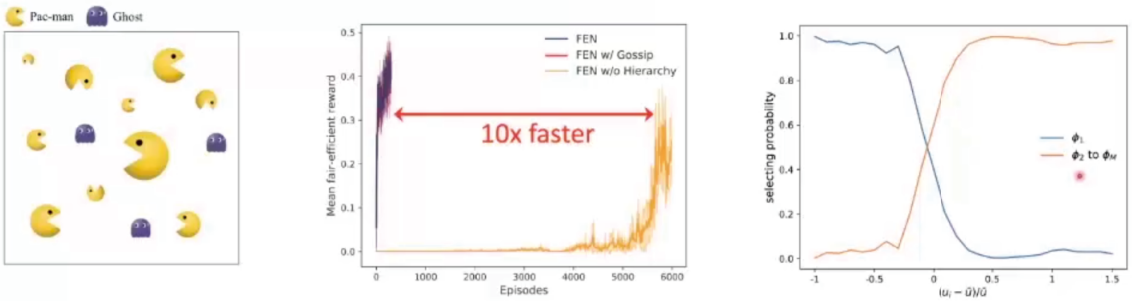


图 7: FEN

图中的实验情景模拟了马太效应，能力较强的智能体更容易吃到食物，因而能力变得更强。如果没有公平性衡量，会陷入强者更强的困境。FEN 对比其他基线方法，在公平与效率指标都是最优的，并且使用了分层结构比不使用分层结构的训练速度快了十倍。右图中可以看到当智能体的效用低于平均效用时，控制器倾向于选择优化环境奖励的子策略，反之则倾向于选择优化信息论目标的子策略，这与人类经验相吻合。



	social welfare	CV	min income	max income	episodes
Random	84 ±30	0.93 ±0.25	1	22 ±10	-
Independent	791 ±62	0.86 ±0.11	1	202 ±33	100
Inequity Aversion	702 ±90	0.80 ±0.16	2	152 ±18	1000
Min	18 ±8	2.04 ±0.66	0	7±4	6000
Avg	527±113	0.86 ±0.21	2	126±18	1000
Min+αAvg	441±75	0.85 ±0.18	1±1	103±16	2000
FEN	**830**±22	**0.06** ±0.01	**79**±2	94±3	300
FEN w/ Gossip	**841**±55	**0.07** ±0.01	**76**±4	95±4	300
FEN w/o Hierarchy	251±12	0.06 ±0.04	23±2	26±1	6000
FEN w/ Random Sub-policy	834±47	0.08 ±0.02	66±7	99±6	-

图 7: The Matthew Effect

四、结语

无论是对于智能体还是人类，通信的作用都非常大，因为通信能够改变学习的过程。从人类的角度来讲，我们很多语言性质，都反映了我们对事物的认知。关于通信如何去改变人类的学习以及人对事物的认知这一方面，当前的研究并没涉及。但随着我们进一步深入的研究，可能会探索到通信以及语言在人类认知、学习上所产生的更深远的影响。

中科院自动化所赵冬斌：从仿真到实体的深度强化学习方法

整理：中科院自动化所 朱圆恒

6 月 23 日，中科院自动化所研究员赵冬斌在 2020 北京智源大会“决策智能”专题论坛上做了《深度强化学习 – 从仿真到实体》的报告。赵冬斌是 IEEE Fellow，是多个国际计算智能领域权威期刊的编委，也是多个国际权威期刊的特邀编辑，担任 IJCNN2019 国际程序委员会的主席。其工作主要在计算智能，深度强化学习，自适应动态规划理论和方法，以及智能车辆、机器人等方面的应用。

在报告中，赵冬斌从三个方面展开介绍:

第一，研究探索更聪明，更智能的游戏 AI 算法。
第二，应用游戏 AI 的算法实现智能驾驶的安全、稳定决策。
第三，在视觉导航、环境探索、协作和博弈对抗领域实现机器人从仿真到实体的迁移。

以下是赵冬斌演讲全文:

今天的报告大家都提到深度强化学习，深度强化学习结合了强化学习的决策能力和深度学习的感知能力，这需要感谢 Google DeepMind 和 David Silver 提出的深度强化学习方法，这个方法也被列为了人工智能近几年几个里程碑的事件之一。

图 1：基于 DRL 的 AI 里程碑事件

图 1 列出了基于 DRL 的 AI 的里程碑事件，包括 2015 年谷歌提出解决 Atari 游戏的深度强化学习方法 DQN；2016 年的 AlphaGo 以 4:1 的大比分战胜了世界围棋顶级选手李世石；2017 年谷歌的 Alpha Zero 用人类的数据自学习，还具有泛化性，可以下国际象棋和日本将棋；2019 年包括谷歌的 Alpha Star 攻破星际争霸，CMU 的六人德扑 Bot Pluribus，这都是不完全信息的博弈，谷歌还将其扩展到三维的第一视角，做了雷神之锤游戏，微软也针对麻将做了一个麻将 AI Suphx，Suphx 的实力达到十段，人类基本在九段左右；2020 年谷歌又提出了 MuZero，既能玩游戏，还可以做围棋，进一步提高了算法的泛化性，也就是我们常说，更通用的人工智能。

我们早期的研究主要围绕强化学习和自适应动态规划，拿到了深度强化学习方向国家第一个自然科学基金；2016 年发表一篇深度强化学习的综述文章；2017 年 IEEE TCDS 论文获得了年度优秀论文（唯一）；2018 年组织深度强化学习的专刊，参加了 Robomaster 全球人工智能挑战赛，全部 4 次最高评价；2019 年参加星际争霸天梯赛，获得学生组冠军，参加 IEEE CoG Fighting AI 比赛，获得亚军等等。这次报告主要是挑选一些从仿真到实体相关的工作跟大家分享。

一、游戏 AI

首先跟大家分享就是格斗游戏的工作，格斗游戏动作空间比较大，动作和连招组合起来有 56 个动作，但是反应时间只有 16 毫秒。虽然对手的状态一目了然，但是对手是否能放大招却无法知道，因此这是一个不完全信息博弈问题。而且有三个对手，需要适应多种不同类型的格斗人物特性，对泛化性有一定的要求。

我们和英国 Simon 教授合作，利用 RHEA 决策和强化学习对手建模的方法做了一个通用的格斗游戏 AI。利用遗传算法的搜索能力强、强化学习对手建模能力强的优势，实现了算法的泛化性，适合不同的格斗角色。我们也和 2018 年排名靠前的 AI 做了一个对比测试，得到了一个效果（如图 2）。我们参加了 2019 年的格斗比赛，在比赛中，我们有对手建模学习的过程，一边和对手比赛，一边学习对手的策略，利用学习到的对手模型，更新策略网络，利用更新的策略获得了比赛的胜利。最终我们在比赛中获得了亚军。

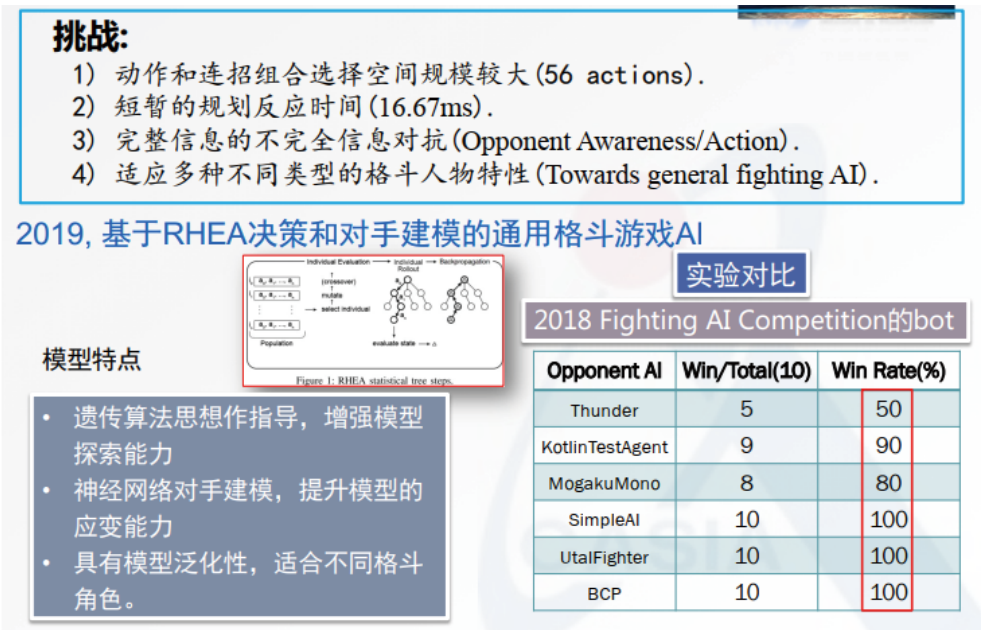


图 2：格斗游戏

前面的格斗游戏是一对一，星际争霸游戏则是多对多。我们针对星际争霸的研究也比较早，主要做星际争霸的微操，针对多智能体、不完全信息和实时决策的问题，我们定义一种高效的状态表示方法（如图 3），包括前一时刻的，自己的动作状态和对手以及队友的状态，输出移动或进攻动作。提出了多智能体梯度下降 SARSA (λ) 的方法来解决星际微操中多智能体的决策控制问题。相关文章发表在 IEEE TETCI 期刊上，得到 Popular Article，同时文章也被谷歌发表在 Nature 的 AlphaStar 论文引用。在游戏 AI 领域，我们发表两篇综述，2016 年发表在控制理论与应用的《深度强化学习综述：兼论计算机围棋的发展》，2017 年《深度强化学习进展：从 AlphaGo 到 AlphaGo Zero》，加起来下载量 1 万余次，也在中国自动化学会等官方微信里面分享了一些体会。组织了一个 IEEE 神经网络与学习系统汇刊专刊，主题为深度强化学习和自适应动态规划。和 Simon 和 Julian 这两位 IEEE ToG 的创刊主编和现任主编，一起组织了深度强化学习和游戏的专刊。

