

哈尔滨理工大学

硕士学位论文

基于遗传算法的BP神经网络在多目标优化中的应用研究

姓名：朱文龙

申请学位级别：硕士

专业：计算机应用技术

指导教师：丁华福

20090301

基于遗传算法的 BP 神经网络在多目标优化中的应用研究

摘要

多目标优化起源于许多实际复杂系统的设计、建模和规划问题，这些系统涉及的领域包括工业制造、资本预算、网络通讯、道路规划等等。现实生活中，几乎每个重要的决策和预测都需要考虑各种制约条件，并处理若干目标的相互冲突，这就涉及多个目标的优化。因为这些目标并不是独立存在的，对其中一个目标进行优化必须以牺牲其它目标为代价，并且各目标的单位又往往不一致，这就导致很难客观地评价多目标优化求解的优劣性。传统的求解多目标优化的方法通常存在许多缺陷，如各目标加权值的分配带有较大的主观性，优化目标仅为各目标加权和，过程中各目标优化的不可操作性等。

本文着重对人工智能中两个热点问题遗传算法和人工神经网络进行分析和改进，尝试对这两种方法进行融合，并将其应用到多目标优化领域。论文主要包括：

首先，介绍遗传算法的原理，重点介绍现今经典的多目标遗传算法的优缺点，分析多目标遗传算法设计中所要解决的问题：适应度分配、多样性保持、收敛性。针对简单遗传算法应用过程中所存在的不易收敛，结果常常陷入局部最优等缺点，在改进型非劣分层遗传算法基础上提出改进的并行混合非劣分类遗传算法。多个算例测试表明，改进的算法在解的多样性和收敛性方面都有不俗表现，其性能明显优于改进型非劣分层遗传算法。

其次，通过对 BP 神经网络的关键技术及其算法的研究，针对 BP 神经网络的限制与不足进行分析，提出一种改进技术。

最后，将改进后的遗传算法与 BP 神经网络融合，提出基于遗传算法的 BP 神经网络融合算法。利用基于遗传算法的 BP 神经网络对多目标优化问题进行求解，充分发挥遗传算法的全局搜索能力和 BP 算法的局部搜索能力，加快收敛速度，提高收敛精度。将其应用于多目标车间作业动态调度实例中，也取得较好的效果。

关键词 遗传算法；BP 神经网络；多目标优化；算法融合

Application Research of BP Neural Network Based on Genetic Algorithm in Multi-objective Optimization

Abstract

Multi-objective optimization orientates from the design, modeling and programming of complicate systems including industrial manufacture, budget, network communication and road layout. In our daily life, a variety of constraints should be considered and the confliction among different objectives needs to be solved in almost every important decision and prediction. So this refers to the issue of multi-objective optimization. As these objectives are not independent, the optimization of one objective is achieved at the cost of the other objectives. The objectives have no common units and thus it is hard to evaluate the solution of multi-objective problems. Traditional methods have many disadvantages, for instance, the distribution of each objective's power is often quite subjective, the final objective is only represented by the weighted sum, and the optimization process can not be controlled.

In this paper, we focus on the two aspects of artificial intelligence, genetic algorithm and artificial neural network. After the analysis, we try to combine these two technologies and apply it into the solving process of multi-objective optimization. The main contents of this thesis are as follows:

Introduction of genetic algorithm theory. We will introduce the shortcomings of current multi-objective genetic algorithm, and discuss major issues waited to be overcome, such as the allocation of fitness, diversity maintenance and convergence. Due to that simple genetic algorithm is hard to converge in practice, and often encounters a local optimum, so we bring forward an improved intercurrent mixed nondominated sorting genetic algorithm II method based on nondominated sorting genetic algorithm II. Several algorithm cases indicate that the improved method is obviously superior to nondominated sorting genetic algorithm II, not only in diversity of solution but also convergence.

By analyzing the shortcomings of BP neural network, the study on key technologies and algorithm of BP neural network is conducted, and an improved method is proposed.

Combining improved genetic algorithm and BP neural network, we put forward the BP neural network based on genetic algorithm. Taking advantage of both genetic algorithm's global search capability and BP algorithm's local optimize performance, improved method's convergence accuracy and rate are raised in solving multi-objective optimization. In practice, a favorable performance is achieved in the of multi-objective workshop dynamic scheduling.

Keywords genetic algorithm, BP neural network, multi-objective optimization, algorithm syncretization

哈尔滨理工大学硕士学位论文原创性声明

本人郑重声明：此处所提交的硕士学位论文《基于遗传算法的 BP 神经网络在多目标优化中的应用研究》，是本人在导师指导下，在哈尔滨理工大学攻读硕士学位期间独立进行研究工作所取得的成果。据本人所知，论文中除已注明部分外不包含他人已发表或撰写过的研究成果。对本文研究工作做出贡献的个人和集体，均已在文中以明确方式注明。本声明的法律结果将完全由本人承担。

作者签名：李文龙 日期：2009 年 3 月 20 日

哈尔滨理工大学硕士学位论文使用授权书

《基于遗传算法的 BP 神经网络在多目标优化中的应用研究》系本人在哈尔滨理工大学攻读硕士学位期间在导师指导下完成的硕士学位论文。本论文的研究成果归哈尔滨理工大学所有，本论文的研究内容不得以其它单位的名义发表。本人完全了解哈尔滨理工大学关于保存、使用学位论文的规定，同意学校保留并向有关部门提交论文和电子版本，允许论文被查阅和借阅。本人授权哈尔滨理工大学可以采用影印、缩印或其他复制手段保存论文，可以公布论文的全部或部分内容。

本学位论文属于

保密 ☐，
不保密 ☒。

在 年解密后适用授权书。

(请在以上相应方框内打√)

作者签名：李文龙 日期：2009 年 3 月 20 日

导师签名：丁华福 日期：2009 年 3 月 20 日

第1章 绪论

在日常生活中，人们经常遇到需要使多个目标在给定可行区域上尽可能最优的决策问题。如投资问题，人们希望所投入的资金最少、风险最小化而所获得的投资收益最大。这种多个数值目标在给定区域上最优化问题就是多目标优化问题(Multi-objective Optimization Problem, MOP)。现实中几乎每个重要的决策和预测问题上都要在考虑不同约束的同时处理若干相互冲突的目标，这些问题都涉及多个目标的优化，这些目标并不是独立存在的，它们往往是耦合在一起的互相竞争、相互束缚的目标，每个目标具有不同的物理意义，对其中一个目标优化必须以其它目标作为代价，而且各目标的单位又往往不一致。因此，多目标优化问题最终是寻找一组既满足约束条件同时使总目标函数最优化的决策变量的取值，其中组成总目标函数的元素是子目标函数^[1]。

传统的多目标优化问题求解方法是將各个子目标聚合成一个加权求和的单目标函数，系数由决策者决定，然后采用单目标优化算法求解。但这样做存在以下缺点^[2]：

1. 每个目标具有不同的物理意义，不易作比较。
2. 系数的决定带有较大的主观性，并且优化目标仅为各目标加权和，优化过程中各目标的优化不可操作。
3. 对 Pareto 最优前端的现状很敏感，不能很好地处理前端的凹部。
4. 各目标互相竞争、相互束缚，使得加权目标函数的拓扑结构十分复杂。另外许多工程因其优化问题性质的复杂性，很难用传统的方法求解。
5. 求解问题所需的与应用背景相关的启发式知识经常不能获得，导致无法正常实施优化。

20 世纪 80 年代中期以来人工智能的进化算法(Evolutionary Algorithm, EA)开始应用于该领域，其中遗传算法是迄今为止进化算法中应用最多、比较成熟、广为人知的算法，在求解复杂优化问题有着巨大潜力并且成功应用在工业工程领域中，因而受到众多学者的广泛关注。

