**Chapter1 概论**

**智能体**

Actor——Agent的前身，具有一定的计算智能；具有交互性（消息发送及反馈）。

Agent-Society——社会中的个体通过交互得到问题的求解，个体就是“Agent”。

Agent——处在某个环境中，能够灵活、自主地行动以实现其目标的个体。

**Agent概念**

基本属性：

1. 自治性：对行为及自我状态进行控制
2. 反应性：可感知环境并根据环境变化作出反应
3. 目标主动性：基于既定目标主动发起行为
4. 社会交互性：通过群体合作与协调实现设计目标

特征：

1. 智能性：根据推理、学习等技术分析解释各种信息和知识的能力
2. 代理能力：Agent能感知外界发生的消息，并根据自己所具有的知识、动作作出反应

体系架构：

1. 基于推理的架构 2. 基于反应的架构（包容体系架构、马尔科夫决策过程）

3. 分层架构（水平分层架构、垂直分层架构） 4. 信念-愿望-意图架构（BDI）

Agent可以根据体系结构、属性或功能的不同来进行分类。

**多智能体系统概念**

研究目的：单个的Agent对问题的解决能力有限，很难完成动态分布、网络和异构情况下的大型、复杂问题，将单个Agent放入多智能体系统中，可以解决大型、复杂问题。

基本属性：

1. 感知：具有感知周围环境的能力
2. 相互通讯：具有与其他智能体交互的能力
3. 计算：信息处理、计算能力
4. 控制：需求响应、决策能力

特性讨论

|  |  |  |
| --- | --- | --- |
| 分布式 | 开放性 | 动态性 |
| Agent感知：局部性 | Agent性质：异质 | Agent状态 |
| Agent交互：局部性 | Agent目标：异质 | Agent交互结构 |
| Agent计算：异步性 | Agent可加入或退出系统 | Agent交互策略 |

交互结构举例：1. 纯分布式结构 2. 联盟结构 3. 等级层次结构

交互结构特征：1. 小世界特性 2. 无标度特性 3. 网络层次特性

系统的同质与异质——同一个系统可以从不同评判角度被认为是同质的，也可以被认为是异质的。

多智能体学习的五大目标：1. 计算性 2. 描述性 3. 规范性 4. 规定性、合作性 5. 规定性、非合作性

**Chapter2 多智能体系统模型及框架**

**智能体与多智能体系统的基本模型与形式化定义**

基本模型与操作流程：



1. Agent从环境获取信息输入（环境感知、信息交互）
2. Agent经过决策做出动作作用于环境

动作输出具体依赖于：

1. Agent 及其环境的先验知识
2. 与环境的交互历史（当前环境的观察值、先前经验和观察值或从学习中获得到的其他数据）
3. 试图达到的目标或整个世界状态的偏好
4. 它能够执行的最原始动作，即能力

Agent的基本形式化定义元素（Agent：a=<R, Loc, Obj, S>）：

资源——R={R1, R2,… Rj};

局部信息——Loc={Loc1, Loc2,… Lock};

目标——Obj={obj1, obj2,… objl};

行为策略——S={s1, s2,… sm};