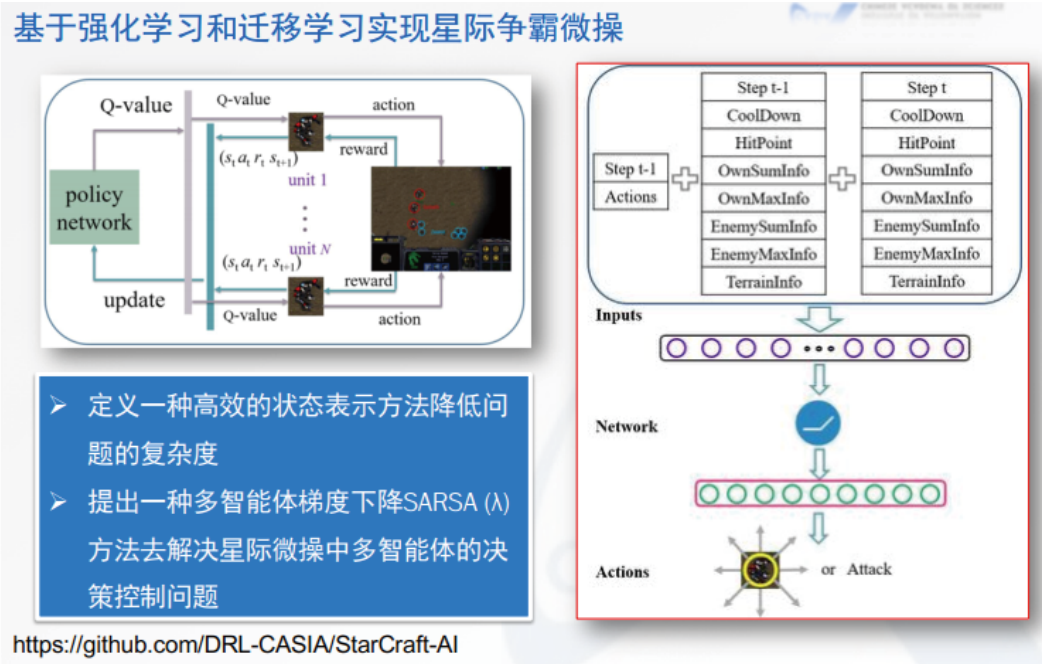


图 3：星际争霸微操

总结一下游戏 AI 的发展，我们分析谷歌和 Open AI 基本都是从游戏开始做，最终把它应用到实际，包括电力系统优化，机器人操作，机器人运动和机器人玩魔方。因为在游戏环境下学习验证，可以避免决策过程遇到的安全、伦理、数据高效和加速等等问题。我们自己做了一个游戏 AI 的发展趋势（如图 4），横轴从单个体到多个体，纵轴是二维到三维。在左下角的 Atari 游戏和围棋这类单个体完全信息问题，目前是解决的比较好的，最典型的包括谷歌的 MuZero。但是包括蒙特祖玛等关于推理的游戏还有待进一步的提升。第一视角三维的游戏，比如赛车、Minecraft 以及 ViZDoom 等还有许多需要做的工作。二维环境下，如星际争霸、DOTA2 等多智能体、不完全信息博弈问题，已经有很多优秀的工作，但是已有的工作对硬件资源需求比较大，如何能够优化算法，更好的把这些算法应用在实际的系统中，还有很多的工作要做。再结合起来就是三维多个体问题，这一块也更具挑战性。

图 4：游戏 AI 发展趋势

二、智能驾驶

我们将上述游戏 AI 的算法尽量应用到实际系统中，第一个考虑的就是智能驾驶。智能驾驶是对车辆通过相机、激光雷达等传感器获得的一些信息来进行周围环境的检测和物体的识别，在这个基础之上，我们再来做车辆的预测和决策的控制，整个链条比较长，这里我只是挑一些识别，还有车辆的纵向控制、换道控制等来跟大家分享一下。

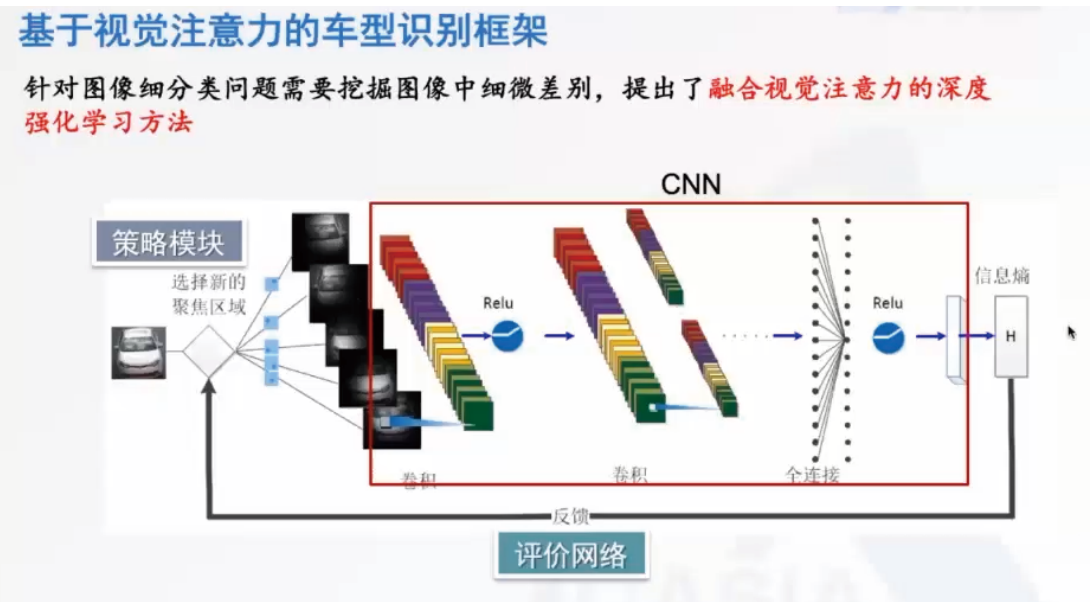


图 5：车型识别

我们在 2016、2017 年做了一个基于视觉注意力的车型识别框架（如图 5），针对图像细分类问题需要挖掘图像中细微差别，提出了融合视觉注意力的深度强化学习方法。人对车型识别时会关注车标、进气格栅、挡风玻璃和雨刷器等等，我们希望把注意力集中在这些位置，通过卷积网络来计算，利用识别信息熵做反馈评价。这种方法使得性能获得了很好的提升，相关文章发表在 IEEE TCDS 期刊杂志上，得到 Popular Article 第一位，被评为年度优秀论文。把这个方法用在实际的多任务学习，进行前方车辆检测和前车距离的检测，这两项我们参加中国智能车未来挑战赛，均获得了第一名。其他的工作包括交通信号检测、车道线识别检测，车道保持状态监测等也都参与了一些。

在得到了检测的信息之后，下一步的工作就是做智能驾驶的决策控制，其中有一个工作比较典型，就是车道保持。我们延续之前的工作，输入的是车辆观察到的图片，用多任务学习的方法得到车辆和车道偏离的距离，包括车辆的偏角以及车道的朝向。再把学习到的结果作为输入，利用 DDPG 方法进行决策，来控制方向盘的转角，进而实现车道保持的任务（如图 6）。

