**分布式多智能体系统的关键技术及应用场景**

# 摘要

分布式多智能体系统（MAS）作为分布式人工智能的重要分支，近年来在解决复杂计算问题中展现出显著优势。MAS由具备自主决策和学习能力的智能体组成，通过协同合作完成任务，其灵活性和适应性使其广泛应用于计算机网络、机器人、复杂系统建模、城市与环境建设及智能电网等领域。

在关键技术方面，MAS涉及智能体通信（如言语行为、消息传递和黑板机制）、开发平台（如基于BDI模型的JaCaMo）、控制技术（包括图论、博弈论和强化学习）以及学习机制（如集中式与分布式学习）。这些技术为智能体的协同、决策和适应动态环境提供了支持。

应用场景中，MAS在云计算中优化资源分配与负载均衡，在社交网络中预测用户行为，在网络安全中检测入侵行为，在机器人领域实现协作与路径规划，在智能电网中平衡能源供需，并在城市管理中优化交通与物流系统。这些应用展示了MAS在提升效率、可靠性和自适应性方面的潜力。

未来研究方向包括提升系统的可扩展性、鲁棒性和安全性，探索多智能体强化学习（MARL）与人类工作流的结合，以及解决跨学科集成中的文化和组织障碍。MAS的进一步发展将为复杂系统的智能化管理开辟新路径。

# 引言

近年来，分布式人工智能(DAI)因其能够解决复杂的计算问题而受到了学术界的极大关注。基于用于解决任务的基本方法，DAI算法被分为三类，即并行人工智能、分布式问题解决(DPS)和多智能体系统(MAS)。并行人工智能涉及开发并行算法、语言和架构，以利用任务并行性来提高经典人工智能算法的效率。DPS涉及将任务划分为多个子任务，每个子任务都分配给一组协作节点中的一个节点，这些节点被称为计算实体。计算实体具有共享的知识或资源，并且与其他实体有预先定义的通信，这反过来限制了它们的灵活性。

多智能体系统(MAS)是本文的主要研究对象，它由被称为“智能体”的自主实体组成。与分布式处理系统中的计算实体类似，这些智能体协同完成任务，但它们具有更高的灵活性，这是因为它们具备学习和自主决策的能力。智能体通过与相邻智能体或环境的互动来学习新的情境和行动。随后，智能体利用其知识来做出决策并在环境中执行行动，以完成分配给它们的任务。这种灵活性使多智能体系统能够适用于解决包括计算机科学、土木工程和电气工程在内的众多学科中的问题。开发多智能体系统需要应对一系列复杂的挑战，例如智能体之间的协调、学习以及安全性。现有的相关综述文章对多智能体系统(MAS)的讨论较为局限，因此我们对多智能体系统的关键技术和应用场景进行了全面的概述。

# 关键技术

## 智能体通信

智能体之间的通信已经被研究了50多年[38]。三种广泛使用的沟通方式包括：

* **言语行为**：在研究言语行为交流的先驱John Austin[38]中，他发现一些言语动词或句子，被称为言语行为，会改变物理环境，例如，在适当的情况下，如果一个适当的人说“我现在让你成为夫妻”，那么这句话就会通过定义新的角色和条件来影响物理环境。智能体可以作为说话者，产生话语来改变听者的信念 [39]。智能体通过执行根据某些语言语法被感知为话语的主要动作来执行话语[40]。可参阅[41]以进一步阅读。
* **消息传递**：在此方法中，智能体直接相互传递消息，如图7a所示。智能体使用点对点通信或广播通信与其他智能体进行通信。在前一种情况下，智能体A知道agent B的地址，可以直接与agent B对话。在广播通信模型中，智能体A向其所有相邻节点发送消息。为确保消息的可理解性，参与通信的各个智能体必须采用一致的结构，这一点将在下文进一步详述。
* **黑板**：在这种通信方式中，智能体可以通过一个名为“黑板”的中央存储库相互协作共享数据。每个智能体将其数据存储在黑板上，其他智能体可以读取这些数据。为了控制智能体的访问权限，黑板使用控制知识。每个智能体可以访问控制知识中定义的多个数据。

消息语义对于确保彼此交流的智能体对交换的数据有相同的理解至关重要。对于异构智能体而言，这尤其具有挑战性。一个简单的例子是，当智能体A将温度信息发送给智能体 B时，其值为12°C那么智能体 B不应将其理解为 12°F。智能体通信语言(ACL)旨在解决上述挑战。ACL为所有智能体提供了一种独特的消息格式和语义框架，以便它们能够进行通信并解读接收到的消息。[42] 将 ACL分为两个主要类别，即程序型和声明型。在程序型 ACL中，智能体之间的通信被建模为程序指令的共享。在声明型 ACL中，声明性语句用于指定定义、假设和断言。

“智能物理智能体基础”(FIPA)提出了一个适用于智能体的全面的ACL框架，该框架目前已被广泛应用在大多数多智能体系统的实例化中都得到了广泛应用。

## 主要平台

自适应多智能体网络是在主平台上开发的，它能够实现信息流、智能学习以及其中包含的智能体和元素的实时决策。这种用于对智能体的结构、行为和社会模型进行建模的抽象框架可以通过适当的智能体导向编程 (AOP) 来定义，AOP 是面向对象编程的一种特例[43]。这些平台最初受到了自适应组织模型的启发，后来在软件工程领域提出了更结构化的技术[44]。计算能力的迅速提高提供了更定制化的模型，如 O-MaSE，它利用了元模型的三个概念、方法片段和基于方法工程概念的指导原则[45]。这些平台因具体情况而异，要么针对特定的应用领域，例如微电网[46]，或者灵活的通用平台[47]。这些平台中的大多数是通过添加推理和认知模型而设计的，例如程序推理系统(PRS)和/或信念-愿望-意图(BDI)模型[49]。PRS有助于对过程进行推理，使智能体能够与动态环境进行交互，并使用程序来选择意图。这些程序在能够有助于实现某些目标时会被触发。在最常见的 BDI模型中，智能体的行为是根据其信念、目标和计划来定义的。在这个模型中，解释器负责根据从环境接收到的反馈以及管理智能体的意图/行动来更新这些特征。这些模型在将人工智能作为可插拔组件[50]或在元级计划[51]中集成方面表现出了极大的成功。面向多智能体的编程(MAOP)平台，如JaCaMo，采用了基于智能体、环境和组织维度这三种概念的结构化方法[52]。它们还成功地与物联网(ITS)相结合，在以人类为中心的环境中提供了自适应应用[53]。系统集成是另一种开发元模型以管理具有不同子系统的多智能体系统的有效方法[54]。在过去几十年里，提出了数十种此类方法及其旧版本的更新版本。为了从提出的众多 AOP 技术和平台中选择最佳选项，可以根据基本平台属性、可用性、可扩展性、稳定性、操作能力、安全管理和其在实践中的适用性等方面进行比较[48]。还需要根据其架构债务以及其对软件系统健康状况的长期影响对其进行研究和评估[55]。关于面向智能体的方法的评估框架在参考文献[56]中进行了回顾。尽管文献中提出了多种面向方面编程(AOP)方法，但它们在对模型成本、偏好、时间、资源以及持续性操作等方面的设计推理和决策模块方面仍存在诸多不足[57]。因此，开发人员仍然不愿转向这些平台，而是倾向于对现有编程语言进行少量修改后继续使用原有的代码[57]。