本章首先将介绍人工智能的两个重要分支遗传算法和 BP 神经网络的基本理论、研究现状以及应用特点，给出多目标优化问题的基本概念以及求解多目标优化问题的传统方法，最后在此基础上给出本文的组织结构及所做的主要工作。

1.1 遗传算法

1.1.1 遗传算法的基本理论

遗传算法是仿真生物遗传学和自然选择机理,通过人工方式所构造的一类搜索算法,从某种角度上说,遗传算法是对生物进化过程进行的数学方式仿真。二十世纪 60 年代中期,霍兰德 (J. H. Holland)在他的著作《Adaptation in Natural and Artificial System》中提出了遗传算法的思想,将自然界的进化方法应用于实现求解复杂问题的自动程序设计中,为所有的适应系统建立了一种通用理论框架,并展示了如何将自然界的进化过程应用到人工系统中去,认为所有的适应问题都可以表示为“遗传”问题,并用“进化”方法来解决^[3]。目前遗传算法已经和人工神经网络、模糊逻辑相结合,形成一门新的综合性学科——计算智能^[4]。

基本遗传算法可定义为一个 8 元组^[5]: $SGA=(C,E,P_0,M,\Phi,\Gamma,\Psi,T)$, 式中: C 为个体的编码方法; E 为个体适应度评价函数; P_0 为初始种群; M 为群体大小; Φ 为选择算子; Γ 为交叉算子; Ψ 为变异算子; T 为遗传运算终止条件。这里种群指遗传算法在最优解的搜索过程中,一般从原问题的一组解出发改进到另一组较好的解,再从这组改进的解出发进一步改进,如此迭代直到满足条件。在算法中,每一组解称为“种群”,而每一个解称为一个“个体”。新种群的产生主要依靠选择算子、变异算子与交叉算子。

一般遗传算法包括以下几个步骤:

1. 随机初始化种群 P 。
2. 初始化进化代数计数器 t 。
3. 对当前群体 $P[t]$ 中的每个个体 x_i 计算适宜度 $F(x_i)$, 适宜度表示该个体性能好坏。
4. 应用选择算子产生中间代群体 $P[t]$, 对 $P[t]$ 应用其他的算子, 产生新一代群体 $x(t+1)$, 这些算子的目的在于扩展有限个体的覆盖面, 体现全局搜索的思想。
5. 个体的交叉、变异操作。
6. 评价新的群体, 计算适宜度 $F(x_i)$ 。
7. 终止条件判断。若不满足终止条件, 则 $t++$, 转到第 3 步, 继续进行进

化操作过程；若满足终止条件，则输出当前最优个体，算法结束。

1.1.2 遗传算法的研究现状

在许多领域，遗传算法已经得到了广泛的应用。遗传算法可以并行地搜索、评估问题空间中的多个解，一般具有较好的全局寻优能力，但在应用过程中也暴露出了许多缺点。主要表现：

1. 全局搜索能力极强而局部寻优能力较差。
2. 对搜索空间变化的适应能力差。
3. 易出现早熟收敛现象。
4. 算法在交叉、变异的进化过程中随机性较强，致使搜索效率低下，具体表现为进化迭代过程中会出现子代最优个体劣于父代最优个体的“退化”现象^[6, 7]。

为了解决上述问题，国内外学者对遗传算法的基本理论进行了深入的研究，提出了很多改进方法。如模拟退火遗传算法：模拟退火算法的基本思想是通过模拟高温物体退火过程的方法来找到优化问题的全局最优或近似全局最优解^[8, 9]。免疫遗传算法：人工免疫算法受生物免疫系统原理的启发，针对求解问题特征进行人工疫苗接种，可充分利用问题本身的信息，和遗传算法结合时，遗传算法的全局搜索能力及免疫算法的局部优化相配合，可大大提高搜索效率^[10]。小生境遗传算法：生物学上，小生境指在特定环境中的一种组织功能，它将每一代个体划分为若干类，每个类中选出若干适应度较大的个体作为一个类的优秀代表，组成一个新种群，再在同一种群中以及不同种群之间进行杂交、变异，产生新一代个体群，同时采用预选择机制或排挤机制或共享机制完成选择操作^[11]。模糊遗传算法：即融合模糊优化设计思想的遗传算法，它把模糊优化和遗传算法优化结合起来，构成种混合优化的设计方法^[12, 13]。其目的是利用模糊优化设计的优点，克服一般遗传算法优化设计存在的不足，从而使系统的优化设计更灵活、方便，取得好的设计效果。

1.1.3 遗传算法特点及其在多目标优化中应用

在遗传算法应用中，应先明确其特点和关键问题，才能对这种算法深入了解，灵活应用，以及进一步研究开发。相比于其它优化算法，遗传算法主要特点是：

1. 遗传算法从问题解的种群开始搜索，而不是从单个解开始。

2. 遗传算法直接以适应度作为搜索信息, 无需导数等其它辅助信息。
3. 遗传算法使用多个点的搜索信息, 具有隐含并行性。
4. 遗传算法使用概率搜索技术, 而非确定性规则。
5. 遗传算法具有极强容错能力, 它通过选择、交叉、变异操作能迅速排除与最优解相差极大的个体。

概括起来说, 遗传算法具有通用、并行、稳健、简单与全局优化能力强等突出优点。

上述优点决定了遗传算法的使用范围应该是复杂、困难的全局优化问题, 而不是通常的数值优化问题。所以对于一些目标函数没有明确的解析表达式或者不可能恰好估值或目标函数有多个峰值、多目标等复杂、困难的优化问题, 已知的优化方法或者根本不可用, 或者可应用但效果不好。相比之下, 遗传算法不但保证可应用, 而且常常显得更为有效。

多目标优化问题中首次应用遗传算法的思想可以追溯到 1967 年, Rosenberg 在他的研究中提出了在模拟单细胞有机物的化学遗传特性中采用多属性研究方法, 也提到了可以考虑用遗传的搜索算法求解多目标优化问题^[14]。虽然他最终只应用了单一属性方法, 他的研究却开创了遗传算法应用于多目标优化领域的研究。90 年代后, 遗传算法在多目标优化中得到广泛应用, 例如, 1993 年, Fonseca 和 Fleming 提出了 MOGA 算法^[15], Srinivas 和 Deb 提出了 NSGA 算法^[16], Horn 和 Nafpliotis 也提出了 NPGA 算法^[17]。

1.2 BP 神经网络

1.2.1 BP 神经网络基本理论

误差反向传播神经网络(Error Back Propagation Neural Network, BPNN), 按神经网络的互联结构, BPNN 属于前馈型神经网络。1986 年 Rumelhart, Hinton 和 Williams 完整而简明地提出这种算法, 系统地解决了多层神经网络中隐单元层连接权的学习问题, 并在数学上给出了完整的推导。BP 算法很好地克服了简单感知器不能解决 XOR 和其他一些问题, 因而得到广泛的使用。

从数学上看, 神经网络是一种函数逼近的方法。一般的神经网络结构模型主要由输出层、输入层以及若干隐含层组成, 每一层中含有若干节点, 每一层的节点把前一层的输出作为输入, 输入层由外部提供信号^[18]。每一个节点的输入信号都对应着一个权值 w 。BP 神经网络就是依据它的训练算法而来的, 其

学习训练过程由网络输入信号正向传播和误差信号反向传播，并按有教师学习方式训练。在正向传播中，输入信息从输入层经隐含层逐层计算传向输出层，在输出层的各神经元输出对应输入模式的网络响应；如果输出层得不到期望输出，则误差转入反向传播，按减小期望输出与实际输出的误差原则，从输出层到中间各层，最后回到输入，层层修正各个连接权值。随着这种误差逆传播训练不断进行，神经网络对输入模式响应的正确率也不断提高，如此循环直到误差信号达到允许的范围之内或训练次数达到预先设计的次数为止。