多Agent系统的基本形式化定义元素（MAS: mas= <A, E, OBJ, Rule>）：

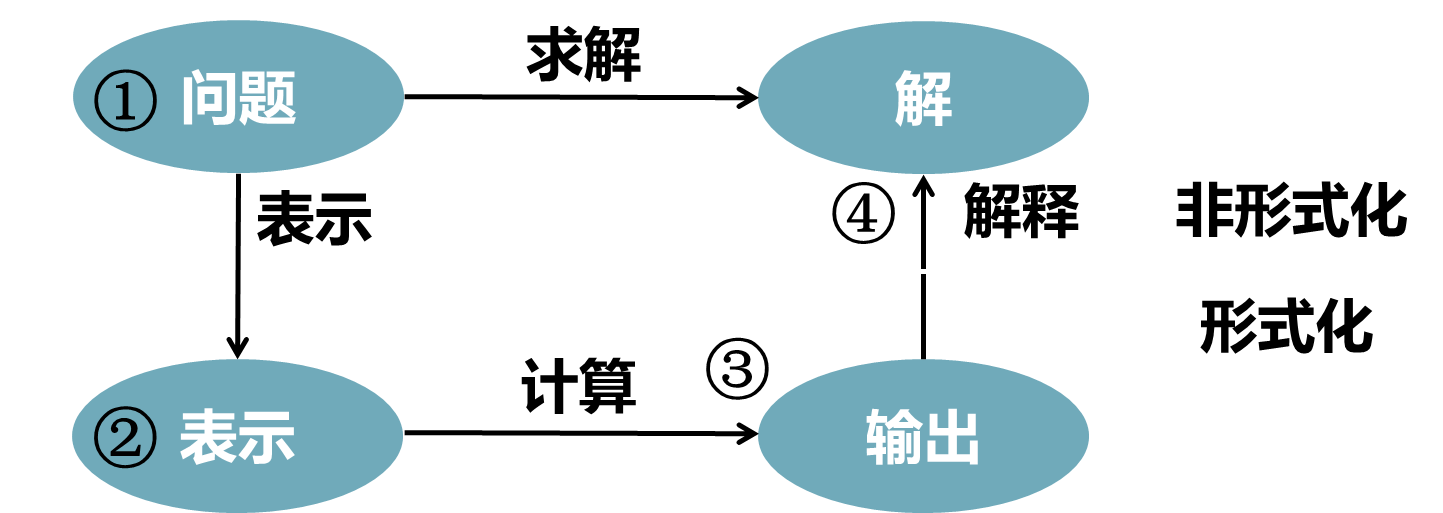
Agent集合——A={a1, a2,… aj};

Agent交互连接——E={(a1, a2), …(ai, aj)};

目标——OBJ={O1, O2,… Ok};

系统规则——Rule={rul1, rul2,… rull};

**多智能体系统问题求解基本框架**



知识——可以用来解决本领域问题的领域信息。

表示方案——Agent 中使用的知识形式

知识库——Agent中存储的所有知识的表示

流程：

1. 确定问题：确定问题解的构成，对问题的描述进行提炼
2. 表示——对问题、环境、系统进行建模，便于计算机解决：

* 符号和符号系统

1. 计算机和人类思维都属于物理符号系统，符号是一种可以被操作的有意义的图案。符号系统对符号进行创造、复制、修改和撤销；符号是符号系统中被操作的基本单元。
2. Agent通过操作符号来产生动作。

* 抽象和抽象层次

1. 所有的模型都是抽象出来的，Agent可以用物理符号系统对世界进行抽象建模，世界模型是世界中真实存在的详情和世界的动态行为的表达。
2. 抽象层次是指按抽象程度排列的一个偏序，低层抽象比高层抽象拥有更多的细节。

* 两个共有的层次

1. 知识层：直接的表达，将Agent所知道的、所相信的以及它的目标抽象出来。知识层考虑的是Agent知道什么，而不进行推理。
2. 符号层：操作符号进行推理，产生答案，该层进行对Agent内部关于外部世界推理的描述。
3. 推理与行为（计算并输出）：符号的操作产生行为推理