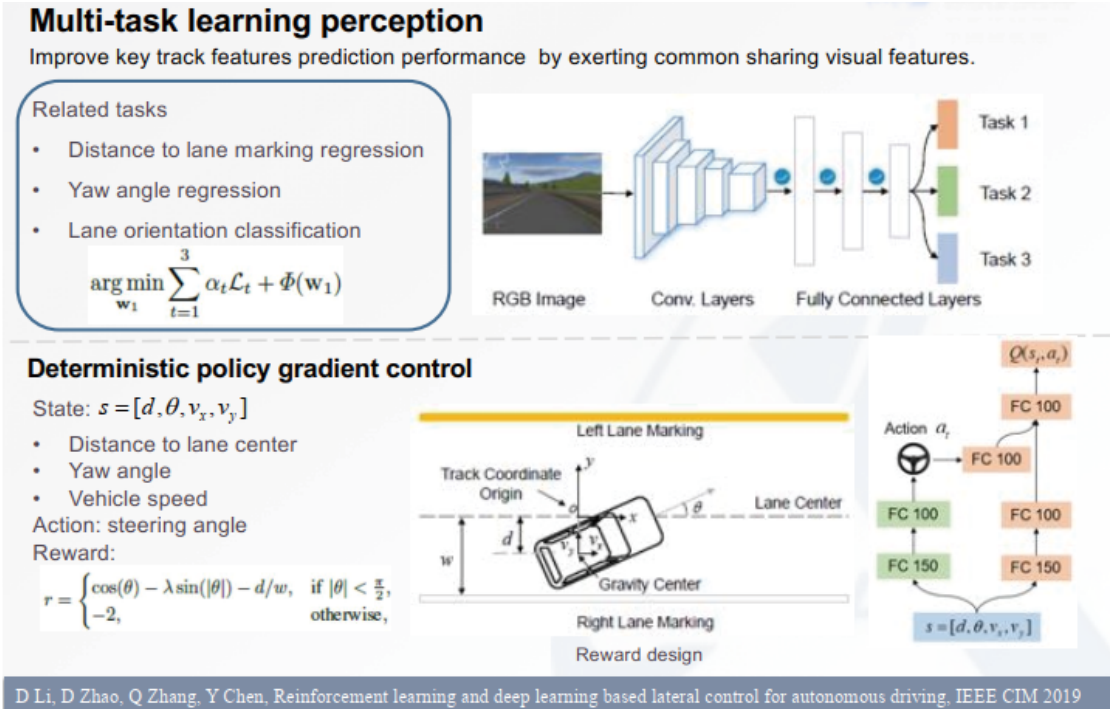


图 6：车道保持

从实验的结果来看，多任务的学习与单任务学习相比，经过不同任务的加权，可以得到更好的性能。将多任务学习与一些基线方法进行比较，结果也更稳定，同时可以泛化到新的赛道环境。总的来说，把多任务的感知和强化学习的控制结合起来，可以实现车道保持的控制，既可以看到自己本身的感知结果，还可以获得目标控制的结果，包括在单车道和三车道环境下的控制。

上面介绍的是单车道的保持，但是如果本车道有一辆开的比较慢的车或者事故车，我们就需要换道通行，这在车辆控制里面属于横纵向综合控制。我们设计了一个分层的强化学习（如图 7），上一层进行高层的决策，判断是否保持车道或是换道，下一层进行轨迹规划进而执行。测试过程中，算法首先在 Udacity 平台进行测试，从

实验结果中可以看到，当规则因素和深度强化学习相结合的时候，可以实现及时换道的功能，单纯使用规则的话，换道决策比较保守。为了验证算法的可迁移性，我们直接把算法迁移到另外一个更专业的仿真测试平台 VTD 里面，在迁移过程中并没有进行网络训练，但是效果非常好，这主要得益于之前设计的分层的架构。

把前面智能驾驶的感知和决策的工作结合起来，我们设计了一个无人配送车。这也是在今年的疫情之后，大家尽量避免在食堂集中就餐，由我们的无人配送车带着装有 200 斤盒饭的拖车，并将其送至我们所里面的两个大厦，让大家分散就餐。这个工作量还是比较大，无人配送车可以减少人和人的接触。无人配送车具有路人检测，路径规划，及时避障等功能。我们的无人配送车在自动化所开放日自动化之光进行了展出，也得到了今日头条的报道。

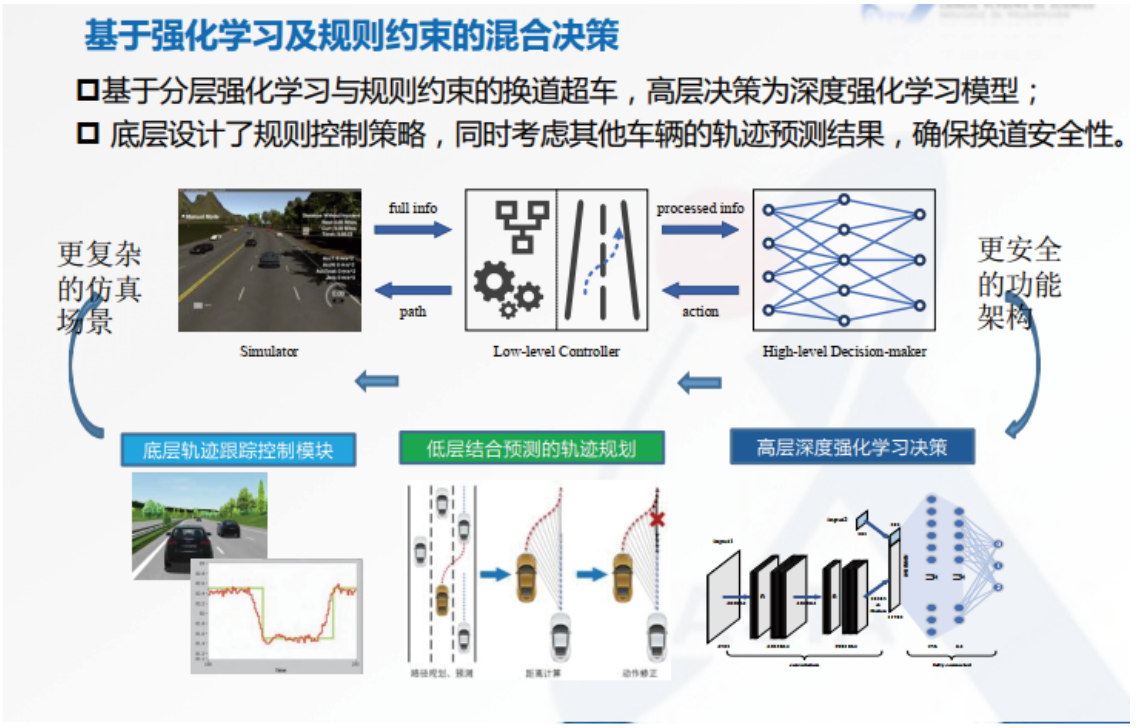


图 7：超车换道

在智能驾驶和深度强化学习领域，大家应该齐力为这个领域做一些贡献。因此我们也做了一些自己的工作，收集了深度强化学习相关的数据集，包括车道线检测和车辆检测、交通标志检测，并将我们参加的比赛和我们自己收集的数据集公布在开源的网站上（如图 8），欢迎大家更好的利用。我们自己的算法也进行开源，包括车道保持算法，希望对智能驾驶和深度强化学习领域有所推进。

图 8：智能驾驶 – 数据公开

今年我们也参与了智能驾驶决策比赛 I–VISTA 的组织，大家感兴趣可以报名（如图 9）。一共有预选赛和决赛两部分，提供专业软件 VTD 让大家试用，参赛选手利用软件进行算法的调试，最终提供一个性能的测试报告，以此判断参赛选手是否可以参加决赛。决赛的时候，由主办方提供装有 VTD 的电脑，来做硬件在环的测试，考验参赛选手的算法是否更适合实际的应用系统。

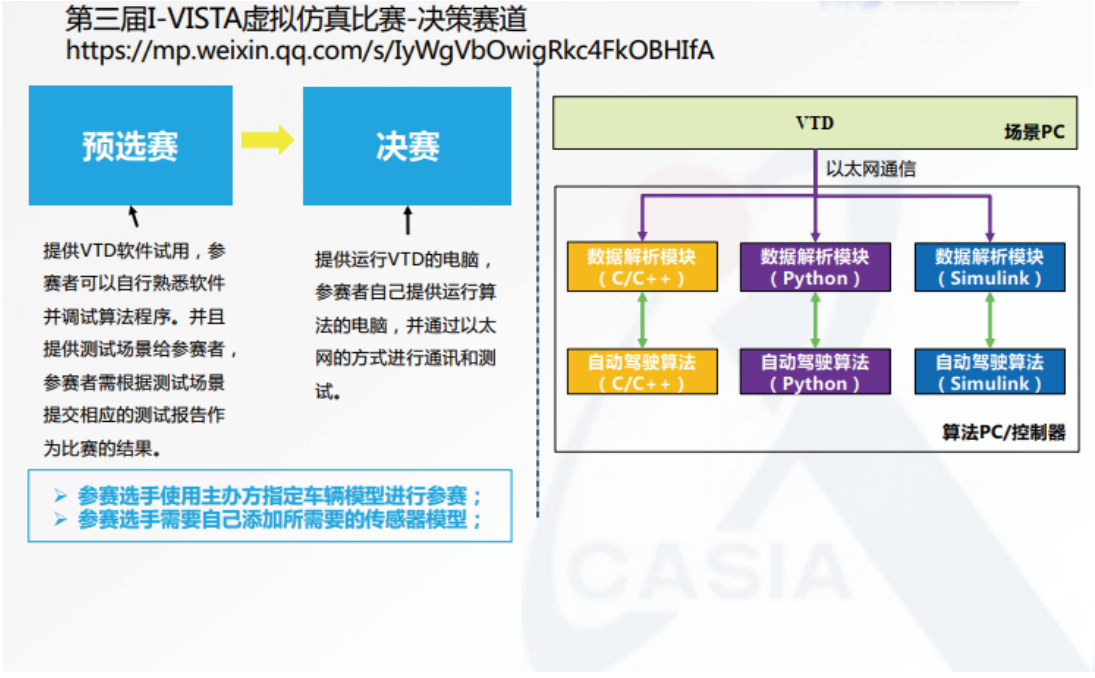


图 9：第三届 I–VISTA 虚拟仿真比赛

智能驾驶的比赛有很多，但是决策相关的比赛很少。前面我提出了对游戏 AI 的展望，针对智能驾驶领域，平台和算法方面也都需要很多的工作，需要大家共同参与。包括设计一个典型的测试环境，到底哪一种环境能够更好的对智能驾驶算法进行测试； 怎么进行交互，如何实现环境车辆的切入干扰，以此来检测算法的智能性，也就是环境和交互； 算法的智能该如何评价，目前比较简单的评价方式就是如果违规就扣一点分，但是这种评价的方式是否科学，是否评价出来的前几名真正代表了算法的智能水平比较高； 在 L4、L5 甚至无人驾驶里面，是不是能够细分更多的评价准则，这都还有很多的问题需要我们共同去解决。我们也希望在决策智能领域，可以做出一个平台提供给大家来进行学习测试。当然，智能驾驶的算法也有很多的工作目前还不是很成熟，主要针对复杂环境有很多开放的不确定的问题，比如有车有人随时出现。算法在智能驾驶里面一定要有可解释性，在一些干扰情况下，算法可能会出现一些误动作，包括各种场景的鲁棒性、泛化性，还有很多的工作要做。

