

第 1 章

绪论

1.1 群体智能简介

1.1.1 群体智能的含义与特点

群体智能 (swarm intelligence, SI) 是人工智能领域中重要的概念, 最早由 Gerardo Beni 和 Jing Wang 于 1989 年提出^[1], 用以研究细胞机器人系统, 如兰顿蚂蚁和康威生命游戏。群体智能在有的文献中也称为集群智能 (collective intelligence, CI)^[2]。在动物社会里, 有些个体微不足道, 例如, 蚂蚁、蜜蜂、鱼等, 然而, 这些个体组成的群体能够解决单个个体难以解决或不可能解决的复杂问题。群体的协调能力都令人难以置信, 可以更容易地捕捉到更大的猎物, 或者更好地保护自己免受捕食者的攻击^[3]。对群体行为的研究促使人们认识到, 群体生活也可以帮助解决超出单个动物能力范围的认知问题^[4-5], 虽然个体动物在认知上相对简单, 在其能达到的目标上受到限制, 然而, 群体的能力则能做出惊人的成就^[6]。

群体智能并不是简单的多个体的集合, 而是超越个体行为的一种更高级表现, 这种从个体行为到群体行为的演变过程往往极其复杂, 是群体中各个个体随时间相互作用的模式和结果, 以至于很难由个体的简单行为来预测和推演。这称为涌现 (emergence), 指推演一个复杂系统中某些新的、相关的结构、模式和性质 (或行为) 的过程。

可以用图 1-1 来说明群体中的个体比个体或小群体中的个体做出更好决策的过程^[7]。个体用黑色圆圈表示, 从源头到接收者的信息流用箭头表示。如图 1-1(a) 所示, 当个体认知能力水平提高时, 个体之间就没有信息流动。相反, 在一个群体中可以减少对捕食风险的感知, 允许个体将更多的认知资源分配到其他任务中, 或者在风险不受 (或更少) 被捕食群体规模影响的情况下对捕食者保持警惕。如图 1-1(b) 所示, 信息可以从所有或部分成员集中, 在那里进行处理, 并作出总体决

策,或将信息提供给群内成员使用。如图 1-1(c)所示,在领导力方面,信息从一个(或几个)具有相关知识的个体流向其他群体成员。如图 1-1(d)所示,在群体智能中,只有信息交流,没有明显的集中者领导关键个体。

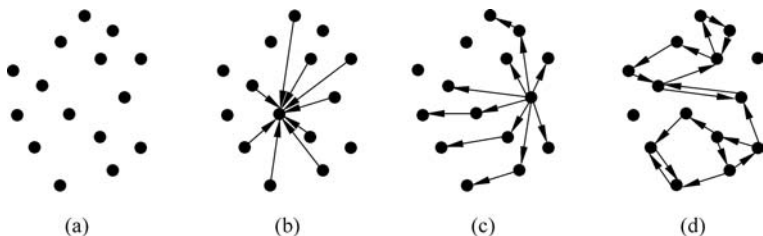


图 1-1 群体智能信息交流

对于一个由众多简单个体组成的群体,若其个体具有能通过彼此间的简单合作来完成一个整体任务的能力,则称该群体具有“群体智能”。群体智能中的“群体”指的是一组相互之间通过改变局部环境信息可以进行直接或间接通信的个体,这些个体能够合作进行分布式问题的求解。群体智能中的“个体”仅具有较为简单的能力,这种能力可用某一简单的功能函数来表示。个体间的相互作用、间接通信也称为外激励(stigmergy)。

在自然界中,通过群体中个体间的相互协作完成任务的例子有:

(1) 白蚁建造的大巢穴结构的复杂度是单个白蚁凭借自己的能力所无法实现的;

(2) 在一个蚂蚁群中,在没有任何中心管理者和任务协调员的情况下,其任务都是动态分配的;

(3) 蜜蜂通过摆动跳舞实现了最佳的觅食行为,觅食行为作为简单的轨迹跟踪行为也在蚁群中出现;