* 三种计算

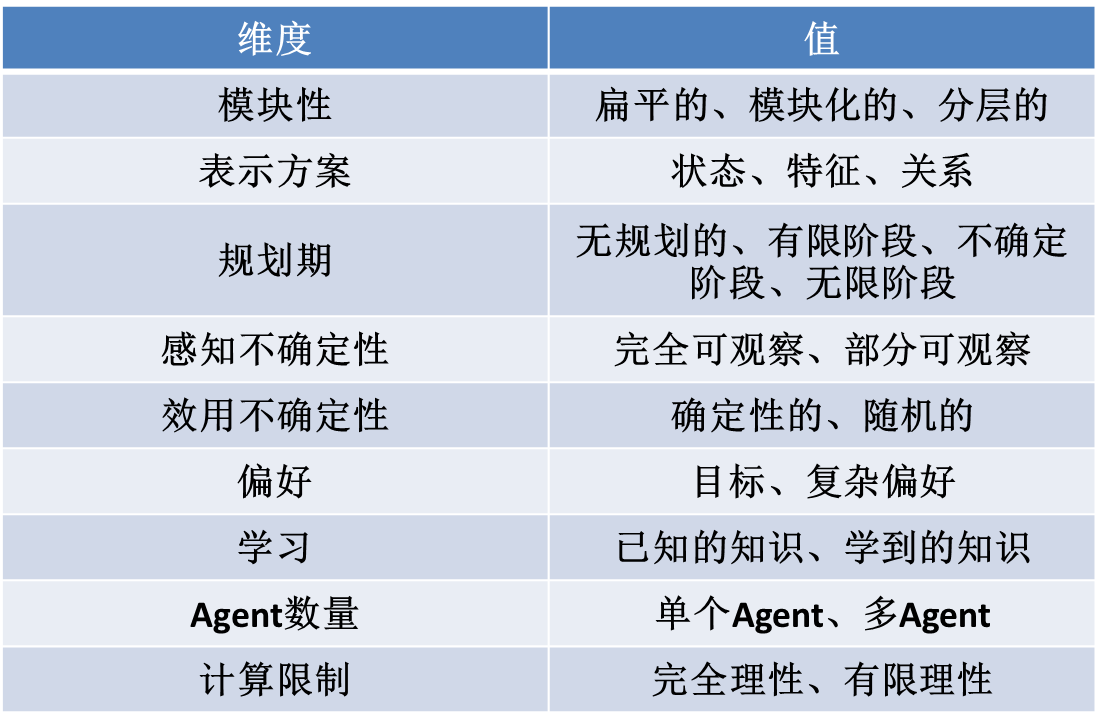
1. 设计时刻推理：设计Agent时进行的推理，由Agent 的设计完成。
2. 离线计算：Agent离线获取背景知识和数据，并将其汇编成可用的形式，即知识库。背景知识可以在设计时或离线状态时给出。离线计算指在观察世界、需要行动之前，Agent基于知识库进行计算。
3. 在线计算：在线状态获取的信息称为观察，在线计算是在观察环境及在环境中进行某些行为时由Agent完成的计算。Agent必须使用知识库和观察来决定要做什么。

* Agent构造策略

1. 简化环境并为这些环境建立复杂的推理系统
2. 在复杂环境中建立简单的agent
3. 定义解：
4. 最优解：问题的最优解是指在所有的可能解中最符合求解质量度量的最好的解。这个度量用序数来描述。但是在一些情况下，例如综合多种标准或不确定推理时，我们需要一种要考虑相对大小的基数度量。
5. 可满足解：满足条件的解。
6. 近似最优解：近似最优解是指其质量度量接近理论上存在的最优解
7. 可能解：一个有希望的解。

**智能体设计的复杂性维度**

定义： 复杂性维度定义了人工智能的设计空间，空间里的不同点可以通过改变维度值来获得。Agent在环境中行为的复杂性各不相同。在Agent设计过程中，存在着多个复杂性维度。这些维度可以分开考虑，但构建智能Agent时必须组合起来。



模块性维度：

|  |  |  |  |
| --- | --- | --- | --- |
| 种类 | 扁平 | 模块化 | 分层 |
| 定义 | 系统中没有组织结构 | 系统可以被分解成独立的可理解的交互模块 | 系统是模块化的，模块本身分解成了交互模块，迭代分层至原子模块 |
| 推理类型 | 单层抽象推理 | 单层抽象推理 | 多层抽象推理 |
| 任务完成方法 | 将所有需要完成的工作按照某一顺序完成 | 将任务分解为若干子任务，再考虑子任务中的细节 | 用分层的方式解决这些子任务，直至将问题分解成一些简单的问题 |

表示方案维度（从左到右表示更简洁）：

|  |  |  |  |
| --- | --- | --- | --- |
| 种类 | 状态 | 特征 | 个体与关系 |
| 包含内容 | 内部状态、环境状态 | 状态的特征值 | 属性、关系 |
| 推理途径 | 通过一系列独立的确定状态进行推理 | 用状态特征或状态真假的命题进行推理 | 用个体属性及它们之间的关系进行相关性描述 |
| 备注 |  | 用状态的紧凑表示法更容易理解 | 处理无限个体集合时，无法用状态或特征描述，只能在关系层面上进行推理 |