三、机器人

最后跟大家分享一下我们在机器人方面的工作，包括视觉导航、环境探索以及实体的对抗。视觉导航是在室内的场景，通过视觉传感器，寻找一个物体，比如找微波炉，机器人能够发现它并能走到附近，就表示这个任务完成。常用的方法是 Slam 建图，然后定位，如果没有地图的时候，我们就用强化学习的方法来实现。但是传统的建图，对物体的位置信息要求比较严格，在新的环境里面可能不适用。我们提出了一个基于 Markov 网络和图神经网络的强化学习方法 MGRL。物体之间的关系，我们用 Markov 网络来表示，用图神经网络来推理，用强化学习的框架来训练。图 10 中，输入是摄像头拍摄的前向、左侧、右侧的图片，用卷积网络提取视觉的特征，同时用 Markov 网络来获得三张图片中离目标点的距离，得到三个具体的信息，把它们定义为系统的状态，用 Actor–Critic 的方式来计算 Value Network 和 Policy Network。

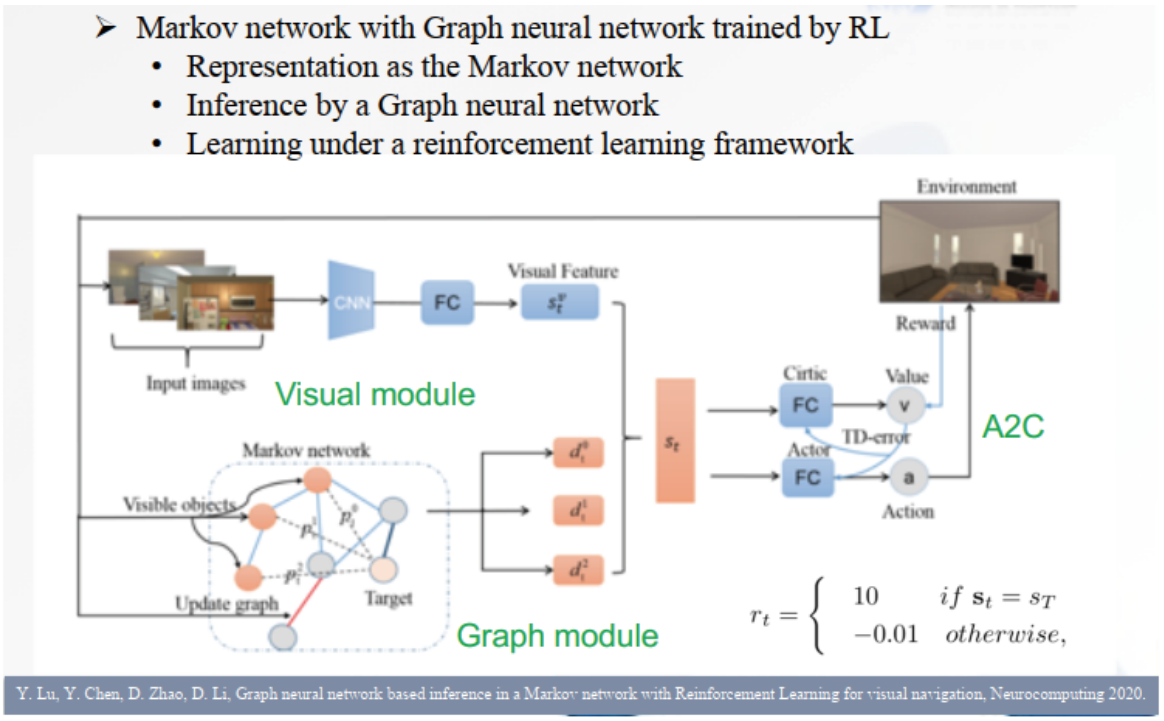


图 10：MGRL

在一个标准的测试环境里面，有四类的场景，包括一个厨房、起居室、卧室等等。我们应用图神经网络强化学习的方法，结合各种消融实验，跟 A2C 进行比较。图 11 右下角是学习到的物体之间的关系，假如物体之间没有关系，它就没有连接线。此外，我们将我们的方法和一些最先进的方法进行比较，比如 ICLR 2019 年的方法，在四个环境里面进行的测试，对比的方法针对每个环境需要一个模型，我们的方法总共只需要一个模型；我们的方法对动作的需求更低； 同时，我们的方法找到目标所需要的最大步数也更少。我们将这个方法迁移到实际的环境中，我们用一个机器人搭载几个摄像头，去找一个目标，通过简单几个步骤就可以找到目标的位置。

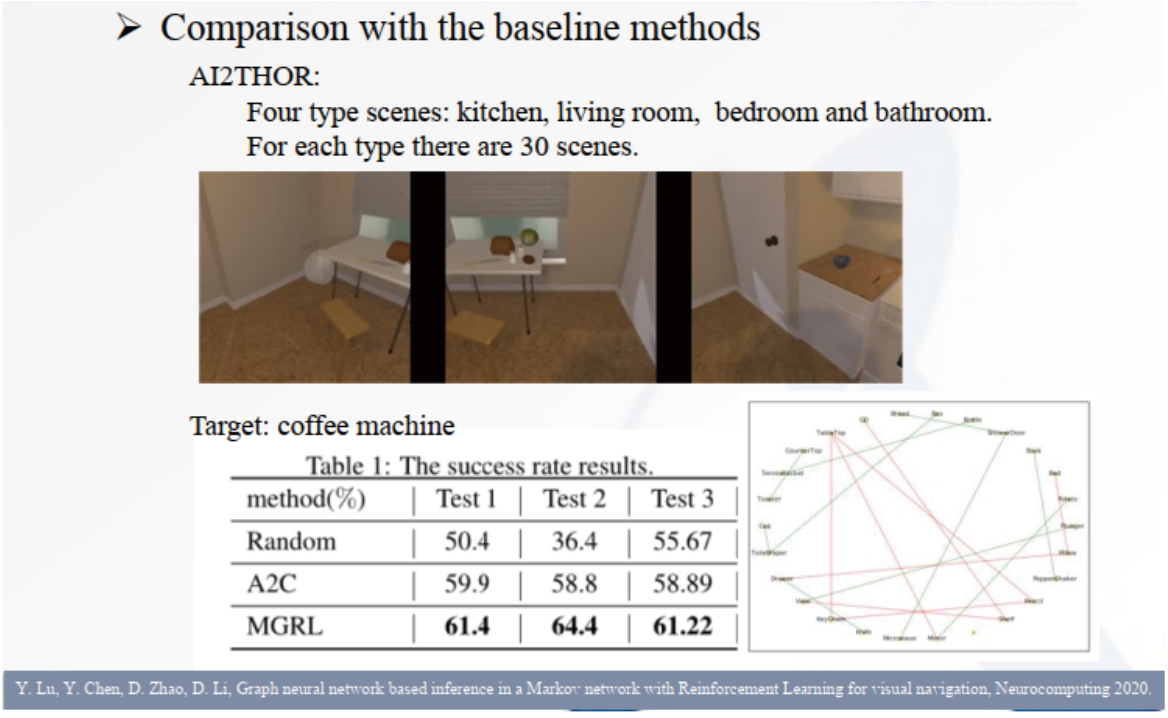


图 11： MGRL 实验结果

关于机器人，还想介绍一下未知环境探索，这也是智能驾驶和机器人等领域里面大家比较关注的问题。一般的方法，可以通过规则的方法来进行探索，但是对于比较杂乱的环境，这个方法就有一些局限性。也有人用端到端的学习方法，但是迁移性比较差，新环境需要大量的样本。我们提出一个深度强化学习的方法，包括几个模块： 决策、规划和建图（如图 12）。决策模块根据机器人已经走过的路径以及现在的地图，来确定下一个要走的位置的点； 规划模块，根据这个点用传统的 A* 方法规划出到达这个点的轨迹； 建图模块是在完成行驶轨迹的过程中，一边做动作，一边收集信息，来完成整个地图的更新。