(4) 鸟群中的鸟和鱼群中的鱼会自组织成最佳的空间模式,鸟群或鱼群通过声音和视觉感知的通信基于少量的邻近个体来确定它们的行为(如调整自己的方向和速度);

(5) 猎食者(如一个狮群)表现出的猎食策略比被猎食者更精明;

(6) 细菌利用分子(类似信息素)通信,共同保持对环境变化的跟踪;

(7) 黏菌由非常简单且能力有限的分子有机体组成,然而,在缺少食物的时期,它们会聚集形成一个移动的块,以便将聚集在一起的个体运送到新的食物区域。

关于群体智能的定义在已发表或出版的文献中都不尽相同。如有的定义为:两个或更多个独立收集信息的个体通过社交互动进行信息交流,并为单个个体无法解决的认知问题提供群体的含义。也有定义为:是指在某群体中,若存在众多无智能的个体,它们通过相互之间的简单合作所表现出来的智能行为。蚁群算法

提出者 Dorigo^[8]对群体智能如此描述：指具有简单智能的个体通过相互协作和组织表现出群体智能行为的特性，具有天然的分布式和自组织特征。

总之，群体智能的核心是由众多简单个体组成的群体能够通过相互之间的简单合作来实现某一较复杂的功能，完成某一较复杂的任务。群体智能可以在没有集中控制并且缺少全局信息和模型的前提下，为解决复杂的分布式问题提供了可能。

群体智能的特点主要包括^[9]：

- (1) 灵活性(flexibility)，群体可以适应随时变化的系统或网络环境；
- (2) 鲁棒性(robustness)，没有中心或者统一的控制，即使单个个体失败，整个群体仍然具有完成任务的能力，不会出现由于某一个或者某几个个体的故障或失败而影响整个问题的求解；
- (3) 自组织(self-organization)，活动既不受中央控制，也不受局部监管。

群体智能的优点主要体现在^[9]：

- (1) 分布性(distribution)，群体中相互合作的个体是分布的，这样更能够适应当前网络环境下的工作状态；
- (2) 简单性(simplicity)，系统中每个个体的能力十分简单，个体的执行时间比较短，并且实现也比较简单；
- (3) 可扩充性(extensibility)，可以仅仅通过个体之间的间接通信进行合作，系统具有很好的可扩充性，因为系统个体的增加而引起的通信开销的增加很小。

Millonas 提出了群体智能遵循的五条基本原则^[10]：

- (1) 邻近原则(proximity principle)，群体能够进行简单的空间和时间计算；
- (2) 品质原则(quality principle)，群体能够响应环境中的品质因子；
- (3) 多样性反应原则(principle of diverse response)，群体的行动范围不应该太窄；
- (4) 稳定性原则(stability principle)，群体不应在每次环境变化时都改变自身的行为；
- (5) 适应性原则(adaptability principle)，在所需代价不太高的情况下，群体能够在适当的时候改变自身的行为。

1.1.2 群体智能算法的分类

群体智能算法是早期学者基于群体智能，针对群体行为特征规律的研究，受社会昆虫(如蚂蚁、蜜蜂)、群居脊椎动物(如鸟群、鱼群和兽群)或其他群居生物的启发，提出一系列具备群体智能特征的算法用来解决复杂问题。自 1992 年意大利学者 Dorigo 从蚁群寻找最短路径的现象中受到启发在他的博士论文中提出蚁群优化(ant colony optimization, ACO)理论开始，群体智能算法作为一个理论被正式提

出,并逐渐吸引了大批学者的关注,应用于不同的领域,从而掀起了研究高潮。1995年,Kennedy等^[11]学者模拟鸟群觅食行为提出了粒子群优化算法(particle swarm optimization,PSO),由于PSO模型简单、参数少得到了广泛的研究与应用,进一步推进了群体智能算法的研究。近年来,众多研究学者模拟不同的群体特征陆续提出了不同的群体智能算法。