1.2.2 BP 神经网络的特点与存在的问题

神经网络学习方法对于逼近实数值、离散值或向量值的目标函数提供了一种健壮性很强的方法。BP 算法是最常用的人工神经网络(Artificial Neural Network, ANN)学习技术。它适合具有以下特点的问题^[7]：

1. 要学习的目标函数是定义在可以用向量描述的实例之上的，向量由预先定义的特征组成。
2. 目标函数的输出可能是离散值、实数值或者由若干实数属性或离散属性组成的向量。
3. 训练数据可能包含错误。
4. 可容忍长时间的训练。
5. 可能需要快速求出目标函数值。
6. 人类能否理解学到的目标函数是不重要的。

BP 神经网络优点主要有：学习功能和自组织能力强；信息的分布存储，主要体现在权值的不同取值方面；允许不完全的输入信息。

BP 神经网络的核心是链式求导法则，但是要实现甚至应用 BP 神经网络还是有很多问题要解决的。其主要表现在：

1. 作用函数一般是应用在隐层中，而输入层和输出层可能采用其他的作用函数，这涉及到数据的归一化和反归一化的问题。
2. 训练中，会发现对某些初始值网络会逐渐收敛到允许误差内，而还有一些则无法收敛，不知道这是进入了平滑区还是局部最优点。
3. 输出的结果难以解释。
4. 采用的是最快下降法原理，找到是局部最优解，不保证获得全局最优解^[19]。

BP 算法属于梯度下降算法,不可避免地存在易陷入局部极小、收敛速度慢、误差函数必须可导、网络结构没有成型的理论指导等缺点,在提高收敛速度方面,目前已有许多学者做了研究并提出了很多方法^[20, 21]。

针对 BP 神经网络本身存在的问题,结合遗传算法并将其应用于神经网络已经得到很多人的关注,提出了不少有效的算法^[22],它们之间的相互补充可增强彼此的能力,从而获得更有力的表示和解决实际问题的能力。目前遗传算法与神经网络融合的主要目标集中在改进神经网络的性能^[23],得到一个实用有效的神经网络系统。其中 GA 用来解决 BP 网络中一些难题,如对输入样本的高品质要求,解释神经网络的黑箱问题。GA 通过对 BP 网络进行预处理,把关于解空间的知识内嵌到 BP 网络初始状态中,使 BP 网络计算工程量大大减少。遗传算法优化 BP 神经网络的基本思想:改变 BP 算法依赖梯度信息的指导来调整网络权值的方法,利用遗传算法全局性搜索的特点,寻找最为合适的网络连接权和网络结构。由于三层神经网络由输入层、输出层和隐层组成,而输入层、输出层节点的个数由建模样本决定,故在优化 BP 网络结构时,主要是优化它的隐节点的个数。遗传算法是一种非导数优化的随机优化方法,可以对一复杂的、非线性及不可微的函数实现全局搜索,即算法对局部搜索比较有效,因此为了使算法能很快的找到满意解,可以先用遗传算法对初始权值进行优化,在解空间中定位出较好的搜索空间。然后用 BP 算法在这些小的解空间中搜索出最优解。

1.3 多目标优化问题

1.3.1 多目标优化问题的数学模型

多目标最优化问题的通用框架如图 1-1 所示。由于在现实世界中的普遍存在性以及求解的困难性,促使人们付出了大量精力来寻求解决的方法,也是近多年来迅速发展起来的应用数学的一门新兴学科。多目标优化问题的相关描述:一般 MOP 由 n 个决策变量参数、 k 个目标函数和 m 个约束条件组成,目标函数、约束条件与决策变量之间是函数关系。最优化目标如下^[24]:

$$\begin{aligned} \min_{\bar{x} \in D} \bar{y} &= \bar{f}(\bar{x}) = (f_1(\bar{x}), f_2(\bar{x}), \dots, f_m(\bar{x})) \\ \text{subject to: } \bar{e}(\bar{x}) &= (e_1(\bar{x}), e_2(\bar{x}), \dots, e_m(\bar{x})) \leq 0 \end{aligned} \quad (1-1)$$

公式(1-1)中 $\bar{x}=(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n) \in X$, $\bar{y}=(\bar{y}_1, \bar{y}_2, \dots, \bar{y}_m) \in Y$, X 为决策空间, \bar{y}

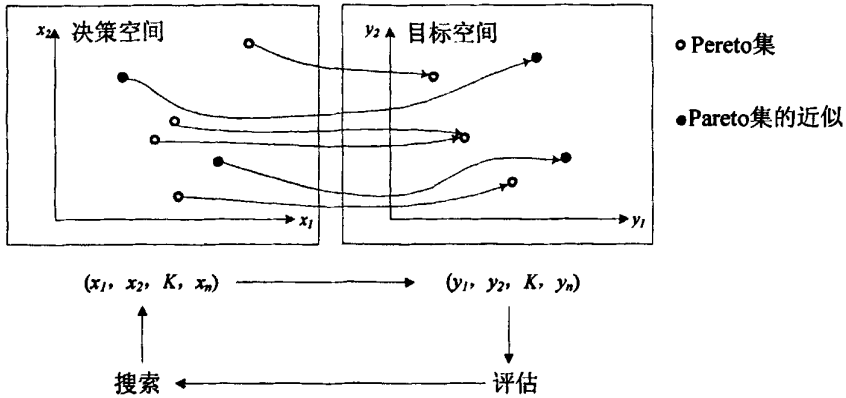


图 1-1 多目标优化问题通用框架

Fig. 1-1 General framework of multi-objective optimization problem

为目标向量， Y 为目标空间， D 为可行区域。

定义 1.1 Pareto 优胜

决策向量 $\bar{x}_u \in X$ Pareto 优胜决策向量 $\bar{x}_v \in X$ ，记为 $\bar{x}_u \prec \bar{x}_v$ ，当且仅当：

1. $\forall i \in \{1, 2, \dots, m\}$ 满足 $\bar{f}_i(\bar{x}_u) \leq \bar{f}_i(\bar{x}_v)$ 。
2. $\exists j \in \{1, 2, \dots, m\}$ 满足 $\bar{f}_j(\bar{x}_u) < \bar{f}_j(\bar{x}_v)$ 。

此时，也称决策向量 \bar{x}_v Pareto 劣于决策向量 \bar{x}_u 。若决策向量 \bar{x}_u 与决策向量 \bar{x}_v 不存在 Pareto 优胜关系，则称它们非劣。

从 Pareto 优胜的概念可以引出 MOP 问题最优解的判据^[14-12]。如图 1-2，A 点在 A、B、C、D、E 点中是最佳的，因为与其相关的决策向量 α 不劣于其他决策向量。这意味着 α 不能再提高任何指标性能，否则至少引起其他一项指标性能下降。诸如向量 α 这样的解方案称为 Pareto 最优解，有时也笼统称为非劣解、有效解或者满意解。

定义 1.2 Pareto 最优解

决策向量 $\bar{x}_u \in X$ 称为 X 上的 Pareto 最优解，当且仅当 $\neg \exists \bar{x}_v \in X$ 使得 $\bar{x}_v \prec \bar{x}_u$ 。

定义 1.3 Pareto 最优解集

对于给定的多目标优化问题 $\bar{f}(\bar{x})$ ，Pareto 最优解集 (ρ^*) 定义为： $\rho^* = \{\bar{x}_u \in X \mid \neg \exists \bar{x}_v \in X, \bar{x}_v \prec \bar{x}_u\}$ 。