## 控制技术

在相关文献中，针对多智能体系统控制问题，采用了多种技术，包括图论、博论、控制论和机器学习等，这些技术被认为是用于此目的最常用的方法[58][59]。此外，还有其他算法，如优化算法和生物启发算法，也被应用于这些系统的集体行为研究[60]。

图论被用于确定多智能体系统的结构可控性[61]。在这个领域中，最常用的技术之一是利用图拉普拉斯矩阵来研究多智能体系统的动态和收敛速度。在文献中，还通过将边的状态定义为边的相对状态差的方式来使用边拉普拉斯矩阵来研究一致性问题[62]。图拉普拉斯谱具有第二小和最大的特征值，它们的比值在多智能体系统的控制中起着重要作用[63]。在最近的文献中，图论主要用于对网络动态进行初步分析，而其他补充方法则用于达到共识状态[64]。图论的应用并不局限于使用拉普拉斯矩阵，而在某些情况下，仅使用邻接矩阵和度矩阵来研究网络动态及其同步[65]。

博弈论是另一种用于对多智能体系统中理性协作智能体的动态行为和决策过程进行建模的流行技术[66]。在大多数分布式游戏中，主要目标是达到纳什均衡，并同时优化自身的性能指标[67]马尔可夫博弈或随机博弈是序列多智能体博弈的初始应用实例，它们可以通过动态规划(DP)Q学习或线性规划技术来解决。智能体的收益和奖励/效用中的不确定性也可以使用贝叶斯斯塔克勒格博弈来建模[68]。进化博弈是博弈论技术的另一种变体，用于对智能体的集体行为进行建模，其中有限理性会反复寻找平衡点[69]。大多数多智能体系统问题都是通过控制理论及其变体(如自适应控制)来解决的。例如，分布式模型预测控制被广泛应用于对不同类型的动态进行建模，其目标包括调节、跟踪或系统中的经济考量。关于多智能体系统中的在线学习控制机制的调查报告已发表。神经自适应最优控制被认为是控制复杂多智能体系统(MAS)的最流行技术之一，其原理是通过求解相关矩阵方程来实现，例如耦合的里卡蒂方程或耦合的哈密顿-雅可比方程[70]。在相关文献中，其他传统的优化技术，如变分不等式[71]、对偶理论[72]以及交替方向乘子法(ADMM)[73]，也被用于优化多智能体系统的操作。

强化学习是应用于自适应多智能体系统同时学习与控制的著名半监督学习技术之一[74]多智能体强化学习(MARL)技术通常用于基于部分信息(如从先前的行动和决策中获得的奖励和惩罚反馈)来感知环境[75]。基于强化学习开发的其他技术包括Q学习和策略梯度技术，它们试图学习智能体的最优策略。具有网络化智能体的MARL技术是MARL算法的一个特殊案例，在这种情况下，智能体可以在时变通信拓扑结构中与相邻智能体进行通信[77]。这些算法是为合作和非合作环境而开发的。在多智能体系统中使用强化学习非常具有挑战性，因为存在联合动作空间和由多个自主决策者生成的动态这使得环境非平稳且难以感知[78]。使用策略的集合是为设计对环境变化和非平稳动态具有鲁棒性的控制框架而提出的另一种技术[79]。[80]提出了 QMIX算法，该算法通过对各参与者的联合价值函数进行线性分解，同时确保局部和全局最大值函数在标准范围内保持单调递增/递减的特性Q-learning。在这些环境中，由于状态-行动对的数量不断增加，近似价值函数或最优策略变得非常困难。针对这一问题，主要的解决方案之一是应用深度强化学习(DRL)，它在强化学习迭代的学习过程中融入了深度神经网络[80]。这些技术的两个著名变体包括深度 Q学习[82]和深度确定性策略梯度(DDPG)，它们是基于带有重放缓冲区的演员-批评网络设计的[83]。像DDPG 这样的策略梯度技术在 MAS 中表现更好，因为其近似与系统动态的关联性无关。在 MAS 中使用基于强化学习的控制机制的其他挑战包括信用分配问题，这反映了缺乏追踪智能体的行动及其对系统结果的影响[84]。这个问题可能导致懒惰的不活跃智能体不愿意为学习系统动态做出贡献。为解决这一挑战而提出的方案之一是使用反事实基线的多智能体策略梯度(COMA)方法，该方法会固定其他智能体的动作同时对单个智能体的动作进行边际化处理[85]。