如图1-2所示,按照启发式方法的发展进行分类。对于优化问题的求解,最早通过数学方法寻求最优解。而启发式算法是受大自然的运行规律或面向具体问题的经验、规则启发出来的方法,在可接受的计算时间或空间范围内给出待解决优化问题的可行解,该可行解与最优解的偏离程度一般不能被预计。由于早期启发式方法在求解大规模问题时容易陷入局部最优,求解效果并不理想,研究学者根据自然界现象提出了自然启发式方法。在自然启发式方法中包括:一类是基于生物行为的算法,即生物启发式方法;另一类是模拟物理现象等自然现象的,如模拟退火算法。而在生物启发式方法中,一类是模拟动物群体行为的群体智能算法,也是本书后续章节中主要讲的内容;另一类是模拟生物进化的进化计算;还有一类是模拟其他生物群体的算法,如文化算法、免疫算法等。

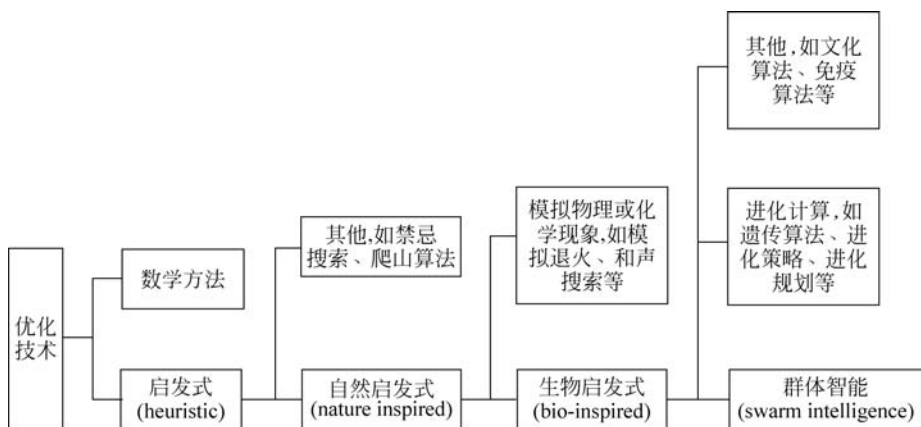


图 1-2 启发式方法分类

由于群体智能算法的灵感主要来自于模拟不同生物群体,按照不同群体的方式进行分类,可以分为昆虫类、细菌类、鸟群类、水生类以及陆生类等,如图1-3所示。由于新的模拟生物群体的算法仍在不断被学者提出,图1-3中无法包括所有的算法。

目前,由于每种算法经过改进后可以衍生出较多算法,为了能够对常用算法的提出思想有所了解,如表1-1所示,给出了常用算法的名称、提出作者、基本思想以及提出的年份^[12-13]。