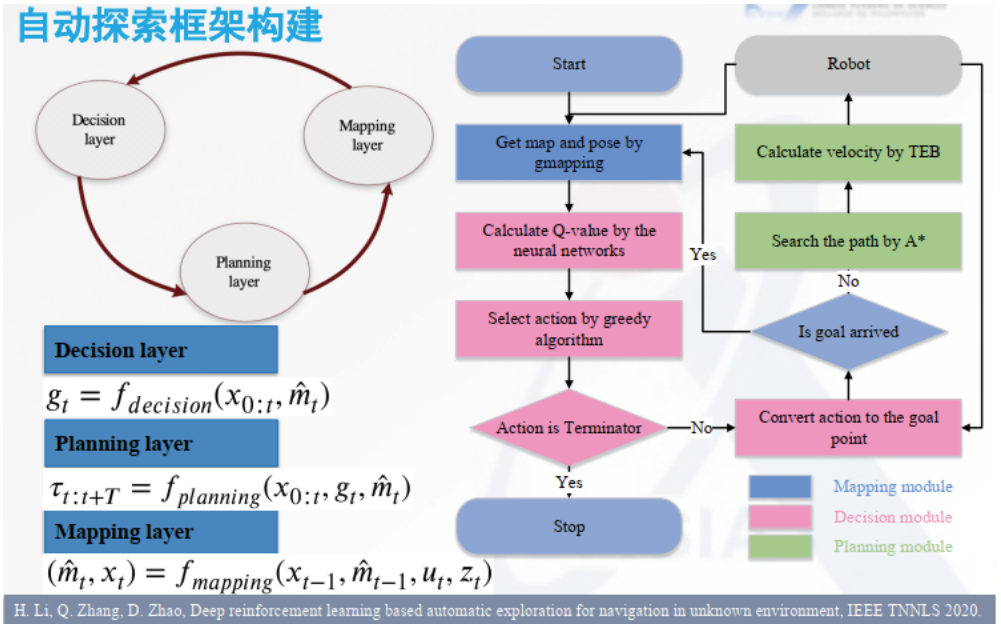


图 12：基于 DRL 的自动探索框架

我们采用一个全卷积神经网络，定义一个性能指标函数，定义地图和已知真值地图的差别。但是真值地图我们很难获得，因此我们将其替换为所获得地图的香农熵，也就是地图的不确定性。同时考虑建图所用的路径的长度，路径越短越好，结合机器人运动的约束，进而得到奖励的定义（如图 13）。

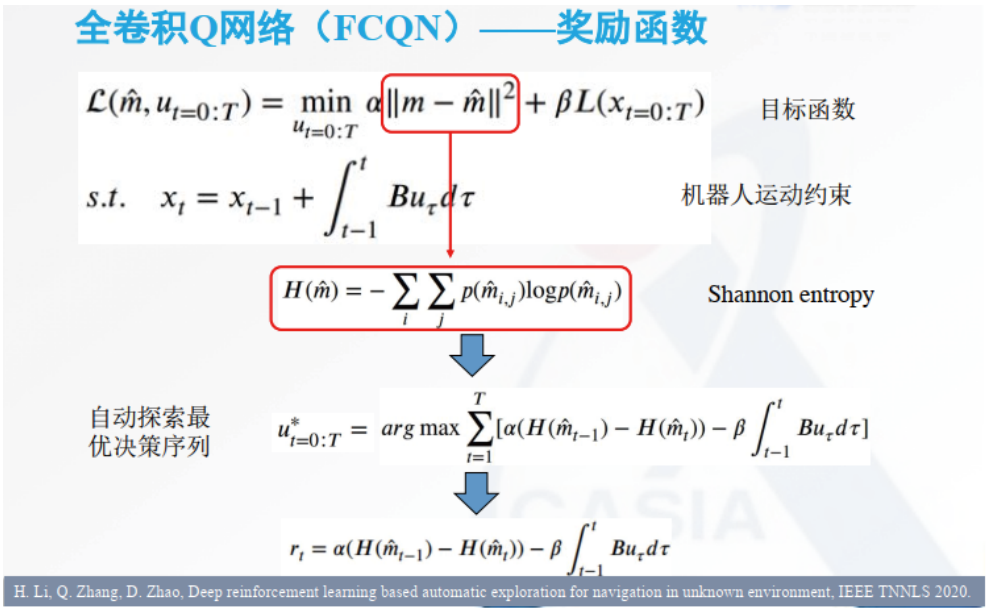


图 13：全卷积 Q 网络

我们设计了基于全卷积 Q 网络的辅助任务，输入地图的位置和边缘的图像，输出是优势函数和状态值函数。将地图进行离散化，判断网格点是否被占用，灰色的网格点是没有探索到的空间，辅助任务就是发现在建图过程

中得到的边缘图像，使得探索更有效率。在地图测试结果中，辅助任务的方法效果都是比较好的。

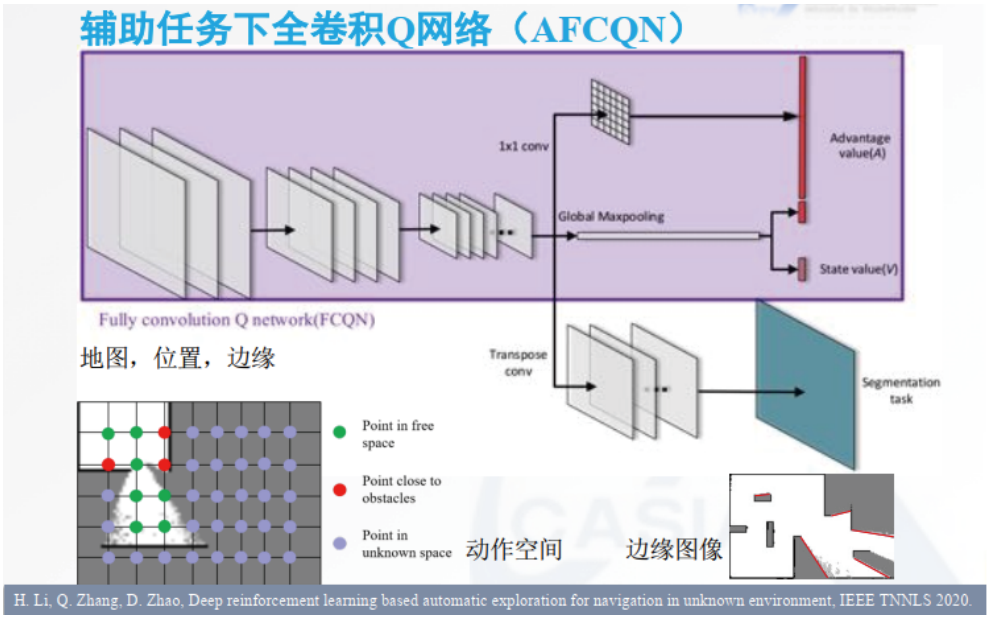


图 14：辅助任务

上述是在仿真环境里进行测试，同时我们也利用自己的比赛地图，来做实际的测试。图 15 可以看到性能的对比，从仿真到实际里面，可以看到算法性能有一点损失，但是大部分还是能够保持算法的效果。分析原因，我们觉得是因为算法的决策输出是一个目标点，而并不是机器人的具体的控制，具体的控制可能不确定性更大一点，而目标点是更中观或者宏观的指标，所以这个算法迁移带来的差异性更小一点。

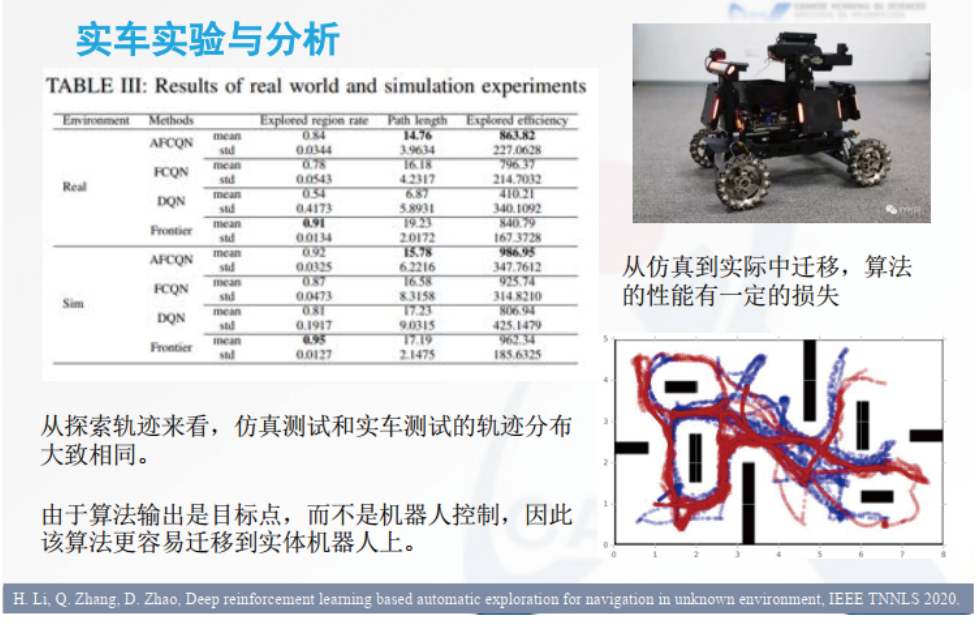


图 15：实车试验与分析

最后我来介绍一下我们在 Robomaster AI 挑战赛的工作，这是大疆在 2018 年开始的在 ICRA 会议上举办的一个比赛。每队有两个机器人，在一个 5 米× 8 米长的空间里面有一些障碍，互相寻找并攻击对方。机器人可以发射炮弹，需要攻击机器人的装甲板，机器人四周有四块装甲板，如果被击中的话，这个机器人就会掉血。在规定的时间里，哪一组机器人的血量保持最多哪一组就获胜。我们在 2018 年获得了算法的最高评价。在 2018 年的时候，大疆的机器人跟所有的对手来进行比赛，他们的机器人性能比较好，速度比较快，因此我们果断采取一个保守的策略。在打斗的时候如果发现装甲板在闪烁，就表示子弹打中对手，队友机器人立刻过来补刀，对手机器人没有灯了，就表示它已经死掉了，最终我们获得了比赛的胜利。2018 年大家的平台不一样，很难公平比较算法的优劣，2019 年都用了同样的平台来进行设计，也增加了一些难度，比如每辆机器人最开始没有子弹，需要到两个固定装弹区域进行装弹，每次装弹数量也是限量的，其他的跟 18 年基本相同。这个实体比赛会有很多不确定性出现，所以实体的比赛和应用还是有很多的难度，我们需要去克服，而且我们还有很多工作需要去做。