## 学习机制

在自适应多智能体网络中，主要的框架之一就是其学习机制。大多数最新的学习框架都是在线式的，能够帮助系统了解其环境的动态以及对这些变化的最佳应对策略。针对多智能体系统学习文献，人们研究了诸如知识获取级别和学习技术等各个方面[86]。智能体可能拥有完全的自主学习能力，并能与其他智能体共享其知识，或者可能仅限于仅与中央学习者进行交流和共享其状态[87]。如果从系统层面进行学习，那么一个智能体或主要管理者会学习整个系统中所有智能体的策略。在这种情况下，学习者能够完全观察到涉及的智能体的状态，而无需详细关注单个智能体的行动。在这种学习机制中，信息从分布式智能体收集，并传递给系统的中央学习者。这有助于获得有关系统动态的高级信息，而不会陷入协调多个学习者之间信息流的困难之中。集中式学习和培训可以与中央决策者相结合，也可以与分散式激励相结合[88]。在第二种类型中，集中式学习者使用引导分布式执行者的准则来学习价值函数[89]。集中式学习机制存在一些问题，例如状态过程和学习过程的复杂性。它们也是基于一些不切实际的假设而开发的，比如能够持续且完整地获取所有执行者的信息。集中式学习的另一个挑战是其对学习者的故障的脆弱性，需要中央学习者具备大量的计算和内存资源，并且对于拥有数千个分布式执行者的大型系统具有可扩展性[90]。通过这种学习方式，可能会导致所有执行者都采用相同的策略的同质团队学习，或者每个个体执行者都有独特行为的异质学习[91]。为缓解上述列出的挑战，该平台的其他变体包括 QMIX的全局和局部的混合成分[92]。通过最大熵 RL技术和策略提炼进行协调采样是为改进集中式学习机制而提出的其他解决方案[93]。还有一些混合技术，即智能体各自独立学习，然后共享一个集中式的共同知识记忆来整理和存储它们的知识[94]。在分布式学习技术中，每个智能体负责自己的学习[95]。在这些学习框架中智能体的可观测性有限，只能探索其周围环境，且无法实时学习系统的整体动态大多数这些学习技术都与控制机制的决策过程相结合。基于角色的学习技术是文献中的另一个趋势，其中复杂的任务被分解为不同的角色。RODE技术就是这种学习的一个例子，它利用智能体聚类来发现角色和所需的学习组。基于特定事件或系统运行过程中的离散或连续时间更新，可以启动学习和更新已有的模型。文献中应用的一些常见学习技术包括强化学习、监督学习、深度学习、博弈论、概率论、群体系统、应用逻辑、进化算法或它们的组合等[96]。由于这些技术大多用于同时对多智能体系统进行学习和控制，因此将在下一节中对其进行回顾。一些提出的用于学习的技术是在获得初始领域知识后启动的。在文献中，这些技术被定义为迁移学习方法[86]。

# 应用场景

## 计算机网络

由于新技术的出现以及联网设备的大量增加，计算机网络的复杂性显著提高。智能体技术被广泛应用于克服这困难。由于多智能体系统在网络中的应用范围广泛我们进一步将其分为四个子类别：

### 云计算

云计算能够实现对可配置系统资源(如CPU、GPU和内存)以及计算服务(如服务器、数据库、网络和软件)的普遍访问，这些资源通常通过互联网提供。云计算利用虚拟化作为底层技术为用户提供服务。通过虚拟化，一台物理机被多个客户共享为多个虚拟机(VM)，每个虚拟机都模拟一台独立的机器。与传统的每个用户都拥有自己专用资源的方式相比，云计算具有多项优势:

* 降低货币成本:云用户租用他们所需的各种资源，这些资源由云服务提供商进行管理。这样就无需再自行购买和维护资源，从而降低了货币成本。
* 可靠性:每个资源都有多个重复备份，这使得云计算能够抵御资源故障，从而提高了其可靠性。如果出现例如 CPU 的故障，其相关任务可以被转移到其他类似的资源上。阐述了云计算中涉及的几个关键概念:管理云资源、进行通信以及核算每个用户对资源和/或服务的使用情况[1]、[2]。

在[3]中，提出了一种基于智能体的框架，用于在云端执行“任务包(BoT)，即一系列独立的任务。一组智能体收集有关每个特定任务的可用资源和云提供商的信息。智能体将每个客户与合适的云提供商进行匹配。然后，客户和云提供商进行协商，就提供给客户的资源以及应支付给云提供商的价格达成协议。通过模拟以及从终端用户的角度考虑的成本，研究了任务的分配、释放和执行时间。仿真结果表明，该智能体系统在任务分配方面比亚马逊 EC2 云资源分配方法更成功。

在任务分配过程中，必须考虑每个资源的负载情况，因为负载过重会导致服务接收出现长时间延迟。[4]提出了一种基于智能体的负载均衡框架，以平衡虚拟机的负载。云提供商定义了一项策略来控制虚拟机的负载。所提出的方法得益于三个智能体，即:负载智能体、迁移智能体和通道智能体。负载智能体通过监测虚拟机的负载来确保其负载与所定义的策略相匹配。为了监测虚拟机的负载，负载智能体会请求通道智能体启动迁移智能体。迁移智能体是移动于网络和收集关于资源信息的移动智能体。这些信息会报告给通道智能体，该智能体负责控制传输策略和选择策略。然后，通道智能体将收到的信息转发给负载智能体。基于收到的信息，负载智能体通过将新任务分配给负载较低的虚拟机来平衡虚拟机之间的负载。

多智能体系统被应用于云环境中的资源监控[3]、安全[5]资源发现[6]以及自动服务管理[7]等方面。有关多智能体系统在云环境中的应用的全面综述可参见[8]。

### 社交网络

随着互联网用户的不断增加，社交网络的普及程度呈指数级增长社交网络由参与者(如用户、群组和服务)组成[9]。社交网络的复杂性源于其动态性，即大量参与者加入或退出网络，或与其他参与者建立新的联系，以及其广泛的用途和服务。多智能体系统可能是解决社交网络复杂性问题的潜在方案。

[10]提出了一种基于智能体的方法来a预测社交网络(如推特)中的用户行为，例如点赞、发布内容和关注等。作者提议使用多个智能体，即参与者，它们分布在社交网络中，以收集用户行为的数据集。然后，这些智能体会对每个特定用户的数据进行主题和情感分类，这些分类结果随后用于构建用户档案。最后，将用户档案输入到预测系统中，该系统能够预测用户的未来行为，包括点赞、主题、回复、发布内容和分享等。

社交网络不一定非得基于网络形式。只要人类出于特定原因聚集在一起进行互动并分享信息，任何这样的场所都可以被视为一个社交网络。[11]将学校视为一个社交网络，并应用多智能体系统来帮助学校管理者了解向不同学校项目(如体育、学术和文化活动)分配资金与学校表现之间的关系。学生、教师和学校部门被视作智能体，它们共同构成了这个社交网络。例如，就是这所学校。每个学生与其他学生或教师之间的互动、过去几年的资金分配政策以及学生在当前和过去几年的表现都被组织成一个层级结构。然后，这个结构会被输入到一个学习功能中，该功能会评估资金分配政策与学生学业成绩之间的关系。

### 安全

自 2002 年以来，关于多智能体系统在网络中的安全应用的研究已展开[12]。多智能体系统是解决网络安全问题的有效方案，因为它们能够主动了解并从而检测出新的安全威胁。