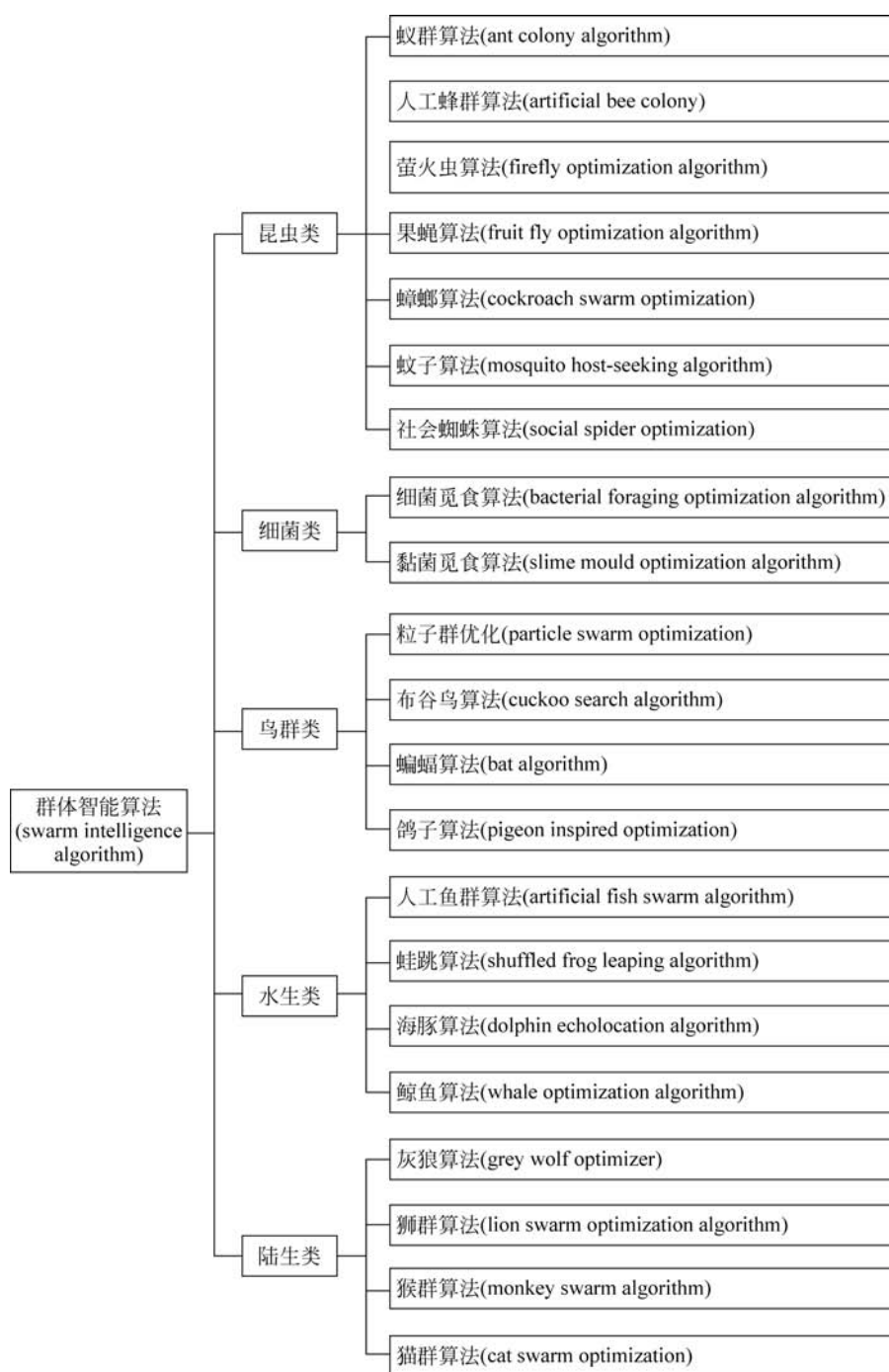


图 1-3 群体智能算法分类

表 1-1 部分群体智能算法基本信息

中文名称	英文名称	作者	基本思想	年份
蚁群算法	ant colony algorithm	Dorigo ^[8]	模拟蚂蚁觅食行为,通过分泌信息素进行个体协作	1992
粒子群优化	particle swarm optimization	Kennedy 等 ^[11]	鸟群飞行觅食行为	1995
细菌觅食算法	bacterial foraging optimization algorithm	Passino ^[14]	大肠杆菌的觅食行为	2002
人工鱼群算法	artificial fish swarm algorithm	Li 等 ^[15]	模拟鱼群觅食和集群游弋行为	2002
菌落趋化性	bacterial colony chemotaxis	Li 等 ^[16]	细菌对化学引诱剂的反应	2005
蜂群优化	bee colony optimization	Teodorovic 等 ^[17]	自然界的蜂群	2005
猫群优化	cat swarm optimization	Chu 等 ^[18]	猫的行为及其追踪、寻找等技能	2006
人工蜂群算法	artificial bee colony	Karaboga 等 ^[19]	蜜蜂的自然觅食行为	2007
布谷鸟算法	cuckoo search algorithm	Yang 等 ^[20]	几种布谷鸟的育雏寄生行为	2009
萤火虫算法	firefly optimization algorithm	Yang ^[21]	萤火虫的行为及闪光(生物发光过程)	2009
萤火虫群优化	glowworm swarm optimization	Krishnanand 等 ^[22]	荧光素诱导的萤火虫发光,用于吸引配偶/猎物	2009
蝙蝠算法	bat algorithm	Yang ^[23]	微生物的回声定位特征	2010
蟑螂算法	cockroach swarm optimization	Chen 等 ^[24]	蟑螂的社会行为	2010
狮群算法	lion swarm optimization algorithm	Rajakumar ^[25]	狮子的社会行为有助于保持哺乳动物在世界上的强壮	2012
灰狼算法	grey wolf optimization	Tang 等 ^[26]	狼通过躲避敌人来寻找食物和生存	2012
果蝇算法	fruit fly optimization algorithm	Xing 等 ^[27]	果蝇的行为	2013
社会蜘蛛算法	social spider optimization algorithm	Cuevas 等 ^[28]	相互作用的社会蜘蛛的合作行为	2013
海豚算法	dolphin echolocation algorithm	Wu 等 ^[29]	海豚回声定位、信息交流、合作	2016
鲸鱼算法	whale optimization algorithm	Mirjalili 等 ^[30]	座头鲸的社会行为——泡泡网捕猎策略	2016