Pareto 最优解集中的个体也称为非劣个体。

定义 1.4 Pareto 前沿

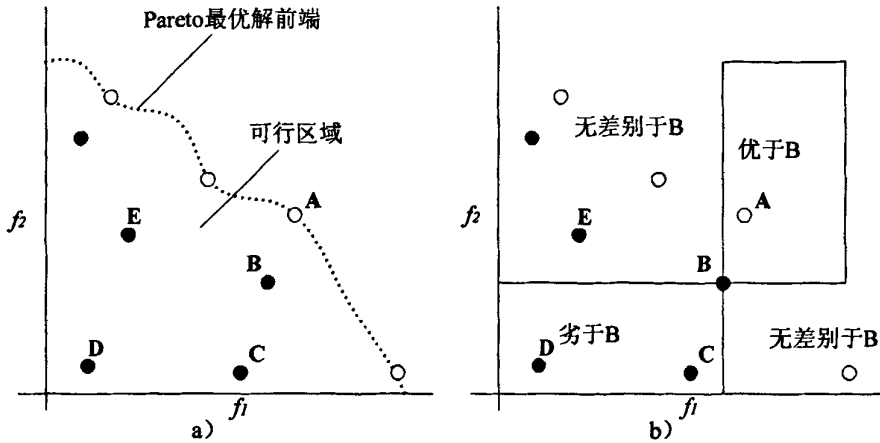


图 1-2 目标空间的 Pareto 最优解 a) 和解之间的关系 b)

Fig. 1-2 Relationship between Pareto optimality a) and solution b) in target space

对于给定的多目标优化问题 $\tilde{f}(\bar{x})$ 和 Pareto 最优解集 (ρ^*) , Pareto 前沿 (ρ^f) 定义为: $\rho^f = \{\bar{u} = \tilde{f}(\bar{x}_*) | \bar{x}_* \in \rho^*\}$ 。显然, Pareto 前沿是 Pareto 最优解集在目标空间中的像。

对上述有关定义可以总结如下:

1. 大多数情况下, 类似于单目标优化的最优解在多目标问题中是不存在的, 只存在 Pareto 最优解。多目标问题的一个 Pareto 最优解只是可以接受的“不坏”的解, 并且多目标优化问题大多数都具有很多个 Pareto 最优解。
2. 若一个多目标优化问题存在所谓的最优解, 则该最优解必定是 Pareto 最优解, 并且 Pareto 最优解也只由这些最优解组成, 不包含其它解。
3. 通常多目标优化问题的 Pareto 最优解是一个集合。对于实际应用问题的最后解决方案决策, 必须根据对问题的了解程度和决策人员的偏好, 从多目标优化问题的 Pareto 最优解集合里挑选出一个或部分解作为所求多目标优化问题的最优解。

1.3.2 多目标问题的研究现状

早在 1772 年 Franklin 就提出了多目标问题矛盾如何协调的问题。但国际上一般认为多目标最优问题最早由法国经济学家 V.Pareto 在 1896 年首次从数学角度提出了多目标最优决策问题, 并引入 Pareto 最优解的概念, 同时,

Kuhn 等人给出了向量极值问题有效解的必要条件^[25]。至此,多目标优化逐渐受到人们的关注。从 20 世纪 50 年代后国外学者在多目标优化问题上提出了如加权和法(Weighted Sum Method)、目标规划法(Goal Programming)、最小-最大法(Min-Max Approach)等基于权重的古典多目标优化方法^[26]。1953 年,Arron 等人对凸集提出了有效点的概念,从此多目标优化逐渐受到人们的关注;1963 年,LA.Zadeh 又从控制论的角度提出多目标控制问题;这期间,Charnes、Karlin、Klinger、Polak、Keeney、Geoffrion 等人先后都做出了较有影响的工作。

90 年代后,国内外学者在多目标优化问题上提出一系列算法,1993 年,由 Fonseca 和 Fleming 提出了多目标优化算法(Multiple Objective Genetic Algorithm, MOGA)^[27, 28],其主要思想是每个个体排序的序号由当前种群中支配它的个体的数量决定,采用共享函数和小生境技术实现种群的多样化。1997 年,Tood 和 Sen 采用改进的 MOGA 解决了大规模组合优化问题——集装箱布局(Container Ship Layouts)问题^[29]。进化算法应用到多目标优化问题的求解主要有非劣分类遗传算法(Non-dominated Sorting Genetic Algorithm, NSGA)、小生境 Pareto 遗传算法(Niche Pareto Genetic Algorithm, NPGA)、基于距离的 Pareto 遗传算法(Distance-based Pareto Genetic Algorithm, DPGA)、强度 Pareto 进化算法(Strength Pareto Evolutionary Algorithm, SPEA)等等^[(1)51-67]。

近年来多目标优化研究重点也逐渐向多个方向发展。例如 Takahama 和 Sakai 提出的一种 α 约束法^[30],Runarsson 和 Yao 提出的随机排序法^[31],Pascoletti 等人提出以 Pareto 优化的强度非线性规划方法^[32],Pradyumn 等人提出的有效前端在下一代普通分界交集的方法^[33],Elghazali Talbi 等人提出的平行逼近优化方法^[34],Hamidreza Eskandari 等人提出的一种快速的 Pareto 遗传算法^[35]。

1.3.3 传统多目标优化方法

传统的多目标优化问题求解方法是将各个子目标聚合成一个加权和的单目标函数,系数由决策者决定,然后采用单目标优化算法求解。常见的有加权法、约束法、目标规划法、最小最大法等。本文选取加权法、约束法这两种多目标优化方法来进行简单的介绍^[(1)28-31]。

1. 加权法 加权法是通过目标函数的线形组合将多目标优化问题转换成单目标优化问题。

$$\begin{cases} y_{\min} = f(x) = \sum_{i=1}^N w_i f_i(x) \\ x \in X_f \end{cases} \quad (1-2)$$

w_i 为权值, 不失一般性, 通常权重可以正则化后使得 $w_1 + w_2 + \dots + w_N = 1$, 求解上述不同权重的优化问题则能够输出一组解集。

假设某个可行决策向量 a 在给定权重组合下使 f 取极大值, 但是 a 不是 Pareto 最优解; 另有一个优于 a 的解向量 b , 对于 $i \in N$, 存在 $f_i(b) > f_i(a)$ 和 $f_i(b) \geq f_i(a)$ 。因此 $f(b) > f(a)$, 这与 $f(a)$ 是最大值相矛盾, 如果能使 f 取极大值, 必然是 Pareto 最优解。这种方法存在最大的缺点就是不能在非凸性的均匀曲面上得到所有的 Pareto 最优解。

2. 约束法 约束法是一种不局限于优化 Pareto 最优前端凸部的方法, 即从 k 个目标函数 $f_1(x), f_2(x), \dots, f_n(x)$ 中, 若能够确定一个主要的目标, 例如 $f_k(x)$, 而对于其他的目标函数 $f_1(x), f_2(x), \dots, f_{k-1}(x), f_{k+1}(x), \dots, f_n(x)$ 只要求满足一定的条件即可, 如满足: $f_i(x) \in [a, b], i \neq k, i \in [1, 2, \dots, n]$ 。约束法的这种计算模型如下所示。

$$\begin{aligned} \max y &= f(x) = f_h(x) \\ \text{subject to } &e_i(x) = f_i(x) \geq \varepsilon_i, \quad (1 \leq i \leq N, i \neq h) \\ &x \in X_f \end{aligned} \quad (1-3)$$

ε_i 作为下界可在优化过程中取不同的数值, 以便找到多个 Pareto 最优解。

1.4 本文研究内容及论文组织结构

1.4.1 本文研究内容

本文的研究内容主要集中在对遗传算法与 BP 神经网络融合的研究, 结合遗传算法的全局寻优能力和 BP 神经网络的非线性逼近能力进行分析和改进, 并研究这两种方法协同进行的学习技术。通过对传统遗传算法与 BP 神经网络的基本特征、发展历程以及在多目标优化问题应用上的概述, 从一个全新的角度对遗传算法、遗传算法的改进以及遗传算法的收敛性作了一些探讨, 然后通过理论分析和实验手段验证本文提出的基于遗传算法的 BP 神经网络对多目标优化问题求解方法的可行性与高效性。