在文献[13]中提出了一种基于自主智能体的入侵检测系统(IDS)。所提出的该入侵检测系统由五种智能体组成，分别是集合智能体、检测智能体、决策智能体、响应智能体和协作智能体。集合智能体从网络中收集简单网络管理协议SNMP)和路由表的内容，并将这些数据发送给检测智能体后者使用误用和异常检测引擎来检测异常的数据包或通信。检测引擎的结果被输入到决策智能体中，该智能体会判断网络中是否发生了恶意活动，如果存在，则决定采取何种适当措施来减轻其影响。然后，所决定的行动会传递给协作智能体，该智能体是一种移动智能体，它将决策传达给响应智能体。响应智能体在网络中执行相应的操作。

基于智能体的入侵检测系统是文献中多智能体系统在网络安全领域的主要应用。然而，多智能体系统也用于其他安全应用。[14]提出了一种基于智能体的安全方法，以保护用户免受劫持攻击。在该攻击中，攻击者在目标用户网络浏览器的闲置标签页上打开随机网站。两个智能体监控打开的标签页，并收集五元组元素，即文本、图像、URL、标题和网站图标，即网页标题图标然后，智能体将运行在打开标签页上的操作的指纹与现有攻击签名进行比较，以检测劫持攻击。使用 JADE模拟器进行了模拟(见第七节)。模拟结果表明，使用智能体，误报率和漏报率低于非智能体方法，攻击检测的准确率达到了 91%。

未来，MAS 有可能被用作解决诸如信任和密钥管理等复杂安全任务的解决方案。此外， MAS 可用于克服物联网(IoT)的安全挑战。[15]中提出了一项关于基于智能体的入侵检测系统的综合研究。

### 路由

路由指的是根据特定指标(例如源节点与目的地之间的跳数)为数据包从源节点向目的地寻找路径的过程。自1998年以来，使用智能体进行路由是多智能体系统研究中的首批应用之一[16]。此后，随着路由协议面临的新挑战不断出现，多智能体系统也不断发展。

[17]提出了一种针对车辆自组织网络(VANET)的去中心化智能体式路由协议，该协议由三组智能体组成。第一组智能体由车辆承载，用于监测车辆参数，例如速度、位置等。第二组智能体由路边基础设施承载，监测车辆之间的通信以及车辆与路边基础设施之间的通信。第三组智能体通过模拟车辆、路边基础设施及其通信来创建一个虚拟环境，形成一个图。智能体通过应用蚁群算法在图中寻找通往目的地的短路径，从而通过向不同车辆发送不同的路径来避免该区域的拥堵。实验结果表明，基于智能体的这种方法的行程时间比其他已研究的路由方法要短。

路由中的一个关键挑战在于所选路径的可靠性，即要确保该路径不会频繁出现故障。[18] 为无线传感器网络提出了一种可靠的路由协议。该方法得益于智能体在转发数据包和汇总传感器数据方面的作用。首先，汇聚节点(即无线传感器网络中的中央控制器)广播一个数据包，以查找节点的可用资源，用于执行特定任务，这些资源用于划分网络。然后，汇聚节点建立移动智能体，以收集和汇总多个区域内的传感器数据。通过跟踪每个移动智能体访问的传感器，汇聚节点根据每个区域中传感器到汇聚节点的距离，为每个传感器生成一个排序的传感器列表，作为通往每个传感器的可靠路径。仿真结果表明，与传统路由协议相比，基于智能体的路由协议的能耗更低，数据包丢失率也更低。

为了进一步提高可靠性，[19] 提出了一种可靠的多播路由协议，该协议由两组智能体组成。第一组是移动智能体，它们在网络中移动以收集节点的能量、带宽、移动性和内存状态。这些信息用于为每个节点计算可靠性因子(RF)。然后，该因子传递给第二组智能体，后者利用具有最高 RF的节点建立多播骨干网。仿真结果证明，基于智能体的方法比不使用智能体的方法具有更高的数据包传输率和可靠性。

## 机器人

关于机器人领域的智能体的研究已经持续了近二十年，首篇相关文章于 1996 年发表，文中阐述了智能体在机器人技术中的利弊 [20]。

[21] 认为，在机器人技术领域存在两个主要挑战:(1)机器人之间的协作与协调，以及(2)规划其移动轨迹。随后，作者提出了一种利用硬件和软件智能体来克服上述挑战的方法硬件智能体指的是构成机器人的物理硬件，而软件智能体则是决策、路径规划、任务管理和通信智能体。一个硬件智能体利用其传感器(例如摄像头)来捕捉环境中的图像。然后，一个通信智能体将这些图像发送给图像处理智能体。后者对图像进行处理，以确定机器人的位置和环境中的障碍物。然后，这些信息被发送给决策制定智能体，该智能体会找到一条障碍物最少的路径以到达目的地。实施结果表明，所提出的方法能够检测到障碍物，并找到一条优化的路径(没有任何障碍物)以到达目的地。

机器人可能会被部署在非确定性的动态环境中，这增加了其决策过程的复杂性。为了研究这种复杂性，[22] 提出了一种基于智能体的足球机器人。智能体(即球员)被分组组成团队。团队中的智能体通过与环境的互动来学习有关对手团队和可能行动的知识，然后它们会将所学策略与团队中的其他智能体共享。强化学习与概率神经网络(PNN)结合使用，以提高智能体所做最终决策的准确性。实施结果表明，与非智能体团队相比，智能体团队中的智能体能够预测出正确的行动，从而提高了球的控制率(这是足球中的一项重要性能指标)

在[23]-[25]中讨论了智能体在机器人技术中的更多应用。

## 复杂系统建模

对复杂动态系统的建模成本高昂，并且会带来大量的处理开销这是因为需要强大的建模平台以及较高的复杂性。而智能体所提供的灵活性、自主性和可扩展性使得基于智能体的建模(ABM)成为一种低成本且资源消耗少的复杂系统建模解决方案。ABM 采用基于规则的方法来对环境进行建，这与使用方程的其他建模方法不同。ABM 的最重要优势包括[26]:i)能够整合多种建模方法，ii)在建模大规模自治系统时具有灵活性，ii)对预定义知识的灵活性，因为智能体可以通过从环境中学习来获取知识，iv)可以并行执行从而加快建模过程，以及v)能够探索由于智能体的主动性而产生的涌现行为。