群体智能算法具有以下特点^[31-34]。

(1) 智能性。群体智能算法通过向大自然界中的某些生命现象或自然现象学习,实现对问题的求解,这类算法中包含了自然界生命现象所具有的自组织、自学习和自适应性等特性。在运算过程中,通过获得的计算信息自行组织种群对解空间进行搜索。由于群体智能算法具有的这种优点,应用群体智能算法求解问题时,不需要已知待优化问题的梯度信息,也不需要事先对待求解问题进行详细数学建模。对于某些复杂性高的问题,高效求解成为可能。

(2) 隐含本质并行性。群体智能算法通过设定相应的种群进化机制完成计算,而种群内的个体则具有一定的独立性,个体之间完全是一种本质上的并行机制。如果使用分布式多处理机来完成群体智能算法,可以将算法设置为多个种群并分别放置于不同的处理机实现进化,迭代期间完成一定的信息交流即可(注:信息交流并不是必要的),迭代完成之后,根据适应度值优胜劣汰。所以,群体智能算法这种隐含的本质并行性,能够更充分利用多处理器机制,实现并行编程,提高算法的求解能力。其更加适合目前云计算等分布式计算技术迅速发展的背景。

(3) 解的近似性。群体智能算法通常来自于对大自然中某种生命或其他事物的智能协作进化现象的模拟,利用某种进化机制指导种群对解空间进行搜索。由于该类算法缺乏严格的数学理论支持,对于问题的解空间采用反复迭代的概率性搜索,所以群体智能算法会存在早熟或求解精度较低等问题,而这也是所有群体智能算法几乎都存在的弱点。所以,很多时候对求解的问题来说,群体智能算法仅仅得到的是一种近似最优解。但是,这种近似最优解可以一次提供多个,为决策者提供更多选择。

1.2 制造中的最优化问题

1.2.1 最优化问题

最优化问题普遍存在,即在特定的现实环境约束下,争取找到最优的解决方案,获得最佳的实践效果。最优化问题的应用领域极其广泛,如设计、加工、调度、物流配送、系统控制、工程、金融、医学等。例如,工程设计中怎样选择设计参数,使得设计方案既满足设计要求又能降低成本;车间自动引导车(automated guided vehicle, AGV)路径如何规划,使得 AGV 既能够满足生产需求又能效率更高成本最低;生产计划安排中,选择怎样的计划方案使得效率和利润都得到提高等。在智能制造的生产过程各阶段实现最优化,对提高生产效率和效益、节省资源、提高顾客满意度等具有重要的作用。