目前利用遗传算法求解多目标优化问题的种类较多, 但是仍然有一些理论

问题值得进一步研究和探讨。这些问题包括如何可靠地保持多目标进化种群的多样性、如何解决高维多目标优化问题以及多目标遗传算法的收敛性等。在多目标遗传算法的研究方面,本文提出并实现了一个改进的并行混合非劣分类遗传算法(IMNSGA-II)。IMNSGA-II 算法是对 NSGA-II 算法的改进,通过对 NSGA-II 和其它一些采用精英策略算法的研究发现精英保持也就意味着强调子代中的最优解,但在程序运行到一定代数后种群中的所有解都是精英,随后各代的操作都将在精英解中进行,非精英解无法参与其中,降低了解的多样性。这样由于非精英解不能参与遗传操作,使得全局解的搜索速度减慢并最终导致种群过早收敛到局部 Pareto 解。在改进的 IMNSGA-II 算法中,提出了外部精英存档保持遗传策略,该策略将生成一个外部精英种群,并与原种群共同进化,但进化方法和手段不同,挑选外部种群的方法手段也有差异。在测试过程中,用 C++语言与 Matlab 实现 IMNSGA-II 算法,将 IMNSGA-II 算法和 NSGA-II 算法在五个多目标测试问题(ZDT1, ZDT2, ZDT3, ZDT4, ZDT6)上作了比较,并引入 DEB 提到的两个分别评测多样性及收敛性的性能度量值 γ 和 Δ 进行测试,进而验证改进算法的优越性。

本课题研究了一个基于遗传算法的 BP 神经网络的融合算法,主要思想是:利用遗传算法可以对复杂的、非线性及不可微的函数实现全局搜索,即算法对局部搜索比较有效,因此为了使算法能很快的找到满意解,可以先用遗传算法对初始权值进行优化,在解空间中定位出较好的搜索空间。然后用 BP 神经网络的非线性逼近能力在这些小的解空间中搜索出最优解。

1.4.2 本文的组织

全文共分五章:

第一章是引言部分,介绍人工智能的两个重要分支遗传算法和 BP 神经网络的基本理论、研究现状以及应用特点,并介绍多目标优化问题的基本概念以及求解多目标优化问题的传统方法,最后,对本课题主要的研究内容和本文的组织结构做出简要的介绍。

第二章概括性地介绍遗传算法的原理,然后给出多目标遗传算法一些典型方法,进一步分析多目标遗传算法设计中所要解决的问题:适应度分配、多样性保持、收敛性。

第三章介绍 BP 神经网络基本理论与方法,并通过对 BP 神经网络的限制与不足进行分析,提出一种改进技术。分析 BP 神经网络在多目标优化问题求

解应用。

第四章在非劣分类遗传算法基础上提出一种新的改进精英保持并行混合遗传算法，给出该改进多目标遗传算法算法的设计原理，算法流程及具体的实现方案，同时对改进后的遗传算法与 BP 神经网络的融合，提出基于遗传算法的 BP 神经网络的融合算法。

第五章首先对改进的多目标遗传算法的性能进行测试，给出了评价多目标优化遗传算法性能的度量方法，通过一些标准测试函数上的实验数据对比，验证改进算法的性能。然后将本文提出的基于遗传算法的 BP 神经网络应用于多目标车间作业动态调度实例中，验证模型的可靠性与合理性。

最后，对所做的工作加以总结，并提出了有待进一步研究的问题。

第2章 多目标遗传算法

现实世界的很多问题通常由多个目标组成，这些目标可能是相互矛盾和冲突的。Van Veldhuizen 等人从决策者的角度将多目标遗传算法分成三类：先验法，决策者首先将多目标合成数量成本函数，然后由算法搜索最优解；渐进法，抉择者和算法是互动的，前者为后者提供目标的优先关系，而后者为前者提供新解以产生更好的目标间优先关系；后验法，算法为决策者提供一组候选解供决策者选择^[36]。近年来出现的算法都有一些共同特征，如倾向构造一个外部种群以保留算法获得的非劣解，当外部种群的规模超过规定值时，采取适当措施移出部分解，以维持种群中的均匀性。

多目标遗传算法(MOGA)是近年来发展起来的利用遗传算法处理多目标优化问题的算法。传统的方法很难处理大规模问题，MOGA 不但能处理大规模问题，而且不受问题性质的限制，能够搜索出问题的全局最优解。

本章将介绍遗传算法的原理与多目标遗传算法一些典型方法，随后分析了多目标遗传算法设计中所要解决的问题：适应度分配、多样性保持、收敛性。

2.1 遗传算法的原理

遗传算法(GA)是一种受生物进化启发的学习方法，在本质上是一种不依赖具体问题的直接搜索方法，它是通过变异和重组当前已知的最好假设来生成后续的假设。遗传算法在模式识别、神经网络、图像处理、机器学习、工业优化控制、自适应控制、生物科学、社会科学等方面都得到应用。遗传算法研究的问题是搜索候选假设空间并确定最佳的假设，“最佳假设”被定义为是使适应度最优的假设，适应度是为当前问题预先定义的数字度量。遗传算法的普及和发展得益于以下因素：

1. 在生物系统中，进化被认为是一种成功的自适应方法，且具有很好的健壮性。
2. GA 搜索的假设空间中，假设的各个部分相互作用，每一部分对总的假设适应度的影响难以建模。
3. GA 易于并行化，且可降低由于使用超强计算机硬件所带来的昂贵费用。

遗传算法采用简单的参数编码技术来表示各种复杂的结构，并通过对一组

编码表示进行简单的遗传操作和优胜劣汰的自然选择来指导学习和确定搜索的方向。遗传算法的操作对象是一群二进制串或十进制串(称为染色体或个体),即群体。这里每一个个体都对应问题的一个解。从初始群体出发,采用基于适应度值比例的选择策略在当前群体中选择个体,使用变异和交叉来产生下一代群体。如此模仿生命的进化一代代演化下去,直到满足期望的终止条件为止。

遗传算法包括三个基本操作:选择、交叉和变异^[37]。

1.选择 在生物的遗传和自然进化过程中,对生存环境适应程度较高的物种将有更多的机会遗传到下一代;而对生存环境适应程度较低的物种遗传到下一代的机会就相对较少。选择的目的是为了从当前群体中选出优良的个体,使它们有机会作为父代产生后代个体。判断个体优良与否的准则就是各自的适应度值。遗传算法使用选择算子(或称复制算子)来对种群中的个体进行优胜劣汰操作:适应度值较高的个体被遗传到下一代种群中的概率较大;适应度值较低的个体被遗传到下一代种群中的概率较小。

作为一种算子,选择操作在遗传算法中有多种方法,其中最简单的一种方法就是采用和适应度值成比例的概率方法来进行选择。具体地说,就是首先计算群体中所有个体总和 $\sum f_i$,再计算每个个体的适应度所占的比例 $f_i / \sum f_i$,并以此作为相应的选择概率。

2.交叉或基因重组 在生物的自然进化过程中,两个同源染色体通过交配而重组,形成新体,从而产生出新的个体或物种。交配重组是生物遗传和进化过程中的主要环节。模仿这个环节,在遗传算法中也使用交叉算子来产生新的个体。对交叉算子的实现过程如图 2-1。