规划期维度——Agent规划的向前时间的程度：

规划期——Agent认为它的动作结果所能影响的向前程度。

阶段——Agent做规划时所考虑的时间点。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 种类 | 非规划Agent | 有限期规划 | 不确定期规划 | 无限期规划 |
| 定义 | 做出动作时，不考虑未来影响 | 遵循有限的时间步 | 能够向前探索但不确定步数 | 一直向前进行规划 |

不确定性维度：

|  |  |  |
| --- | --- | --- |
| 种类 | 感知不确定性 | 效用不确定性 |
| 定义 | 能否感知环境中的状态 | 做出动作的效果不确定 |
| 分类 | 1. 部分可观察：不能直接观察状态或相同的观察结果导致不同状态 2. 完全可观察：直接观察世界状态 | 1. 确定性的：动作所导致的状态由动作及之前的状态决定。 2. 随机的：对于结果状态，只能给出概率分布。 |

偏好维度——做出某一动作优于另一动作选择的唯一原因是其偏好动作会导致更理想的结果：

|  |  |  |
| --- | --- | --- |
| 种类 | 目标偏好 | 复杂偏好 |
| 定义 | 在某一最终状态下需完成的目标，或是在所有已访问过的状态中必须被保持的目标 | 在不同时期权衡各种期望的结果：序数偏好表示只注重偏好的排序，基数偏好涉及有关值的大小 |

Agent数量维度：

|  |  |  |
| --- | --- | --- |
| 种类 | 单Agent | 多Agent |
| 假设 | 其他Agent为环境的一部分 | 考虑其他Agent的推理 |
| 使用情况 | 没有其他Agent，或者其他Agent的动作不会对自身造成影响 | 其他Agent的目标或偏好部分依赖于此Agent的行为，或Agent必须与其他Agent通信 |
| 难点 |  | Agent同时进行动作，或环境部分可观察 |

学习维度——先验知识或其他资源帮助Agent进行决策：

学习——找到与数据相符的最好模型

种类：已有的知识、学到的知识（从数据或经验获取）

计算限制维度——受限于计算资源和计算时间的限制，Agent给出的计算结果可能不是最优的：

|  |  |  |
| --- | --- | --- |
| 种类 | 完全理性 | 有限理性 |
| 定义 | Agent可以推理出最佳方案，不需要考虑有限的计算资源 | 在给定的计算资源限制上，Agent决定所能找到的最佳方案 |

任意时间算法——解的质量随时间的推移而提高。Agent可以在任意时间内产生当前最佳解，但如果给定更多时间，可能会产生更好的解决方案。

**Chapter3 多智能体系统交互结构**

|  |  |  |
| --- | --- | --- |
|  | 现实现象仿真 | 实际问题求解 |
| 交互结构的影响 | 1. **影响个体行为决策过程** 2. **影响系统行为显现** | 1. **影响系统效率** 2. **影响系统可靠性** |

**交互结构的表示方法**

图表示（G(A, E)——A表示顶点，E表示边）

方向性（支持混合连接）：

1. 无向连接(a1, a2)
2. 有向连接<a1, a2>

数据结构对比（n为智能体数量，m为连接数量）：

|  |  |  |
| --- | --- | --- |
| 数据结构对比 | 邻接表 | 邻接矩阵 |
| 表示方法 | 由数组与链表表示，数组表示节点，链表表示边 | 由二维数组表示 |
| 空间复杂度 | O(m+n) | O(n2) |
| 适用情况 | 稀疏图（m<<n2） | 稠密图 |
| 访问特性 | 空间复杂度小，多用于大规模多智能体系统交互结构 | 可以直接访问特定节点的直接邻居/修改交互结构，多用于小规模结构 |

**交互结构的度量指标**

|  |  |  |  |
| --- | --- | --- | --- |
| 度量 | 度（k） | 平均路径长度（L） | 聚类系数（C） |
| 定义 | 节点连接的边数 | 任意两节点间最短路径长度d的平均值 | 表示图中某个节点聚集程度的系数（节点若只有一个邻居，则该点局部聚类系数为0） |
| 公式 |  |  | 局部聚类系数：  平均聚类系数：  全局聚类系数： |
| 主要作用 | 衡量智能体的交互地位 | 评估智能体间交互的平均耗费 | 反映智能体局部的行为强化作用 |
| 符号说明 |  | ：总节点集 | ：节点的邻居之间存在的实际边数  ：节点的邻居数 |