最优化问题,就是在满足一定的约束条件下,寻找一组参数值,以使某些最优性度量得到满足,使系统的某些性能指标达到最大或最小^[35]。最优化问题根据其

目标函数、约束函数的性质以及优化变量的取值等可以分成许多类型,每一种类型的最优化问题根据其性质的不同都有其特定的求解方法。

通常情况下,最优化问题是寻找最小值问题(寻找最大值问题可以转化为寻找最小值问题)。不失一般性,以最小值问题为例,最优化问题的一般形式可以用以下两种方式表示。

一般式:

$$\left. \begin{array}{l} \min_{X \in \Omega} f(X) \\ \text{s. t. } h_j(X) = 0, j = 1, 2, \dots, n \\ s_i(X) \geq 0, i = 1, 2, \dots, m \end{array} \right\} \quad (1-1)$$

向量式:

$$\left. \begin{array}{l} \min_{X \in \Omega} f(X) \\ \text{s. t. } \begin{array}{l} \mathbf{h}(X) = 0 \\ \mathbf{s}(X) \geq 0 \end{array} \end{array} \right\} \quad (1-2)$$

式中, $f(X)$ 称为目标函数。优化过程就是定义域中在满足约束的情况下选择合适的 X ,使目标函数 $f(X)$ 达到最优值。 $s_i(X)$ 称为不等式约束,它的向量表示法可以写成 $\mathbf{s}(X)=[s_1(X), s_2(X), \dots, s_m(X)]^T$; $h_j(X)$ 称为等式约束,它的向量表示法可以写成 $\mathbf{h}(X)=[h_1(X), h_2(X), \dots, h_n(X)]$ 。

当约束函数 $\mathbf{s}(X)$ 和 $\mathbf{h}(X)$ 所限制的约束空间为整个欧氏空间时,上述最优化问题就简化为无约束优化问题,当然也可以通过某些方法(如惩罚函数法)将有约束优化问题转换为无约束优化问题。

如果目标函数 $f(X)$ 和约束函数 $\mathbf{s}(X)$ 和 $\mathbf{h}(X)$ 都是线性的,并且变量 X 是连续的,那么称其为线性规划问题。如果 $f(X)$ 、 $\mathbf{s}(X)$ 和 $\mathbf{h}(X)$ 中至少有一个为非线性函数时,则上述问题为非线性规划问题。非线性规划问题相当复杂,其求解方法多种多样,但到目前为止仍然没有一种有效地解决所有问题的方法。

当优化变量 X 仅取整数值时,上述问题为整数规划问题,特别是当 X 仅能取0或1时,上述问题即为0-1整数规划问题。由于整数规划问题属于组合优化范畴,其计算量随变量维数的增长而指数增长,所以存在着“维数灾难”问题^[35]。

1.2.2 最优化问题的分类

最优化问题存在多种不同分类方法,结合在群体智能中的应用情况,下面介绍两种分类。

(1) 按照最优化问题对象,可以分为组合优化和函数优化。

组合优化和函数优化的区别在于应用领域优化问题对象是离散的还是连续的。组合优化问题的可行空间是离散的值,而函数优化问题的可行空间是一定范围内的连续值。

函数优化算法性能的比较通常是基于基准函数进行的,常用的基准函数有 Schwefel、Rosenbrock、Griewank、Sphere、Quartic、Ackley、Kowalik、Hartman、Penalized 等。鉴于许多工程问题或工艺参数问题存在约束条件,受约束的函数优化问题也一直是优化领域关注的主要对象。

组合优化的对象是解空间的离散状态,组合优化往往涉及排序、分类、筛选等问题,典型的组合优化问题有旅行商问题(traveling salesman problem)、加工调度问题(scheduling problem)、工艺规划问题(process planning)、聚类问题(clustering problem)、背包问题(knapsack problem)和着色问题(graph coloring problem)等。在智能工厂与智能车间领域中,存在较多组合优化问题,如 AGV 的路径问题、车间调度问题等。

(2) 按照求解问题的优化方法特点,可以分为全局最优和局部最优。

针对一定条件/环境下的一个问题/目标,若一项决策和所有解决该问题的决策相比是最优的,就可以被称为全局最优。

将上述定义用数学公式表示为:在无限制环境集合 R 内,假设限制条件/环境为集合 D (D 包含于 R),问题代价或目标函数为 $f(x)$,其中 x 指决策,那么全局最优就是指决策满足 $f(x_0) = \min\{f(x)\}, x_0 \in D$ 。