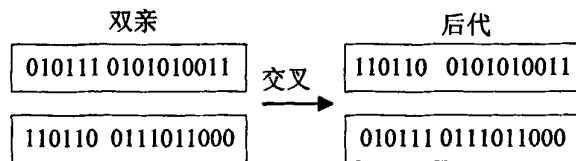


图 2-1 遗传算法中交叉算子

Fig. 2-1 Cross operator in genetic algorithms

3.变异 在生物的遗传和自然进化过程中,其细胞分裂复制环节有可能会因为某些偶然因素的影响而产生一些复制差错,这样就会导致生物的某些基因发生某种变异,从而产生出新的染色体,表现出新的生物性状。在遗传算法中变异是一个重要的操作,该算子的实现方法如下:对于群体中的某个个体,即基因链码,随机选取某一位,将其翻转(0 改为 1, 1 改成 0)。在遗传算法中

使用变异算子主要目的是改善遗传算法的局部搜索能力与维持种群的多样性，防止出现早熟现象。变异算子的设计包括如何确定变异点的位置和如何进行基因值替换。

遗传算法的实现主要有六个因素：参数的编码、初始种群的设定、适应度函数的设计、遗传操作、算法控制参数的设定、约束条件的处理，其构造步骤如图 2-2 所示。

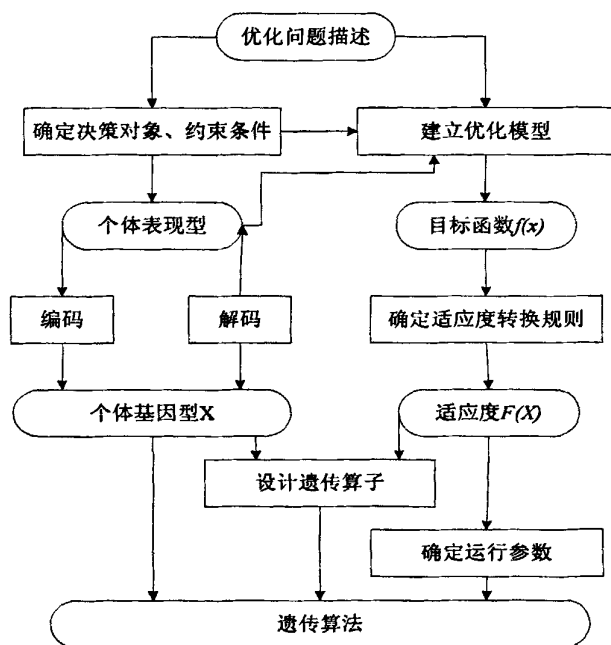


图 2-2 遗传算法的构造过程

Fig. 2-2 Tectonic process of genetic algorithms

遗传算法的伪代码如下：

```

Status SGA(void){
    Initialize(P[0]); //随机初始化群体 P[0]={x1,x2,...,xM}
    UINT r=0, t=0, i=0;
    do{
        for(i=0 ; i < M ; ++i){EvaluateFitness(P[t], i);
            /*对当前群体 P[t]中的每个个体 xi 计算适宜度 F(xi),
            适宜度表示该个体性能好坏*/}
    }
    
```

```

for(i=0 ; i<M ; ++i){SelectOperation(P[t], &r, i);
/*应用选择算子产生中间代群体 P[t]*/}
/*对 P[t]应用其他的算子，产生新一代群体 x(t+1)，这些算子
    的目的在于扩展有限个体的覆盖面，体现全局搜索的思想*/
for(i=0;i<M/2 ;++i){CrossoverOperation(P[t], i);}
for(i=0 ;i<M ;++i){MutationOperation( P[t], i);}
for(i=0;i<M;++i){CopyOperate(P[t+1], P[t],i);}
++ t;
}while(t<=T);
return StatusType;
}

```

2.2 主要的多目标遗传算法

求解多目标问题的遗传算法的基本结构与求解单目标问题的遗传算法的基本结构大致相似。但另一方面，在利用遗传算法进行多目标问题求解时，需要考虑评价 Pareto 最优解以及如何设计适合于多目标问题的选择、交叉和变异算子等问题。下面将选取向量评估遗传算法(VEGA)、多目标遗传算法(MOGA)、基于距离的 Pareto 遗传算法(DPGA)、小生境 Pareto 遗传算法(NPGA)、非劣分类遗传算法(NSGA)这些经典的多目标遗传算法的基本原理和各自的优缺点做简单的介绍。

2.2.1 向量评估遗传算法

向量评估遗传算法(Vector Evaluated Genetic Algorithms, VEGA)是由 Schaffer 于 1985 年提出的第一个用于寻找一组非劣解的多目标进化算法。VEGA 包含了原始简单遗传算法(SGA)的一些操作算子，对选择机制进行了改进以便处理多个目标函数。

VEGA 的基本思想：如果要处理多个目标(假设为 k 个)，那么在每一代将种群随机地均分成 k 个大小相等的子种群。根据不同的目标函数对每个子种群分配一个适应值。这样，每个目标函数用于评估一个子种群的所有个体，每个子种群的规模为 N/k ，其中 N 为整个群体规模。

VEGA 的主要优点是思想简单，易于实现。只须对简单遗传算法作一个较

小的改变就能将它转换成多目标进化算法并且不会产生任何附加的计算复杂性。在这一点上 VEGA 明显优于所有其它的多目标进化算法。

VEGA 本身具有缺陷, 因为通过每个子种群来产生新个体是针对其中一个目标函数进行评估的, 这个子种群的适应度评估情况不能应用到其他目标函数中。如果所有的遗传算法参数选择适当, VEGA 搜索到的解可能非常接近 Pareto 最优域, 但它最终注定趋向于单个目标函数最优的个体, 因此失去了 Pareto 最优域的中间解。VEGA 运行到相当多的代之后, 它没有维持沿着 Pareto 最优域的解的多样性分布而收敛到一个单个目标函数最优的个体附近。虽然使用 VEGA 可能产生不了前端上均匀分布的解点和大量的解结果, 但是能够找到 Pareto 前端曲线或曲面上的始末端点, 有时成为 VEGA 的长处, 所以 VEGA 的这一特性现在仍然具有价值和应用场合, 或者在实际问题求解中作为其他多目标遗传算法求解的一个补充。

2.2.2 多目标遗传算法

Carlos Fonseca 和 Peter Flemming 在 1993 年提出了对遗传算法的种群进行非劣分级的多目标遗传算法(Multiple Objective Genetic Algorithm, MOGA), 通过采用通用的 GA 方法框架来有效地解决多目标问题。这里需要注意的是, 这里的 MOGA 名称不能从字面上来理解其覆盖的内容太大, 容易使用误解为通用多目标遗传算法, 好像是采用遗传算法来求解多目标优化问题的统称, 但是在最初论文发表时是这样提出来的, 所以后人只能沿用这种称呼。

MOGA 对种群中每一个个体的排序数是基于 Pareto 最优的概念对当前种群中优于该个体解的其他个体数目进行统计, 并采用一种基于排序数的适应度赋值方式。另外, 该算法采用自适应的小生境技术与受限杂交技术来提高种群多样性, 防止解群过早收敛。MOGA 的独特之处是种群中每个解的适应值分配方式。首先检查种群中每个解的占优性。对于一个解 i , 它的等级 r_i 等于 1 加上优于它的解的个数 n_i 。由于种群中没有优于非劣解的解, 所以非劣解的等级等于 1。很显然, 种群中至少有一个解的等级等于 1, 任何一个解的等级不可能大于 N (N 是种群大小)。一旦算出所有解的等级, 对每个解根据它的等级分配一个初始适应值。对每个个体分配一个初始的适应值。通常选择使分配的适应值在 $1 \sim N$ 之间的对应函数。此后, 一次考虑一个等级的个体并求它们初始适应值的平均值。这个平均适应值被称为每个等级每个个体的指定适应值。这样, 每个等级的分配的总的初始适应值等于总的指定适应值。而且, 这个对应