[27]提出了一种智能体模型来模拟供应链。供应链中的每个实体都单独进行建模，拥有自己的政策，并能够定义与其他实体的互动方式。所提出的方法包含两组智能体，即:规划智能体和物理智能体。客户和供应商使用规划组中的六种智能体来协商并就产品的价格达成协议。规划组中的智能体如下:i)需求满足智能体负责管理客户需求;ii)物料资源规划智能体，与生产商沟通并采购产品;iii)需求预测智能体，根据当前和历史需求情况预测客户需求;iv)主计划智能体，汇总生产计划;v)生产计划智能体，由主计划智能体汇总的计划进行分解;vi)调度智能体，对多个智能体中的任务进行调度。物理组中的三个智能体执行物理任务，即接收和存储原材料制造以及向客户交付产品。

[28] 中提出了一种 ABM(智能体行为模型)，用于研究城市配送中心(UDC)的使用情况。在智能城市中，该配送中心通过优化和调度用于货物运输的路线，来集中管理产品配送，从而控制城市拥堵污染、配送时间和可靠性。

在该方法中，移动自动化系统由卡车、货物、客户、用户数据中心以及城市车辆组成。停车场中未获许可或超时停放的车辆会导致卡车的停车问题，从而造成货物配送的延误。用户数据中心的智能体在设计配送路径时会考虑到这一挑战，以进一步缩短配送时间。与传统管理系统相比，该方法为店铺业主最大限度地降低了配送成本为配送公司最大限度地提高了配送利润。

基于[29]中所提出的观点，现有的电力行业建环境无法有效地对复杂的可扩展电力网络进行建模，例如拥有大量不同能源来源和用户的智能电网。智能体模型是一种适用于智能电网的有效分布式建模方法，这使得智能体模型成为该领域的一个有趣的研究方向[30]。参考[25]和[30]以获取关于智能体模型的更详细讨论。

## 城市与环境建设

近年来，利用智能体来管理城市和建筑物的研究受到了研究人员的极大关注。在城市中，货物的无序分配会增加成本、污染和拥堵。[31]提出了一种基于智能体的方法来解决这一挑战，该方法使用了六个智能体，分别是:RFIDG、零售商、供应商、运输商、网络和城市智能体。RFIDG智能体使用 RFID标签来管理资源供应客户向零售商或供应商智能体发送购买商品的请求。收到请求后，零售商或供应商智能体会搜索其数据库以找到所请求的商品。接下来货物会被发送给运输商智能体，以便将其送达客户手中。网络智能体确定减少城市拥堵的最优路径，以便运输商智能体能够将货物送达客户。城市管理员智能体会将交易的相关政策和规则(例如商业和货物运输)告知双方，例如供应商和客户。

城市面临的另一个重大挑战是控制和管理不断扩张的都市区的交通系统和交通状况。[32]提出了一种基于智能体的方法来解决这一难题。该方法考虑了诸如票价和乘客满意度等参数，用于定义和分析交通模型。他们使用了两组智能体:旅客和车辆。这些组中的智能体共享关于拥堵和交通状况的信息，其他智能体会利用这些信息来决定一条低拥堵的通往目的地的短路径。

MAS 还被用于管理建筑物。在[33]中提出了一种基于智能体的建筑物供暖管理方法。智能体的主动性和灵活性使它们成为管理由建筑物周围分布的异构供暖设备和传感器组成的供暖系统的有效解决方案。一个需求智能体会检查建筑物内的温度，并将数据传递给燃气加热器、缓冲器和热泵智能体。后者的智能体会使用从需求智能体接收到的数据以调节建筑物的温度。作者们在一套公寓中实施了所提出的方案，并将其与集中式方法进行了对比，作为基准测试。实施结果表明，采用智能体方法相比集中式方法，每日的电力消耗显著降低。

关于该领域的进一步阅读内容可参见文献[33]和[34]。

## 智能电网

在相关文献中，智能电网中的“智能体”被用于应对一系列挑战包括平衡发电量与用电量、在能源消费者与生产者之间就能源价格进行协商、在家庭储能设备中储存能源以及进行能源恢复。

[35] 提出了一个基于智能体的服务模式。例如，为智能电网提供能源和恢复功能，以支持分布式能源存储。该系统使用两个智能体，即:切换智能体和分布式能源存储智能体。切换智能体平衡能源负载，并检测并隔离故障，即由于某些能源生产者或能源分配系统中的故障而导致的停电。能源存储智能体根据电网是否连接到智能电网或是否处于隔离状态，为电网提供能源。所提出的方法通过有效的恢复来降低系统损耗，并实现动态孤岛化，以恢复正在断开连接(孤岛化)的电网部分。

智能电网的参与者要么是能源生产商，要么是能源消费者。

能源生产商旨在通过以更高的价格出售能源来增加利润。为了实现这一目标，[36] 提出了一种储能管理方法。每个能源生产商(智能体)都拥有一个储能设备，并旨在通过分析包括客户和其他能源生产商在内的其他智能体所生成的价格信号来增加自身利润。每个智能体都会将储能使用情况记录在一个独特的存储方案中。该智能体利用博弈论方法对存储方案进行分析，以预测未来的使用情况，从而决定是出售还是储存其产生的能源。

参考文献[37]以进一步了解智能电网中多智能体系统的应用情况。

# 未来方向

尽管分布式多智能体系统（MAS）取得了显著进展，但其在各行业中的集成和应用仍面临各种挑战。最突出的挑战之一是这些系统的可扩展性。随着业务的增长和任务复杂性的增加，保持高效性能变得至关重要。随着更多智能体加入系统，潜在交互的指数增长需要有效的资源分配，以防止瓶颈并确保平稳运行。此外，可扩展性不仅需要在性能方面进行评估，还需要在不增加资源的情况下保持运营效率。

另一个紧迫的问题是MAS的鲁棒性和弹性。虽然这些系统提供了诸如容错性等优势——即一个智能体的故障不会破坏整个系统的性能——但在动态环境中设计支持自组织和适应性的架构仍然是一个挑战。确保智能体之间的安全通信也至关重要，特别是在金融系统和军事行动等敏感应用中，需要强大的加密和认证机制来保护信息交换的完整性和机密性。

未来分布式MAS的研究方向应侧重于通过跨学科方法解决这些开放性问题。需要不仅促进技术采用，还能管理组织变革中以人为本的框架。正如van Der Aalst等学者所强调的，理解文化和组织障碍对于在抵制此类创新的行业中集成AI驱动的变革至关重要。