和全局最优不同,局部最优不要求在所有决策中是最好的。

针对一定条件/环境下的一个问题/目标,若一项决策和部分解决该问题的决策相比是最优的,就可以被称为局部最优。

将上述定义用数学公式表示为:按照上面的定义,对于 D 内的一个子集 D_n ,局部最优就是指决策满足 $f(x_0) = \min\{f(x)\}, x_0 \in D_n$ 。

对于优化问题,尤其是最优化问题,总是希望找到全局最优的解或策略,但是当问题的复杂度过高,要考虑的因素和处理的信息量过多的时候,我们往往会倾向于接受局部最优解,因为局部最优解的质量不一定是差的。尤其是当我们有确定的评判标准表明得出的解是可以接受的话,通常会接收局部最优的结果。这样,从成本、效率等多方面考虑,才是实际工程中会采取的策略。

1.2.3 计算复杂性与 NP 问题

算法的时间和空间复杂性对计算机的求解能力有重大影响,算法对时间和空间的需要量称为算法的时间复杂性和空间复杂性。问题的时间复杂性是指求解该问题的所有算法中时间复杂性最小的算法的时间复杂性。问题的空间复杂性也可类似定义。

算法或问题的复杂性一般表示为问题规模 n 的函数,时间复杂性记为 $T(n)$,空间复杂性记为 $S(n)$ 。在算法分析和设计中,沿用实用性的复杂性概念,即把求解问题的关键操作,如加、减、乘、比较等运算指定为基本操作,算法执行基本操作的次数则定义为算法的时间复杂性,算法执行期间占用的存储单元则定义为算法

的空间复杂性。在分析复杂性时,可以求出算法的复杂性函数 $p(n)$,也可用复杂性函数主要项的阶 $O(p(n))$ 来表示。若算法 A 的时间复杂性为 $T_A(n) = O(p(n))$,且 $p(n)$ 为 n 的多项式函数,则称算法 A 为多项式算法。时间复杂性不属于多项式时间的算法统称为指数时间算法。

按照计算复杂性理论研究问题求解的难易程度,可把问题分为 P 类、NP 类和 NP 完全类。P 类问题是指对具有多项式时间求解算法的问题类;反之,为非多项式确定问题,即 NP 问题。在实际工程应用中,更多的属于 NP 完全类问题。有兴趣的读者可以阅读王凌^[36]编写的《智能优化算法及其应用》一书了解它们之间的详细内容。

1.3 群体智能的应用与发展

1.3.1 群体智能的应用

随着群体智能理论以及群体智能算法的发展,其在数学、物理学、计算机科学、社会科学、经济学以及工程应用等领域得到了广泛的应用^[37]。本书主要针对群体智能算法在智能加工、智能车间和制造服务中的应用来进行讲解,如表 1-2 所示,给出在本书中群体智能算法的应用场景。

表 1-2 群体智能应用

序 号	应用方向	具体应用场景
1	智能加工	a. 加工工艺参数优化:选择合适的工艺参数可以提高产品精度、降低成本、减少能耗; b. 机器人路径规划:保证机器人移动过程中不碰到障碍物,并满足路线最短等性能指标; c. 增材制造加工优化:加快产品的制造周期,具有较高的柔性,提高材料利用率
2	智能车间	a. 智能工艺规划:对产品的质量、成本和生产效率具有重要影响; b. 智能车间调度:对车间生产调度优化,可提高设备利用率,降低生产成本; c. 装配序列与产线平衡:好的装配序列和合理的装配线布局,将直接影响到装配制造系统的生产效率和产品质量
3	制造服务	a. 供应链物料采购优化:帮助企业更好地优化采购管理,帮助供应商预测企业需求; b. 云制造服务组合优化:提升云制造服务组合的柔性,在异常发生时,结合相应的调整策略自适应地进行处理; c. 维护,维修和大修(MRO)优化:考虑维修时间及维修成本,求解更为复杂的 MRO 服务调度问题