和平均的过程确保了较好等级的个体有一个较高的指定适应值。这样，强调了种群中的非劣解。

另外，MOGA 采用适应度共享方式，称之为拥挤操作算子和小生境操作算子，其中的重要参数是 σ_{share} 。 σ_{share} 是指小生境数目，必须小心设置，因为它对进化结果比较敏感。Fonseca 和 Flemming 采用小生境方法是在表现性空间实施的，这里使用了 $\alpha=1$ 的共享函数，距离尺度的计算利用的是目标函数值，而不是参数值。因此，在一个等级内的任意两个个体 i 和 j 之间的标准化距离的计算如下：

$$d_{ij} = \sqrt{\sum_{s=1}^k \left(\frac{f_s^{(i)} - f_s^{(j)}}{f_s^{\max} - f_s^{\min}} \right)^2} \quad (2-1)$$

f_s^{\max} 和 f_s^{\min} 是第 k 个目标函数的最大值与最小值。对于解 i ，计算与解 i 有同样等级 r_i 每一个解 d_{ij} 使用方程：

$$Sh(d_{ij}) = \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma_{share}} \right)^\alpha, & d_{ij} \leq \sigma_{share} \\ 0, & Other \end{cases} \quad (2-2)$$

此后，小生境数就是共享函数值之总和。

MOGA 方法的优点是适应值分配方案简单。由于在目标空间使用了小生境策略，MOGA 可以很容易地应用到其它优化问题，例如组合优化问题。如果在目标空间需要一组分布较好的 Pareto 最优解，MOGA 是一个合适的选择。

MOGA 的主要不足在于如果小生境数目信息是基于目标函数的，那么两个具有相同目标函数向量的不同个体无法在同一代种群中存在，这显然是人们不希望看到你，因为这样的两个解方案有可能恰恰就是决策者想要得到的结果。

2.2.3 基于距离的 Pareto 遗传算法

Osyczka 和 Kundu 在 1995 年提出了一种使用最优保存策略的遗传算法，称基于距离的 Pareto 遗传算法 (Distance-based Pareto Genetic Algorithm, DPGA)。该算法使用一种适应值分配措施来试图强调向 Pareto 最优域发展和保持最优群中非劣解的多样性。该算法保持两个种群：一个使用遗传算子的标准

遗传算种群 P_n 和一个包含迄今为止发现的所有非劣解的最优种群 E_n 。

DPGA 的基本思想：首先建立种群大小为 N 的随机初始种群 P_0 ，分配给种群的第一个个体一个正的随机适应值 F_1 并将该个体自动地加入最优组 E_0 。进而对种群中的每个个体，根据它们与最优组 $E_n = \{e^{(m)}, m=1, 2, \dots, M\}$ 中个体的最小距离分配一个适应值。这里 M 是最优组的个体数。个体 x 与最优组中第 m 个个体的距离计算如下：

$$d^{(m)}(x) = \sqrt{\sum_{i=1}^k \frac{e_i^{(m)} - f_i^{(x)}}{e_i^{(m)}}} \quad (2-3)$$

对于个体 x ，寻找对于所有 $m=1, 2, \dots, M$ 的最小 $d^{(m)}(x)$ ，该最小距离的索引为 m^* ，记 $F(e^{(m^*)})$ 为 F^* 。此后，如果个体 x 相对于现行的最优组是非劣解，那么接受它到最优组并计算它的适应值为与它距离最小的最优个体的适应值加上该最小距离，即 $F(x) = F^* + \min_{m=1}^M d^{(m)}(x)$ 。如果有劣于个体 x 的最优个体，那么从最优组中删除所有这些个体。另一方面，如果个体 x 劣于任一最优解，那么拒绝它到最优组并计算它的适应值为：

$$F(x) = \max \left[0, (F^* - \min_{m=1}^M d^{(m)}(x)) \right] \quad (2-4)$$

这样，当评估种群的个体以计算它们的适应值时，最优组被不断地更新。当评估完所有 N 个个体时，计算现行的最优个体中的最大适应值 F_{\max} ，并且所有现行的最优个体被分配一个等于 F_{\max} 的适应值。最后对这一代种群使用选择、交叉和变异算子建立新一代的种群。

DPGA 的优点是对于按某些顺序排列的个体的适应值评估，不需要任何明确的小生境方法就可以达到朝 Pareto 最优域进展和保持非劣解的多样性两个目标。DPGA 的缺点是没有限制最优组的大小且允许增加到任意大小，因此，该算法的计算复杂性将随着代数的增加而增加。然而，可以限制最优组的大小克服该困难。只有当最优组的大小没有超过用户具体指定的限制时，才允许将一个个体放入最优组。

2.2.4 小生境 Pareto 遗传算法

小生境 Pareto 遗传算法(Nicked Pareto Genetic Algorithm, NPGA)是由 Horn、Nafpliotis 和 Goldberg 提出的，它与 SGA 的主要区别在于选择机制和适应度赋值方式不同。NPGA 采用 Pareto 优胜关系进行锦标赛选择，并使用适应

度共享机制。NPGA 的锦标赛选择机制与普通基于两个个体之间的直接比较方案不同的是, NPGA 还额外地从种群中选取一定数量的其他个体参与非劣最优解的比较。NPGA 锦标赛选择过程: 1. 从亲代种群 P 中挑选一个种群大小为 t_{dom} ($t_{dom} \ll N$) 的子种群 T_y ; 2. 分别计算 T_y 中优于 i, j 的个体数 a_i, a_j ; 3. 如果 $a_i=0$ 且 $a_j>0$, 那么 i 获胜, 该过程结束; 4. 如果 $a_i>0$ 且 $a_j=0$, 那么 j 获胜, 该过程结束; 5. 如果子代种群 $|Q|<2$ 那么 i 和 j 各有 0.5 的概率获胜而被选择, 该过程结束。否则将 i 和 j 独立地放入当前的子代种群 Q 中并计算它们的总数 hc_i 和 hc_j 。计算 hc_i 与个体 i 的距离 d_{is} 在 σ_{share} (σ_{share} 是设定的小生境参数) 以内的子代个数 k 。个体 i 和 s 标准化距离计算公式如下:

$$d_{is} = \sqrt{\sum_{m=1}^k \left(\frac{f_m^{(i)} - f_m^{(s)}}{f_m^{\max} - f_m^{\min}} \right)^2} \quad (2-5)$$

f_m^{\max} 和 f_m^{\min} 是第 k 个目标函数的最大值与最小值。同理可计算出 d_{js} 。

6. 如果 $hc_i \leq hc_j$, 那么个体 i 获胜; 否则个体 j 获胜。

NPGA 的主要优点是不必使用明确的适应值分配, 而主要缺点是需要固定两个重要的参数 σ_{share} 和 t_{dom} 。

2.2.5 非劣分类遗传算法

Sriivas 和 Deb 在 1994 年提出非劣分类遗传算法(Nondominated Sorting Genetic Algorithm, NSGA), 该算法与 SGA 的过程类似, 但选择操作算子不同。NSGA 通过用偏好非劣解的适应值分配方案和保持每个非劣层的个体的多样性的共享策略实现了多目标优化算法的双重目标。

群体在操作之前, 根据每个个体的非劣和支配情况进行分类排序。在对个体排序之后, 找到的非劣解点被赋予一个很大的假设适应度值, 它们作为第一批非劣前端。由于它们都是 Pareto 最优的, 因此赋予了相同的适应度值, 保证每个个体都具有相同的复制机会。

NSGA 的主要优点是根据非劣层分配适应值。由于有计划地强调了较优非劣组, NSGA 以组的方式朝向 Pareto 最优域进展。而且, 当使用 NSGA 时, 在参数空间实行共享尤其使得互异的解产生。如果需要, 共享也可以在目标空间实行。