整个 Robomaster 比赛还是蛮复杂的，我们做了一个系统架构的图（如图 16），包括底层的驱动，如何根据它做运动； 包括相机、激光雷达这样的驱动，将得到的数据到输入感知层，实现机器人的定位； 前边车辆的检测，自己车辆的检测、敌方车辆检测，车辆检测的装甲板在哪里，根据检测结果融合、追踪和预测节点。接下来做规划，规划过程中要兼顾全局规划和局部的避碰，最终实现底盘的控制、云台控制以及射击控制，两个机器人之间的协作也会有一个学习的策略。

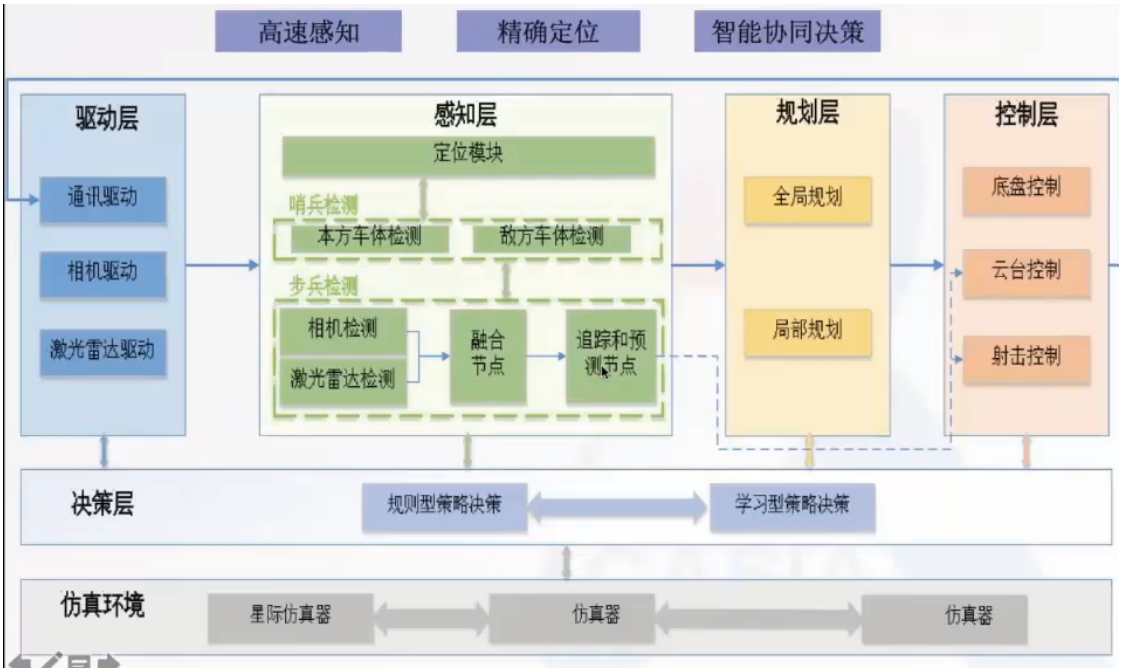


图 16：Robomaster 系统架构和关键技术

如果直接进行实体实验来完成上述工作，实验很难进行，因为实体实验经常发生碰撞，对机器人有损耗。所以我们开发仿真平台，仿真平台课题提供机器人汆的第一视角，能够为检测提供一些数据。在二维的环境下，可以来演练两个机器人协作的策略，验证包括瞄准、射击，提高射击的准确率。还包括整个全局协作的工作，后

续还将这些工作迁移到实体平台当中。如图 17 所示，我们针对仿真到实体的工作，将实体环境抽象简化，简化成二维的环境，我们用一个类似于星际的环境进行策略学习，学习两个机器人之间如何进行协作。再把它分层强化，在下一层的时候把策略转化为每个个体、每个机器人执行的任务，在执行的过程中进行路径规划、寻找目标、目标检测，最后迁移到实体里面。这也是一个多智能体强化学习协同博弈问题，这还是比较复杂的一项任务。

图 17：Robomaster—从仿真到实体

四、总结

我们主要在做深度强化学习相关的工作（如图 17），在游戏 AI 包括单个体格斗的游戏，例如星际争霸和 ViZDoom 等方面进行研究。游戏环境是基础，我们继续探讨更智能的游戏 AI 的算法，更具可解释性的算法。进而把这些方法应用到智能驾驶领域，包括车辆检测、车辆的保持和决策的控制，还包括机器人导航、位置环境探索，以及协作和博弈对抗的问题中。

启元世界高超：启元星际指挥官——基于高效平台训练的最高级强化学习智能体

整理：启元世界

6 月 23 日，启元世界技术副总裁高超在 2020 北京智源大会“决策智能”专题论坛上作了《启元星际指挥官：基于高效平台训练的最高级强化学习智能体》的报告。高超曾先后就职于易趣、百度、阿里巴巴等知名企业。在离开阿里巴巴前负责广告投放平台团队。 2018 年加入启元世界，任技术副总裁，负责强化学习平台产品的建设，在分布式系统、高性能计算、大数据等领域有 13 年的工作经验。

在报告中，高超从三个方面展开介绍：

第一，介绍启元星际指挥官达到职业玩家水平，其背后的技术要点
第二，介绍启元世界强化学习平台的设计要点
第三，介绍启元世界强化学习平台与产业的结合点

以下是高超演讲的全文：

首先，向大家介绍一则消息，6 月 21 日，启元星际 AI 顶级职业选手挑战赛中，启元星际指挥官以两个 2 ：0 的比分，分别击败了中国两位顶尖星际职业选手黄慧明和李培楠。其中，李培楠更是现役中国选手中排名最高的。这一成绩说明了启元世界已经具备解决强化学习领域最为复杂问题的能力，以及世界一流的技术水平。

一、星际争霸智是强化学习领域最为复杂的问题之一

星际争霸是电竞领域最为经典的游戏之一，也被认为是最具挑战的即时战略游戏。在该款游戏中，玩家需要同时考虑经济建设、基础设施建设、科技发展、战斗单元的建造。尤其是该游戏中各个兵种有相克关系，玩家在建造建设的过程中，还要综合考虑战斗规划、兵种配合、资源合理分配等要素，从而达到相对的战斗力最大化。最后，玩家还需要有快速响应的临场操控能力，指挥部队战斗。

启元世界从 2017 年 8 月，公司诞生之初就开始基于星际争霸的环境研究强化学习技术。在 2018 年 4 月，北京大学举办的第 42 届 ACM-ICPC 全球总决赛上发布了星际争霸人机协作挑战赛。2018 年 11 月，启元星际指挥官在 Mini Game 中达到职业选手水平，并受邀在加拿大举办的 AIIDE 会议上演讲。之后，启元星际指挥官全面进入 Full Game 的研究，并于 2019 年 9 月达到白金水平，3：0 击败人类黄金级选手。2019 年 12 月，达到钻石水平，并在同年的 NeurIPS 会议中展示 Demo，成为该届会议最火爆的体验项目。2020 年 6 月达到宗师水平，并于当月击败了人类顶级选手。

星际争霸是当今强化学习届最为复杂的问题之一，其决策复杂度对比围棋有数十个数量级的提升，其表现如下图：

图 1：围棋 AI 与星际 AI 对比图

1. 围棋是完全信息下的博弈，而星际是非完全信息的；
2. 围棋是回合制对战，而星际是毫秒级的变频决策；
3. 围棋仅对一枚棋子进行操作，而星际是要从上百个单位中选择若干；
4. 围棋仅有落子一种指令，而星际有建造、移动、攻击等上百种指令；
5. 围棋的指令目标在 19*19 的格子中选择一个，而星际的目标则要在 256*256 的区域，或者数百个单位中做选择；
6. 围棋平均每局只需要决策 100 次，而星际需要 2000 次以上的决策。

综合上述复杂度，围棋的决策空间是 361，而星际是 10 的 26 次方。

二、启元世界星际指挥官的算法实现要点

启元世界星际指挥官为了应对如此复杂的问题，设计了模仿学习、强化学习、演化学习三位一体的训练流程。

模仿学习可以以较低的成本初始化星际 AI 智能体，从而节约大量的、低效的智能体初期探索算力。模仿学习以少量人类数据作为输入，输出超过人类黄金选手水平的智能体。同时，以人类数据初始化的智能体，可以为后续的训练过程带来打法上的多样性，从而保证在与人类对抗中的鲁棒性。模仿学习的另一个好处是，可以构建一个快速验证算法策略的技术架构。