**交互结构的基本模型**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 网络结构 | 全局耦合网络（GC） | 最近邻耦合网络（NC） | 星型网络（Star） | 随机网络（ER） |
| 交互结构分类 | 规则交互 | | | 随机交互 |
| 定义 | 任意两节点间均有边相连 | 任意节点均与其左右个邻居节点相连(K为偶数) | 存在一个中心节点，其余节点均只与该中心节点连接 | 任意两点间以一定概率p建立交互连接 |
| 度 |  |  | 1（非中心） |  |
| 平均路径长度 | 1 |  | 2 |  |
| 聚类系数 | 1 |  | 0 |  |
| 性质 | 1）聚类性  2）小世界性质  3）便于全局协调 | 1）聚类性  2）难以全局协调（当节点数量大时，平均路径长度很大） | 1）小世界性质  2）便于全局协调  3）鲁棒性差 | 1）小世界性质  2）p很小时，没有高聚类特性 |

|  |  |  |  |
| --- | --- | --- | --- |
| 网络结构 | WS小世界模型 | NW小世界模型 | 无标度网络（BA） |
| 交互结构分类 | 小世界交互 | | 无标度交互 |
| 构建方法 | 建立规则图：建立含有N个节点的最近邻耦合网络  随机化重连：以概率p随机地重连网络中的每条边 | 建立规则图：建立含有N个节点的最近邻耦合网络  随机化加边：以概率p随机地在任意一对节点间增加边 | 网络增长：每次增加一个节点，连接到已有节点上  优先连接：新节点与连接概率为 |
| 度 | *K* |  |  |
| 平均路径长度 |  | |  |
| 聚类系数 |  |  |  |
| 性质 | 1）聚类性  2）小世界性质  3）便于全局协调 | 1）聚类性  2）难以全局协调（当节点数量大时，平均路径长度很大） | 1）网络增长特性  2）优先连接特性：新节点更倾向于与度大的节点建立连接  3）小世界特性 |
| 演化特性 | 六度分离理论 | | 1. 富者更富 2. 赢者通吃 |
| 鲁棒性 |  |  | 1）对于随机故障策略具有良好鲁棒性  2）对于蓄意攻击策略具有高度脆弱性 |

**交互结构的应用性指标**

节点度量：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 度量指标 | 度中心性（Degree centrality） | 介数中心性（Betweenness centrality） | 接近中心性（Closeness centrality） | k-壳&k-核 | 资源富集性 |
| 定义 |  | 经过该节点的最短路径数目 | 节点与全局的接近程度 |  | 个体在一定的局部范围内能够获取到的资源 |
| 定量计算 |  | ：从节点到的最短路径数量  ：从节点到的最短路径中经过节点的数量 |  | 迭代去掉度为的节点及其连接，直至没有度为1的节点。被去掉的节点及连接称为 -壳  -壳指标 的 -壳并集称为 -核 | 上述公式由以下几个方面决定（考虑（1~n阶）邻居）:  邻居的数量  邻居的资源状态  邻居所在位置 |
| 评估意义 | 一般而言，某节点度越大，则该节点在网络局部越重要 | 发现交互中“关键”节点 | 发现对于交互具有最优“损耗”的节点 | 大规模网络粗粒化分类 |  |

交互连接指标：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 度量指标 | 连接强度 | 嵌入性 | 介数中心性 | 正负性 |
| 定义 |  | 两个节点共同邻居数量（邻居重叠度的分子） | 该交互连接上所有节点对间最短路径流量之和 | 朋友间的连接为（+），敌人间的连接为（-） |
| 定量计算 | 邻居重叠度： |  | 网络中所有最短路径中经过该边的路径的数目占最短路径总数的比例 | 平衡关系：  +++/+--  平衡结构：  交互结构中所有三角关系都平衡 |
| 评估意义 | 如果两个节点的共同邻居在所有邻居中比例越高，则认为这两个点间的连接强度越大 | 连接大量低嵌入性边的节点可以跨越系统中的结构洞，常认为系统中存在的结构洞可以为处于该位置的个体带来信息和资源上的优势 | 发现交互中“最关键”的边 |  |