此外，将多智能体强化学习（MARL）与结构化人类工作流相结合，提供了另一个值得探索的方向。当前的研究往往忽视了这种集成，留下了一个必须弥合的空白，以实现多智能体系统（MAS）在现实世界应用中的全部潜力。随着城市交通格局的演变，技术创新与公平、环保的城市生活之间的协同作用将变得越来越重要，这将使MAS成为塑造未来城市环境的关键参与者。

参考文献

1. J. Bajo, F. De la Prieta, J. M. Corchado, and S. Rodríguez, A low-level resource allocation in an agent-based cloud computing platform, Appl. Soft Comput., vol. 48, pp. 716728, Nov. 2016.
2. J. Fiosina and M. Fiosins, Density-based clustering in cloud-oriented collaborative multi-agent systems, inProc.Int.Conf.HybridArtif.Intell. Syst., 2013, pp. 639648.
3. J. O. Gutierrez-Garcia and K. M. Sim, Agent-based cloud bag-of-tasks execution, J. Syst. Softw., vol. 104, pp. 1731, Jun. 2015.
4. A. Singh, D. Juneja, and M. Malhotra, Autonomous agent based load balancing algorithm in cloud computing, Proc. Comput. Sci., vol. 45, pp. 832841, Jan. 2015.
5. K.Govinda and E. Sathiyamoorthy, Agent based security for cloud computing using obfuscation, Proc. Eng., vol. 38, pp. 125129, Jan. 2012.
6. R. Nikbazm and M. Ahmadi, Agent-based resource discovery in cloud computing using bloom lters, in Proc. 4th Int. eConf. Comput. Knowl. Eng. (ICCKE), Oct. 2014, pp. 352357.
7. F. Hou and X. Mao, Cross-clouds services autonomic management approach based on self-organizing multi-agent technology, Concurrency Comput., Pract. Exper., vol. 28, no. 11, pp. 32133237, 2016.
8. K.M.Sim, Agent-based cloud computing, IEEETrans.Serv. Comput., vol. 5, no. 4, pp. 564577, Oct. 2012.
9. Y. Jiang and J. C. Jiang, Understanding social networks from a multiagent perspective, IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 10, pp. 27432759, Oct. 2014.
10. M. Gatti et al., Large-scale multi-agent-based modeling and simulation of microblogging-based online social network, in Proc. Int. Workshop Multi-Agent Syst. Agent-Based Simulation, 2013, pp. 1733.
11. L. Ma and Y. Zhang, Hierarchical social network analysis using multiagent systems: Aschoolsystemcase, inProc.IEEEInt.Conf.Syst.,Man Cybern. (SMC), Oct. 2014, pp. 14121419.
12. V. Gorodetski and I. Kotenko, The multi-agent systems for computer network security assurance: Framework sand case studies, inProc.IEEE Int. Conf. Artif. Intell. Syst. (ICAIS), Sep. 2002, pp. 297302.
13. L.Mechtri, F. D. Tolba, and S. Ghanemi, Masid: Multi-agent system for intrusion detection in MANET, in Proc. 9th Int. Conf. Inf. Technol., New Generat. (ITNG), Apr. 2012, pp. 6570.
14. S. Sarika and V. Paul, AgentTab: An agent based approach to detect tabnabbing attack, Proc. Comput. Sci., vol. 46, pp. 574581, Jan. 2015.
15. S.Shamshirband, N.B.Anuar, M.L.M.Kiah, and A.Patel, Anappraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique, Eng. Appl. Artif. Intell., vol. 26, no. 9, pp. 21052127, 2013.
16. G. Di Caro and M. Dorigo, An adaptive multi-agent routing algorithm inspired by ants behavior, in Proc. 5th Annu. Austral. Conf. Parallel Real-Time Syst. (PART), 1998, pp. 261272.
17. R. Claes, T. Holvoet, and D. Weyns, A decentralized approach for anticipatory vehicle routing using delegate multiagent systems, IEEE Trans. Intell. Transp. Syst., vol. 12, no. 2, pp. 364373, Feb. 2011.
18. M. Bendjima and M. Feham, Multi-agent system for a reliable routing in WSN, inProc. Sci. Inf. Conf. (SAI), Jul. 2015, pp. 14121419.
19. S. Manvi and M. Kakkasageri, Multicast routing in mobile ad hoc networks by using a multiagent system, Inf. Sci., vol. 178, no. 6, pp. 16111628, 2008.
20. G.Dudek,M.R.Jenkin,E.Milios,andD.Wilkes, Ataxonomyformulti agent robotics, Auto. Robot., vol. 3, no. 4, pp. 375397, 1996.
21. C.G.Cena,P. F. Cardenas, R. S. Pazmino, L. Puglisi, and R. A. Santonja, A cooperative multi-agent robotics system: Design and modelling, Expert Syst. Appl., vol. 40, no. 12, pp. 47374748, 2013.
22. Y. Duan, B. X.Cui, and X.H.Xu, Amulti-agentreinforcement learning approach to robot soccer, Artif. Intell. Rev., vol. 38, no. 3, pp. 193211, 2012.
23. J. Ota, Multi-agent robot systems as distributed autonomous systems, Adv. Eng. Informat., vol. 20, no. 1, pp. 5970, 2006.
24. P. Inigo-Blasco, F. Diaz-del-Rio, M. C. Romero-Ternero, D. Cagigas-Muñiz, and S. Vicente-Diaz, Robotics software frameworks for multi-agent robotic systems development, Robot. Auto. Syst., vol. 60, no. 6, pp. 803 821, 2012.
25. A. Soriano, E. J. Bernabeu, A. Valera, and M. Vallès, Multi-agent systems platform for mobile robots collision avoidance, in Proc. Int. Conf. Practical Appl. Agents Multi-Agent Syst., 2013, pp. 320323.
26. D. Helbing, Agent-based modeling, in Social self-Organization. Berlin, Germany: Springer, 2012, pp. 2570.
27. R.Domínguez, S.Cannella,andJ.M.Framinan, SCOPE:Amulti-agent system tool for supply chain network analysis, in Proc. IEEE Int. Conf. Comput. (EUROCON), Sep. 2015, pp. 15.
28. O. Wangapisit, E. Taniguchi, J. S. Teo, and A. G. Qureshi, Multi-agent systems modelling for evaluating joint delivery systems, Proc.-Social Behavioral Sci., vol. 125, pp. 472483, Mar. 2014.
29. S. D. J. McArthur et al., Multi-agent systems for power engineering applications Part I: Concepts, approaches, and technical challenges, IEEE Trans. Power Syst., vol. 22, no. 4, pp. 17431752, Nov. 2007.
30. P. Ringler, D. Keles, and W. Fichtner, Agent-based modelling and simulation of smartelectricity grids andmarkets Aliteraturereview, Renew. Sustain. Energy Rev., vol. 57, pp. 205215, 2016.