强化学习以模仿学习阶段输出的模型作为起始模型，使用带有 GAE 的近端策略优化算法（PPO）进行优化。为了在强化学习阶段保持人类多变的打法，会比较正在优化的策略分布与模仿学习策略分布的 KL 距离，并在损失函数中加入一个惩罚项。

为了在强化学习过程中演化出更为强劲鲁棒的打法，加入了智能体联赛机制，以期在智能体相互博弈的过程中，共同促进成长。在该联赛中，主要包括三类智能体。其一是 Main Agent，其目的为变强和变鲁棒。它会与联赛中的所有打法的智能体及其历史版本进行对抗。第二类是 Main Exploiter，其目的是发现 Main agent 的缺陷，并针对缺陷训练应对的策略，最终作为 Main Agent 的陪练，补强其漏洞。第三类是 League Exploiter，其目的是发现 league 中全体智能体的打法缺陷，并找到应对策略，最终作为 Main Agent 的陪练，补强其漏洞。

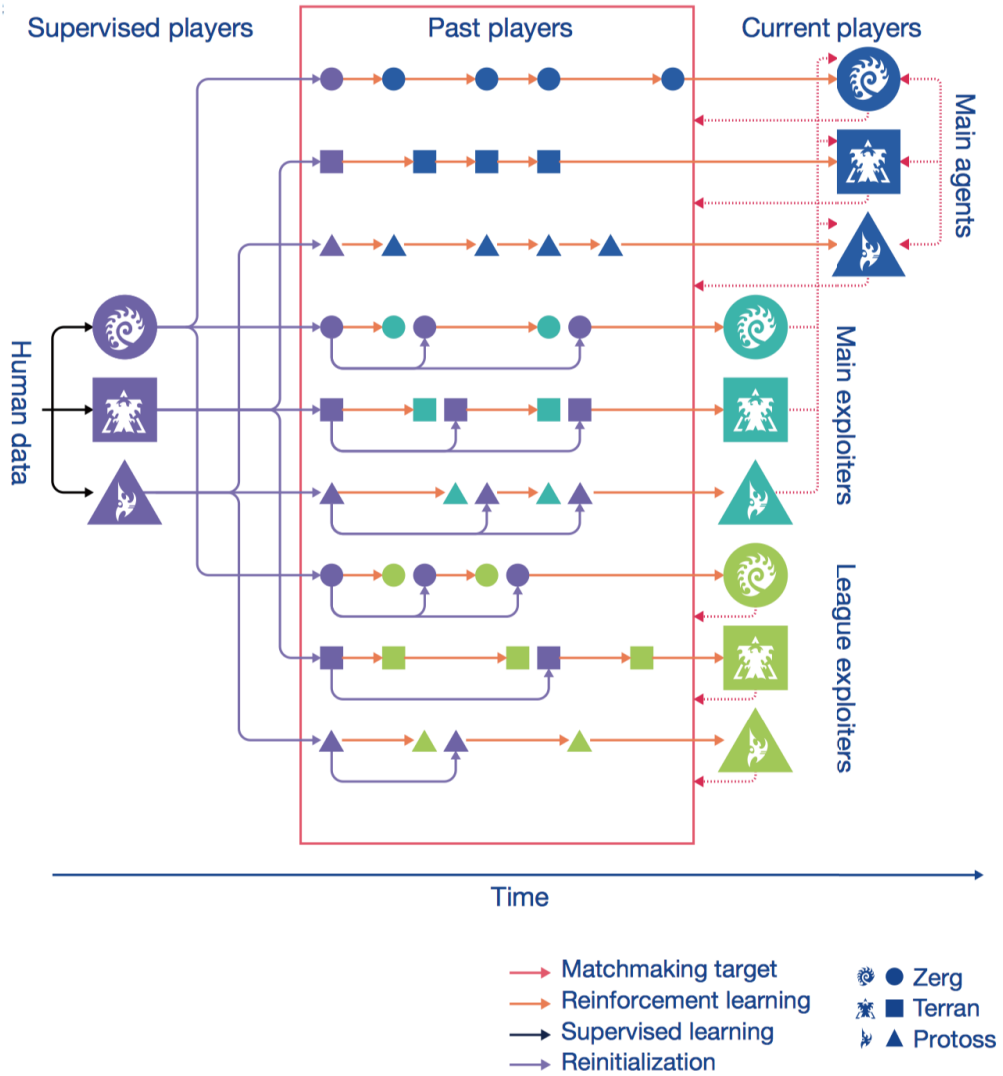


图 2：Commander Neural Network

启元星际指挥官使用了启元世界自主设计的网络模型 Commander Neural Network。这是一套类似于图像处理领域中 ResNet 等模型一样的标准化模型。它将输入、输出、奖励标准化，可以端到端的解决基于空间和时间特征关联的，博弈对抗类型的强化学习问题。

三、高效的智能体训练平台

启元世界从建立之初，就确定了一横两纵策略。通过竞赛项目或星际争霸类的研究型项目积累技术经验，并沉

淀到智能体训练平台中。通过平台的转化，输出给业务项目。而通过业务项目积累的行业经验再次反哺到平台中，为日后的项目节约解决方案上的成本。

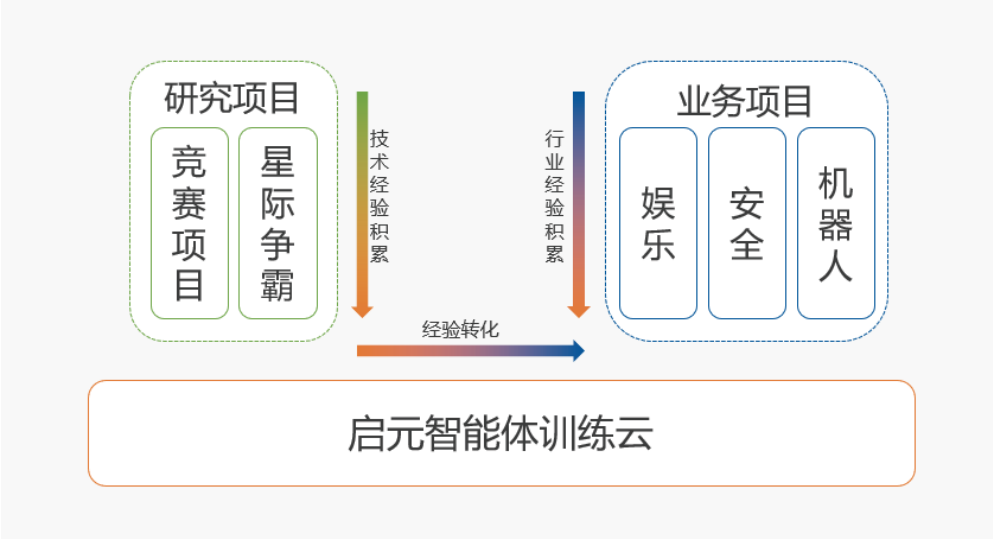


图 3：启元智能体训练云

启元世界面向于产业级大问题提供解决方案，通过大量案例的分析，将该类问题分解成两个关键要素： 问题规模和业务效果，其关系如下图所示:

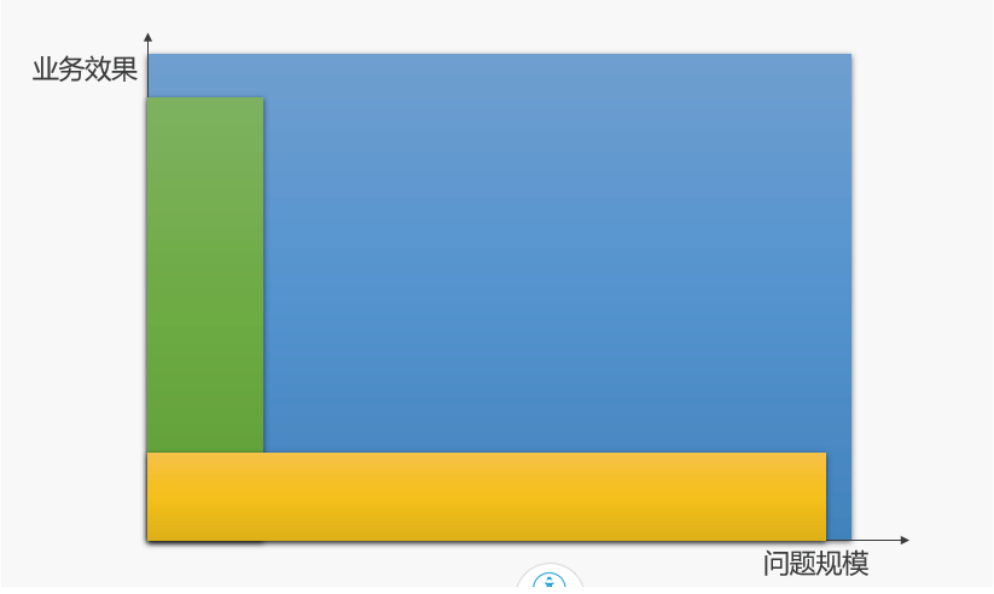


图 4：两个关键要素

问题规模主要依靠分布式技术解决，而业务效果主要依靠算法能力。启元世界在算法能力之上进行了进一步的抽象，总结了在相关领域的最佳实践，进一步提升了算法结合产业的效果。

根据上述要素，我们进一步分解，设计了五层系统架构，如下图所示:

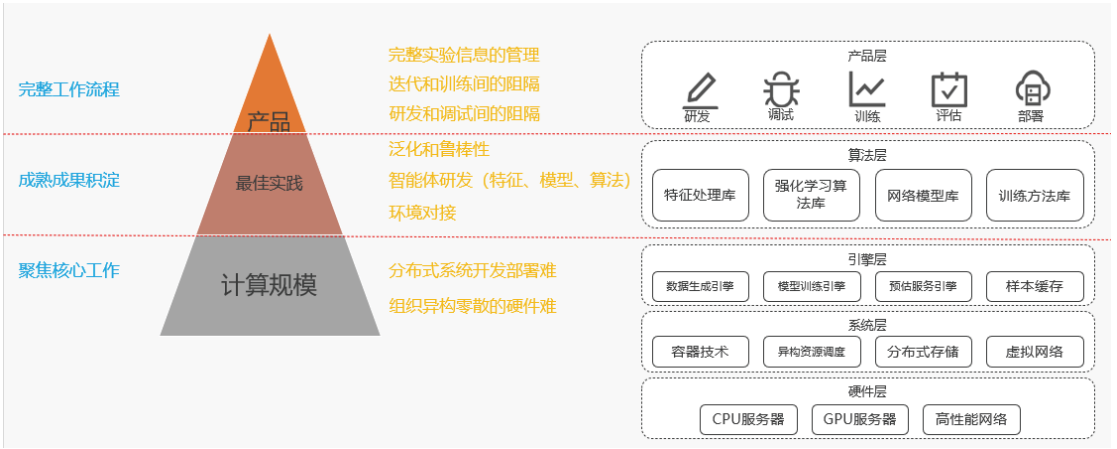


图 5：对要素进行分解，设计一个五层系统架构

通过硬件架构的设计，提供了高计算密度、低通信延时的集群部署方案。通过分布式操作系统层的抽象，将众多服务器抽象成如同一台电脑的、可灵活分配的算力模型。最后通过四大引擎的计算抽象，将大规模算力转化为数据生产和消费的能力，从而简化了大规模算法的实施成本。

算法层涵盖了算法实施的全过程，包括原始数据处理的特征库、支撑强化学习优化方法的算法库、上面提到的标准化神经网络模型结构及组件的模型库，以及优化智能体鲁棒性的训练方法库。

最后，启元还将上述技术结合十余年的系统建设经验，构建了涵盖工程师从研发、调试、到模型训练、评估、部署，全周期的产品解决方案。

上述架构的运行逻辑如下图:

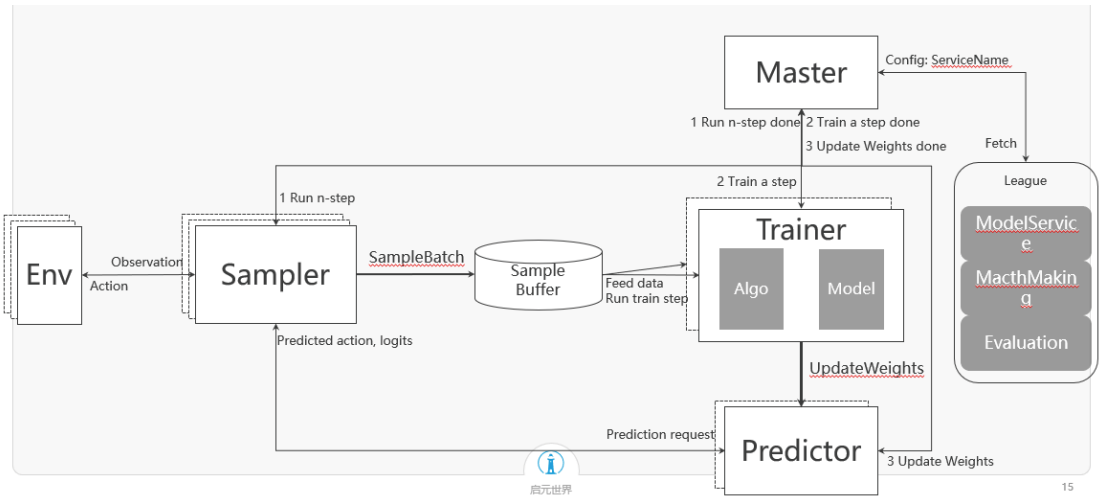


图 6：循环逻辑

四、启元智能体训练平台的技术优势

启元世界拥有业界一流的系统设计和实施能力，根据强化学习计算特点设计的计算模型、网络通信优化、数据 Pipeline 优化、计算优化、Placement 优化等多种手段，将平台的计算能力推高到开源实现的 10 倍以上。

另外，启元世界还开创性的提出了对抗博弈类的标准网络结构，Commander 网络模型（群体指挥控制）和 Hero 网络模型（多智能体协作）。配合标准化的训练方法（模仿学习、自博弈训练、联赛机制），可以产出达到甚至超过人类水平的智能体。

启元智能体训练平台于其它竞品的对比如下图:

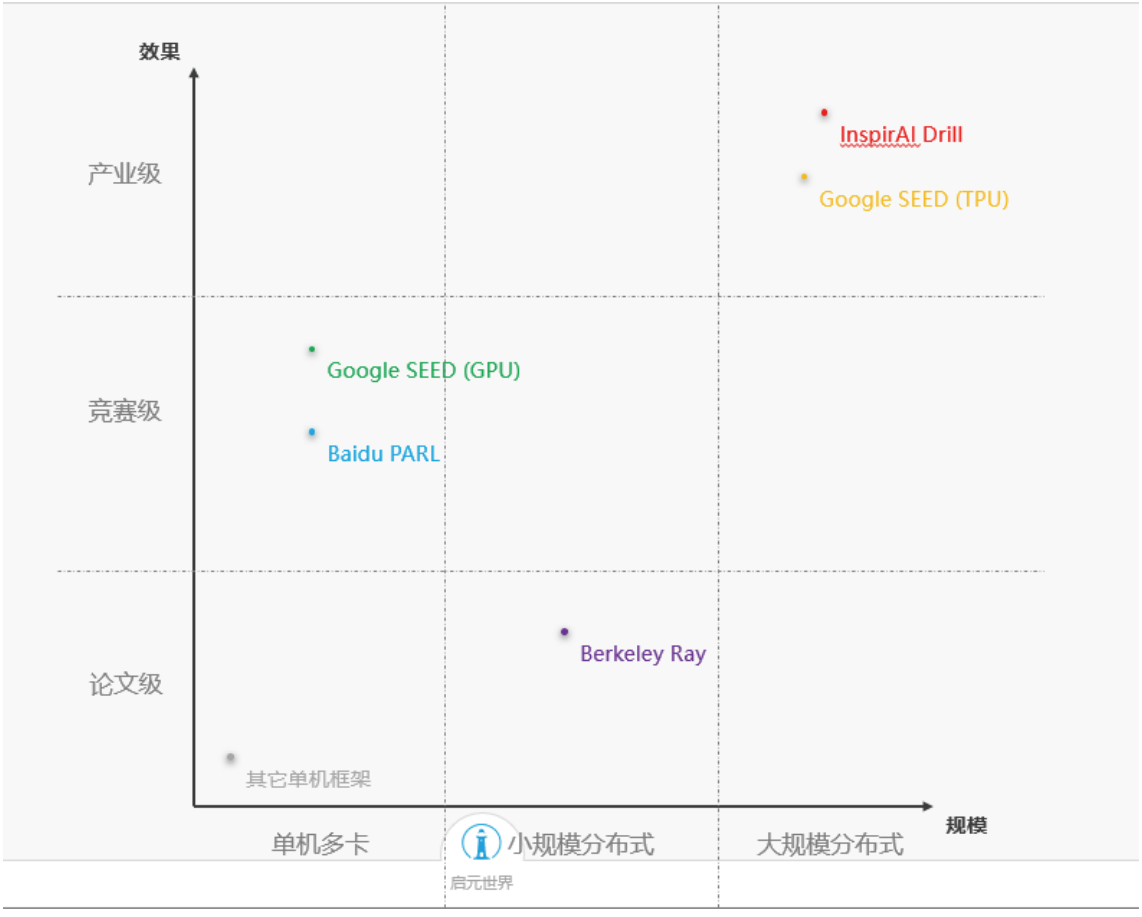


图 7：竞品对比

启元智能体训练平台和 Google SEED(TPU) 同属于产业级大规模分布式强化学习计算平台，因为二者都已经解决了该领域最为复杂的问题。如果 SEED 部署在 GPU 集群上，最多支撑单机 8 块 GPU，也就退化到单机多卡的领域，与其类似的是 PARL，从发布的信息上看，它们更多的是在解决竞赛问题。Berkeley Ray 可以较大规模的部署，但受限于分布式多进程的计算模型及实现，其规模扩展也受到限制，并且多用于研究领域。而其它的单机框架更是只能解决非常小规模的问题。

五、启元智能体训练平台的商业应用

启元世界依托于成熟的平台技术，凭借高强度的计算性能、大规模生产问题的算法能力、低成本的算法实施方案，可以将强化学习技术应用于诸多行业，如：智慧防务、数字娱乐、金融科技等。

图 8：平台产品：跨行业通用技术，市场空间极为广阔

六、结语

启元世界以星际争霸为平台，通过三年的潜心研究，将公司的强化学习技术水平推升至世界一流。并通过标准化、平台化，赋能于众多行业，为我们的生产生活带来巨大的社会和商业价值。