**Chapter4 多智能体协作**

**研究背景**

主要研究问题：

1. 如何将问题分解为可分配给单个智能体的较小任务？
2. 如何将这些子任务分配给各个智能体？
3. 使用什么样的技术来协调智能体的活动，实现各智能体间的动态协作？
4. 如何有效地将子问题的结果综合成问题的解？

多智能体协作研究过程：问题分解-子问题求解-进行解综合

优化目标：使整个多智能体系统性能达到最佳并且可以迅速的完成任务

**多智能体协作方式介绍**

|  |  |  |
| --- | --- | --- |
| 协作方式 | 集中式协作方式 | 分布式协作方式 |
| 结构定义 | 1. 存在一个智能体做决策并集中控制整个系统 2. 是一种自上而下的层次控制结构 | 1. 不存在集中控制单元，与其它智能体通讯进行协调 2. 各智能体具有高度的自治能力，自行处理信息、规划与决策、执行指令 |
| 主要方法 | 1. 集中式协商（投票） 2. 运筹学方法 3. 集中式学习 | 1. 合同网模型 2. 分布式黑板模型 3. 分布式学习 4. 博弈论 |
| 优点 | 1. 协调性较好 2. 可得到全局最优解 | 1. 容错能力和扩展能力较好 2. 不需要集中控制单元 |
| 缺点 | 1. 实时性、动态性差 2. 对环境变化响应能力差 | 1. 对智能体间通讯要求较高 2. 多边协作效率较低，难以保证实现全局目标 |

**分布式协作方式介绍**

黑板模型：

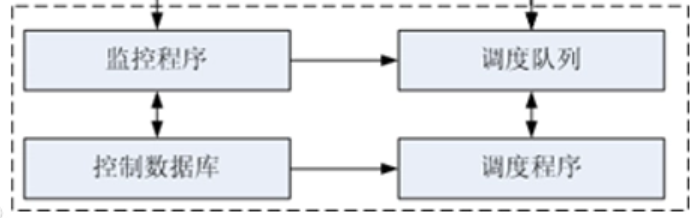
基本思想：解决分布在不同物理环境下多个实体协作完成任务的并行和分布计算模型，能实现异构知识源的集成。

分类与特性：

1. 中心式黑板：简单易行、易产生性能瓶颈
2. 分布式黑板：可靠性高、灵活性强

组成主体：

1. 知识源：知识源是描述某个独立领域问题的知识及其知识处理方法的知识库。在系统中具有多个知识源，每个知识源可用来完成某些特定的解题功能。知识源具有“条件-动作”的形式。条件描述了知识源应用求解的前提，动作描述了知识源的行为。当条件满足时，知识源被触发，其动作部分增加或修改黑板上的内容。
2. 黑板：黑板是用来存储数据、传递信息和处理方法的动态数据库，是系统中的全局工作区。黑板主要存放知识源所需要的信息和求解过程中的解状态数据，如初始数据、部分解、替换解、最终解等。
3. 监控机制：监控机制依据某种控制策略，根据黑板上的问题求解状态和各知识源的求解技能动态地选择和激活合适的知识源，使知识源能实时地响应黑板的变化。



模型特点：

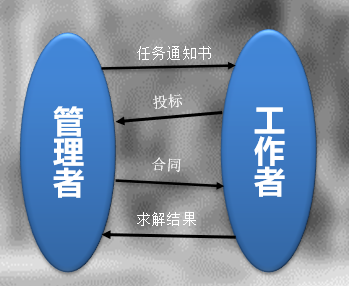
1. 各主体（即知识源）之间相互独立，主体之间不存在相互作用
2. 能够灵活表示信息
3. 使用共同的交互语言
4. 具有独立的监控机制
5. 适合于在多重抽象级上描述与处理问题
6. 是一种机遇问题求解机制，适合于事先无法确定问题求解次序的复杂问题
7. 提供了集成现有软件的一种方法，适用于软件的开发