31. M. Khayyat and A. Awasthi, An intelligent multi-agent based model for collaborative logistics systems, Transp. Res. Procedia, vol. 12, pp. 325338, Jan. 2016.
32. K. Hager, J. Rauh, and W. Rid, Agent-based modeling of traf c behavior in growing metropolitan areas, Transp. Res. Procedia, vol. 10, pp. 306315, Jan. 2015.
33. O. van Pruissen, A. van der Togt, and E. Werkman, Energy ef ciency comparison of a centralized and a multi-agent market based heating system in a eld test, Energy Procedia, vol. 62, pp. 170179, Jan. 2014.
34. J. Cai, D. Kim, R. Jaramillo, J. E. Braun, and J. Hu, A general multi agent control approach for building energy system optimization, Energy Buildings, vol. 127, pp. 337351, Sep. 2016. C. P. Nguyen and A. J. Flueck, Agent based restoration with distributed
35. energy storage support in smart grids, IEEE Trans. Smart Grid, vol. 3, no. 2, pp. 10291038, Jun. 2012.
36. P. Vytelingum, T. D. Voice, S. D. Ramchurn, A. Rogers, and N. R. Jennings, Agent-based micro-storage management for the smart grid, in Proc. 9th Int. Conf. Auto. Agents Multiagent Syst., 2010, pp. 3946.
37. G.H.Merabetetal., Applications of multi-agent systems in smart grids: Asurvey, in Proc. Int. Conf. Multimedia Comput. Syst. (ICMCS), 2014, pp. 10881094.
38. B. J. Austin, V. Heine, and L. Sham, General theory of pseudopotentials, Phys. Rev. J. Arch., vol. 127, no. 1, p. 276, 1962.
39. R. Kibble, Speech acts, commitment and multi-agent communication, Comput. Math. Org. Theory, vol. 12, nos. 23, pp. 127145, 2006.
40. M. Colombetti and M. Verdicchio, An analysis of agent speech acts as institutional actions, in Proc. 1st Int. Joint Conf. Auto. Agents Multiagent Syst., 2002, pp. 1157 1164.
41. M. Wooldridge, An Introduction to Multiagent Systems. New York, NY, USA: Wiley, 2009.
42. P. Balaji and D. Srinivasan, An introduction to multi-agent systems, in Innovations in Multi-Agent Systems and Applications. Berlin, Germany: Springer, 2010, pp. 127.
43. Venkatesan, D. A Novel Agent-Based Enterprise Level System Development Technology. Ph.D. Thesis, Anna University, Tamil Nadu, India, 2018.
44. Bellifemine, F.; Bergenti, F.; Caire, G.; Poggi, A. JADE—A java agent development framework. In Multi-Agent Programming; Springer: Boston, MA, USA, 2005; pp. 125–147.
45. DeLoach, S.A.; Garcia-Ojeda, J.C. O-MaSE: A customisable approach to designing and building complex, adaptive multi-agent systems. Int. J. Agent-Oriented Softw. Eng. 2010, 4, 244–280.
46. Kantamneni, A.; Brown, L.E.; Parker, G.; Weaver, W.W. Survey of multi-agent systems for microgrid control. Eng. Appl. Artif. Intell. 2015, 45, 192–203.
47. Cardoso, R.C.; Ferrando, A. A Review of Agent-Based Programming for Multi-Agent Systems. Computers 2021, 10, 16.
48. Kravari, K.; Bassiliades, N. A survey of agent platforms. J. Artif. Soc. Soc. Simul. 2015, 18, 11.
49. Bordini, R.H.; El Fallah Seghrouchni, A.; Hindriks, K.; Logan, B.; Ricci, A. Agent programming in the cognitive era. Auton. Agents Multi-Agent Syst. 2020, 34, 37.
50. Costantini, S. ACE: A flexible environment for complex event processing in logical agents. In International Workshop on Engineering Multi-Agent Systems; Springer: Cham, Switzerland, 2015; pp. 70–91.
51. Araujo, P.; Rodríguez, S.; Hilaire, V. A metamodeling approach for the identification of organizational smells in multi-agent systems: Application to ASPECS. Artif. Intell. Rev. 2018, 49, 183–210.
52. Boissier, O.; Bordini, R.H.; Hübner, J.F.; Ricci, A. Dimensions in programming multi-agent systems. Knowl. Eng. Rev. 2019, 34, e2.
53. Rahimi, H.; Trentin, I.F.; Ramparany, F.; Boissier, O. SMASH: A Semantic-enabled Multi-agent Approach for Self-adaptation of Human-centered IoT. arXiv 2021, arXiv:2105.14915.
54. Baek, Y.M.; Song, J.; Shin, Y.J.; Park, S.; Bae, D.H. A meta-model for representing system-of-systems ontologies. In Proceedings of the 2018 IEEE/ACM 6th International Workshop on Software Engineering for Systems-of-Systems (SESoS), Gothenburg, Sweden, 29 May2018; pp. 1–7.
55. Pigazzini, I.; Briola, D.; Fontana, F.A. Architectural Technical Debt of Multiagent Systems Development Platforms. In Proceedings of the WOA2021: Workshop “From Objects to Agents”, Bologna, Italy, 1–3 September 2021.
56. Jazayeri, A.; Bass, E.J. Agent-Oriented Methodologies Evaluation Frameworks: A Review. Int. J. Softw. Eng. Knowl. Eng. 2020, 30, 1337–1370.
57. Logan, B. An agent programming manifesto. Int. J. Agent-Oriented Softw. Eng. 2018, 6, 187–210.
58. Zheng, Y.; Wang, L. Consensus of heterogeneous multi-agent systems without velocity measurements. Int. J. Control 2012, 85, 906–914.
59. Rizk,Y.; Awad, M.; Tunstel, E.W. Decision making in multiagent systems: A survey. IEEE Trans. Cogn. Dev. Syst. 2018, 10, 514–529.
60. Rossi, F.; Bandyopadhyay, S.; Wolf, M.T.; Pavone, M. Multi-Agent Algorithms for Collective Behavior: A structural and application-focused atlas. arXiv 2021, arXiv:2103.11067.
61. Chen, F.; Ren, W. Multi-Agent Control: A Graph-Theoretic Perspective. J. Syst. Sci. Complex. 2021, 34, 1973–2002.
62. Zelazo, D.; Rahmani, A.; Mesbahi, M. Agreement via the edge laplacian. In Proceedings of the 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007; pp. 2309–2314.
63. You, K.; Xie, L. Network topology and communication data rate for consensusability of discrete-time multi-agent systems. IEEE Trans. Autom. Control 2011, 56, 2262–2275.
64. Shi, C.X.; Yang, G.H. Robust consensus control for a class of multi-agent systems via distributed PID algorithm and weighted edge dynamics. Appl. Math. Comput. 2018, 316, 73–88.
65. Wang, Q.; Wang, Y. Cluster synchronization of a class of multi-agent systems with a bipartite graph topology. Sci. China Inf. Sci. 2014, 57, 1–11.
66. Shoham, Y.; Leyton-Brown, K. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations; Cambridge University Press: Cambridge, UK, 2008.
67. Zhu, M.; Martínez, S. Distributed coverage games for energy-aware mobile sensor networks. SIAM J. Control Optim. 2013, 51, 1–27.
68. Sengupta, S.; Kambhampati, S. Multi-agent reinforcement learning in bayesian stackelberg markov games for adaptive moving target defense. arXiv 2020, arXiv:2007.10457.
69. Sun, C.; Wang, X.; Liu, J. Evolutionary game theoretic approach for optimal resource allocation in multi-agent systems. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 5588–5592.
70. Yan, B.; Shi, P.; Lim, C.C.; Shi, Z. Optimal robust formation control for heterogeneous multi-agent systems based on reinforcement learning. Int. J. Robust Nonlinear Control 2021, 32, 2683–2704.
71. Wang,D.;Wang,Z.; Chen,M.; Wang,W. Distributed optimization for multi-agent systems with constraints set and communication time-delay over a directed graph. Inf. Sci. 2018, 438, 1–14.
72. Wai, H.T.; Yang, Z.; Wang, Z.; Hong, M. Multi-agent reinforcement learning via double averaging primal-dual optimization. arXiv 2018, arXiv:1806.00877.
73. Jian, L.; Zhao, Y.; Hu, J.; Li, P. Distributed inexact consensus-based ADMM method for multi-agent unconstrained optimization problem. IEEE Access 2019, 7, 79311–79319.
74. Kapoor, S. Multi-agent reinforcement learning: A report on challenges and approaches.
75. Busoniu, L.; Babuska, R.; De Schutter, B. A comprehensive survey of multiagent reinforcement learning. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) 2008, 38, 156–172.
76. Zhang, K.; Yang, Z.; Ba¸ sar, T. Decentralized multi-agent reinforcement learning with networked agents: Recent advances. arXiv 2019, arXiv:1912.03821.
77. Papoudakis, G.; Christianos, F.; Rahman, A.; Albrecht, S.V. Dealing with non-stationarity in multi-agent deep reinforcement learning. arXiv 2019, arXiv:1906.04737.
78. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. arXiv 2017, arXiv:1706.02275.
79. Rashid, T.; Samvelyan, M.; Schroeder, C.; Farquhar, G.; Foerster, J.; Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 4295–4304
80. Zhang, K.; Yang, Z.; Ba¸ sar, T. Multi-agent reinforcement learning: A selective overview of theories and algorithms. In Handbook of Reinforcement Learning and Control; Springer: Cham, Switzerland, 2021; pp. 321–384.
81. Du, W.; Ding, S. A survey on multi-agent deep reinforcement learning: From the perspective of challenges and applications. Artif. Intell. Rev. 2021, 54, 3215–3238.
82. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Proceedings of the International Conference on Machine Learning, PMLR, New York, NY, USA, 20–22 June 2016; pp. 1928–1937.
83. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. arXiv 2015, arXiv:1509.02971.
84. Panait, L.; Luke, S. Cooperative multi-agent learning: The state of the art. Auton. Agents Multi-Agent Syst. 2005, 11, 387–434.
85. Foerster, J.; Farquhar, G.; Afouras, T.; Nardelli, N.; Whiteson, S. Counterfactual multi-agent policy gradients. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
86. Da Silva, F.L.; Costa, A.H.R. A survey on transfer learning for multiagent reinforcement learning systems. J. Artif. Intell. Res. 2019, 64, 645–703.
87. Dusparic, I.; Cahill, V. Autonomic multi-policy optimization in pervasive systems: Overview and evaluation. ACM Trans. Auton. Adapt. Syst. (TAAS) 2012, 7, 1–25.
88. Sharma, P.K.; Fernandez, R.; Zaroukian, E.; Dorothy, M.; Basak, A.; Asher, D.E. Survey of recent multi-agent reinforcement learning algorithms utilizing centralized training. In Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III; International Society for Optics and Photonics: Orlando, FL, USA, 2021; Volume 11746, p. 117462K.
89. Foerster, J.; Farquhar, G.; Afouras, T.; Nardelli, N.; Whiteson, S. Counterfactual multi-agent policy gradients. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
90. De Lemos, R.; Giese, H.; Müller, H.A.; Shaw, M.; Andersson, J.; Litoiu, M.; Schmerl, B.; Tamura, G.; Villegas, N.M.; Vogel, T.; et al. Software engineering for self-adaptive systems: A second research roadmap. In Software Engineering for Self-Adaptive Systems II; Springer: Heidelberg, Germany, 2013; pp. 1–32.
91. Panait, L.; Luke, S. Cooperative multi-agent learning: The state of the art. Auton. Agents Multi-Agent Syst. 2005, 11, 387–434.
92. Rashid, T.; Samvelyan, M.; Schroeder, C.; Farquhar, G.; Foerster, J.; Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 4295–4304.
93. Chen, G. A NewFrameworkfor Multi-Agent Reinforcement Learning–Centralized Training and Exploration with Decentralized Execution via Policy Distillation. arXiv 2019, arXiv:1910.09152.
94. Pesce, E.; Montana, G. Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication. Mach. Learn. 2020, 109, 1727–1747.
95. Czarnowski, I.; J˛edrzejowicz, P. An agent-based framework for distributed learning. Eng. Appl. Artif. Intell. 2011, 24, 93–102.
96. D’Angelo, M. Engineering Decentralized Learning in Self-Adaptive Systems. Ph.D. Thesis, Linnaeus University Press, Vaxjo, Sweden, 2021.