合同网模型：

基本思想：源自人们在商务过程中用于管理商品和服务的合同机制，是一种分布式工作分配方式。

组成主体及其职责：

1. 管理者：对每一待求解任务建立任务通知书（Task-Announcement），将任务通知书发送给有关的工作者主体；接收并评估来自工作者的投标；从投标中选择最合适的工作者，与之建立合同（Contract）；监督任务的完成，并综合结果。
2. 工作者：接收相关任务通知书；评价自己的资格；对感兴趣的子任务返回任务投标；如果投标被接受，按合同执行分配给自己的任务；向管理者报告求解结果。



任务通知书：

* 作用：在合同网协作方法中，不需要预先定义主体的角色：任何主体通过发布任务通知书而成为管理者；任何主体通过应答任务通知书而成为工作者。这一灵活性使任务能够被层次地分解分配，形成一个动态确定的树结构
* 构成：

1. 收件人（Addressee）：任务通知书可发送给一个或多个可能求解任务的主体。
2. 资格说明（Eligibility Specification）：规定投标主体应满足的条件。
3. 任务简述（Task Abstraction）：对任务的简要描述，是工作者决策是否投标的依据。
4. 投标说明（Bid Specification）：向各主体说明投标必须提供的信息。
5. 终止时间（Expiration time）：接收投标的截止时间。

模型特点：

1. 灵活性较好：合同网中任务的产生、任务的分配、管理者、合同者的产生均是动态的。
2. 智能体间订立合同需要预先订立协议，因而对于任务不明确的大规模、动态环境下的任务分配问题，管理者并不了解实时情况下有哪些智能体可以完成任务。
3. 不保证最优解：当更有能力的主体在合同建立之前处于忙状态时，管理者只能选择一个有限能力的智能体。
4. 合同网模型忽略了任务之间可能存在的各种关联，因此更适用于任务能够较容易地独立分解，即分解后各个的子任务之间不存在相互作用的问题。
5. 通信效率问题：在协商过程中合同网的各智能体间以广播的方式互相通信。

联盟形成：

基本思想：在多智能体系统中，自治的智能体会结成联盟解决单个智能体无法解决的问题，提高问题求解的收益。

形成过程：

* 产生联盟结构（集合划分问题）——找到能够达成目标的智能体构成联盟：

分类：非重叠联盟结构生成（形成的各联盟之间互不相交）/重叠联盟结构生成

方法：基于动态规划的启发式算法/基于图论的启发式算法/智能算法与进化算法

* 任务分配——将每一个可能的联盟的资源和任务进行组合分配，求得相应的联盟值，用于评估联盟的收益，并根据收益进行任务分配：

问题：将任务分派给哪个智能体联盟

目标：使所有智能体的收益和最大

智能体联盟的特点：一个智能体联盟可以完成多项任务；同个智能体联盟完成不同的任务有不同的收益；不同智能体联盟完成相同的任务有不同的收益

* 效用分配——为保证联盟中的智能体能够长期稳定的在当前联盟中提供资源，需进行合理的收益分配：

原则：

1）集体理性原则：联盟效用要分配给联盟中的全部联盟成员；

2）个体理性原则：加入联盟中的智能体所获得的效用要大于智能体独立工作所获得的收益

如果智能体对收益不满，那么其将退出联盟，破坏联盟形成的稳定性

强化学习：

基本思想：强化学习是训练自主学习系统的主要方法之一，解决了智能体如何通过与环境交互学习以最大化累积奖励值的问题。

研究背景与待解决问题：在多智能体协作的情景下，每个智能体所面对的环境包括外界环境与其他智能体。智能体是动态变化的，造成了多智能体强化学习所面临的环境是不稳定的。

解决方案：

* 智能体独立训练：每个智能体独立学习，互不干涉，将其他智能体视为环境，通过智能体与环境的交互间接实现智能体之间的交互。由于存在多个智能体的环境极其不稳定，这种方法效果较差，尤其在部分可观察的环境中。
* 智能体间共享参数：智能体间共享参数可以充分利用所有样本训练出一个较好的策略，提高样本的利用效率，然而该方式会交互没有必要的信息，很难实现智能体间的协作，策略收敛速度较慢。由于学习样本极易过期，使用过期的样本训练策略会导致策略性能下滑，所以使用参数共享机制需要识别并处理过期的样本。

**博弈论**

博弈三要素：

1. 参与者：至少2个
2. 策略集：一组如何执行具体行为的备选项，备选项称为参与者的可能策略。
3. 收益：每个行为策略均会使参与人得到一个收益，此收益会受到博弈中其他参与者策略选择的影响，且同一种策略，不同的参与人得到的收益可能不同。

基本假设：

1. 已知的收益矩阵：每个参与人对于收益矩阵均有充分的了解。
2. 理性的参与者：参与者始终追求博弈中自身收益最大化。
3. 独立的决策：参与者进行决策时不能进行协商。

不同策略：

最佳应对策略：当参与人乙采取策略，如果甲采用策略得到的收益大于等于任何其他策略，则称甲的策略是乙的策略的最佳应对。，为不是策略的任何策略。

严格最佳应对策略：当参与人乙采取策略，如果甲采用策略得到的收益大于任何其他策略，则称甲的策略是乙的策略的严格最佳应对，。

占优策略：如果甲的策略，对于乙的每一个策略都是最佳应对，则称策略为甲的占优策略。

严格占优策略：如果甲的策略，对于乙的每一个策略都是严格最佳应对，则称策略为甲的严格占优策略。

弱劣势策略：对于乙的每一个策略，如果甲的策略收益总小于等于另一个策略（至少有一个收益关系为小于），则称策略为甲的弱劣势策略。

严格劣势策略：对于乙的每一个策略，如果甲的策略收益总小于另一个策略，则称策略为甲的严格劣势策略。

纳什均衡：

存在性定理：具有有限参与者和有限纯策略的博弈一定存在纳什均衡。

假设：在纳什均衡状态，任意一方均没有动机去改变策略；如果两人同时改变策略可能会更好，但单方面改变不会得到额外好处。

纯策略纳什均衡：

定义：如果为的最佳应对，且同时为的最佳应对，则称策略组为纳什均衡。

多重均衡博弈（多个纯策略纳什均衡点）：

协调博弈：判断参与者的决策，可以通过局部影响。

非对等协调博弈：可以预期参与者会选择收益更高的纳什均衡点，或通过参与人决策偏好来预测。

混合策略纳什均衡：

定义：博弈参与人甲采用混合策略为参与人乙采用混合策略的最佳应对，且博弈参与人乙采用混合策略为参与人甲采用混合策略的最佳应对。

混合策略：参与人将以一定的概率进行策略的选择，每种概率选择对应一种混合策略，参与人的各策略概率和为1。

预期收益：假设参与人甲以一定概率采取策略，以概率采取策略，假设参与人乙以一定概率采取策略，以概率采取策略，则乙的收益为。

求解方法：混合策略纳什均衡点的收益=其包含的纯策略的收益；混合策略包含纯策略的收益相等

零和博弈：参与人收益和为0，一方获益造成另一方损失，不存在纯策略纳什均衡。

多策略博弈求解：迭代剔除劣势策略，缩小策略空间。

重复博弈：

定义：多次同结构博弈的重复，其中每次博弈称作“阶段博弈”。

分类：有限次重复博弈/无限次重复博弈

收益计算：，为折损因子，为在第轮的收益。

有限重复博弈：阶段博弈重复次数有限，因此最后一轮博弈相当于单轮博弈；当双方都知道最后一轮结果时，上一轮博弈同样相当于单轮博弈，可以进行逆向推理。

无限重复博弈：

TFT策略：首先采取合作策略C，不会首先背叛；对于对方的背叛行为，一定报复即采取背叛行为；如果对方改变为合作行为，则同样改为采取合作策略。

TFT’ 策略：在第一轮选择D策略，随后选择对手上一轮采用的策略。

合作动机：如果一方攻击另一方，对方势必反击，己方也会伤亡；如果一方不攻击，对方可能也不会攻击。

存在噪声的无限重复博弈：

TFTT策略：初始采用C策略并维持；若对手连续两次采用D策略，则下一轮采用D策略；如果对手采用C策略，则下一轮采用C策略。

GTFT策略：发现对手采用D策略后，在下一轮仍有一定概率选用C策略。

思路：