NSGA 的缺点是计算效率较低, 计算复杂度为 $O(MN^3)$ (M 为目标数量, N 为种群大小), 未采用精英保留策略(Elitism), 共享参数 σ 需要预先确定。

NSGA 执行对参数 σ_{share} 敏感。

Deb.et 等学者在 2000 年对原始的 NSGA 进行改进, 提出改进型非劣分类遗传算法(Nondominated Sorting Genetic Algorithm II, NSGA-II)。对第一版 NSGA 的最大改进是根据产生的各种非劣前端, 采用更好的记账策略, 从而减少算法运行的整体时间^[38]。

2.3 多目标遗传算法的关键理论

基于 Pareto 最优概念的多目标遗传算法是当前进化计算领域的研究和应用热点。多目标遗传算法的研究目标主要是使算法种群快速收敛, 并且均匀分布于问题的非劣最优域。这里在 MOGA 设计时要考虑三个目标^{[1]77-101}:

1. 为了避免早熟收敛, 使进化过程朝 Pareto 集定向搜索, 如何实施适应度分配与选择; 改善算法收敛性能。
2. 保持非劣解集多样性, 如何保持种群的多样性。
3. 如何提高多目标演化算法的求解效率。

2.3.1 适应度分配

单目标优化中的目标函数常与适应度函数是一样的, 但 MOP 中的适应度赋值和选择必须考虑到几个子目标。目前最常见的适应度赋值和选择方法包括: 基于聚合的策略、基于准则的策略和基于 Pareto 优胜关系的策略。

基于聚合的策略建立在传统方法的基础上, 组合多个目标成为一个多目标函数, 以此来产生均衡曲面。另外, 在优化过程中系统地调整该函数中的参数, 以便找出一组非劣解而非单个的均衡解。

基于准则的策略在选择阶段变换所选的优化目标。每当个体选中后进行复制, 则根据不同的目标来决定是否被复制至配对池。

基于 Pareto 优胜关系计算个体的适应度, 由 Goldberg 提出的多目标竞争选择机制。后人大多数在此基础上进行改进, 如将群体划分成几个有序的子群体。基于 Pareto 优胜关系思想为: 对于群体中的每个个体 x_i , 都与群体中随机选取的 r 个个体分别进行比较, 统计这 r 个个体中所有 p 个优化目标函数值都优于给定的个体 x_i 的对应的目标函数值的个数 w_i 。这个优越的个数 w_i 就作为这个个体 x_i 的多目标优化进化计算的适应值 F_i 。多目标进化优化的过程是使整个进化群体的 F_i 都趋于极小, 也即使整个进化群体不断地向 Pareto 边界趋近的过程。

2.3.2 多样性保持

群体多样性的维护问题对多目标优化遗传算法的性能会产生重要的影响，这个问题解决的不好，很易导致算法早熟，从而生成极端最优解，不能获得分布均匀的 Pareto 最优解。目前常用的解决方法有共享函数法(如 MOGA、NSGA、NPGA)和聚类分析技术(如 SPEA)等。遗传算法由于遗传算子的随机误差，在使用有限群体时易收敛于单个解。但多目标优化是要找到整个 Pareto 最优前端，而不是单个非劣解。在自然界中，当某一种群中的个体大量繁殖生存时，为争夺有限的生存资源，群体中个体之间的竞争压力必然加剧，因此个体的寿命和出生率低。基于这种机制，Holland 提出使用“排挤”算子，即新生成的子代替或排挤相似的父代个体，该方法可有效提高群体的多样性。DeJong 使用排挤算子做实验，并度量两个基因型个体的相识性，即在待比较的两个个体中统计相等染色体的位数。DeJong 在他的模拟模型中使用了两个参数：代沟(G)和排挤因子(CF)。G 表示群体中允许被复制的个体所占百分率，CF 规定了候选解被后代取代的个体数目。在采用覆盖群体模式的情况下，排挤方法的过程可描述：

1. 设定参数 CF；
2. 从群体中随机挑选 CF 个体组成个体集；
3. 从该个体集内淘汰一个个体，该个体与新个体的海明距离最短。

2.3.3 收敛性

多目标与单目标优化问题解的本质区别是，前者通常是一组或多组连续的集合，而后者是单个解或一组不连续的解结果。在多目标遗传算法中，当进化代数 t 趋于无穷时产生的 Pareto 前端近似 $A^{(t)}$ 的序列收敛于真实的 Pareto 前端 Y^* ，则认为该多目标遗传算法是全局收敛的。Rudolph、Hanne 和 Agapie 等人相继提出了一些选择策略用以排除遗传算法的退化现象，保证收敛^[39]。基本思想是当解受控于其他个体时，则丢弃该解，保证接收德尔夫个体具有充分的单调性。虽然目前大多数多目标遗传算法都特别考虑了优胜准则密度信息，但可以证明如果这些算法连续采取这种选择方式，将会导致退化现象。

由于群体规模的限制，要找到任意目标空间 Y 上的 Pareto 前端通常不具有可行性。针对实际应用的需要，目前有学者提出 ε -近似 Pareto 集的概念来表征向量 r^* 及其分量，其概念示意如图 2-3 所示^[(11)101-102]。

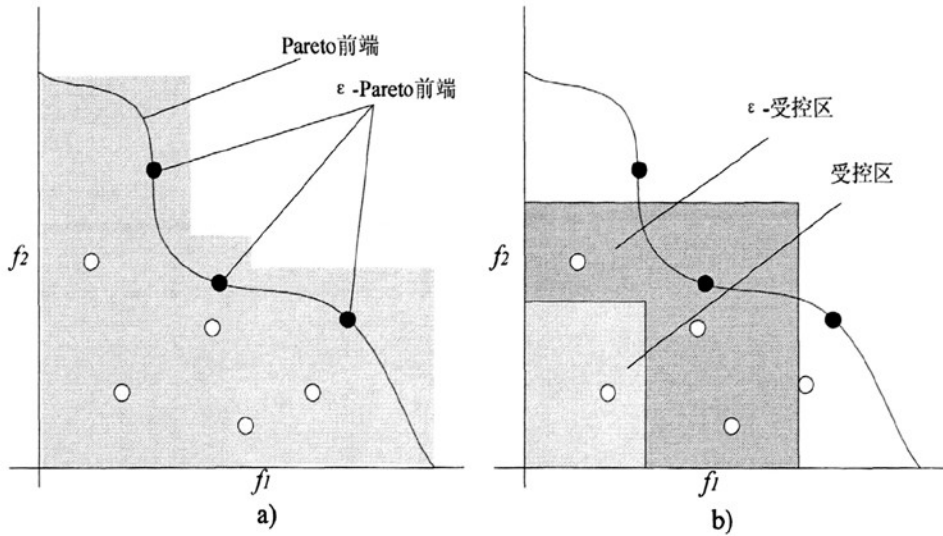


图 2-3 ϵ -优胜和 ϵ -Pareto 前端的概念示意图

Fig. 2-3 Conceptual diagram of ϵ -dominance and ϵ -Pareto front

2.4 本章小结

本章通过引用基本遗传算法的原理，着重介绍了目前已出现的一些主要多目标遗传算法，并阐述了它们的优缺点和求解性能。介绍了 VEGA、MOGA、DPGA、NPGA、NSGA 这几种基于遗传算法的多目标优化算法，分别对其适应度的分配机制和各自的优缺点进行了详细的阐述。最后分析了多目标遗传算法设计中所要解决的问题，通过这些研究也为下一步的研究工作做技术准备。

第3章 BP 神经网络的理论与应用

神经网络作为人工智能的一个重要分支，备受关注。神经网络系统理论是以人脑的智能功能为研究对象，研究人类大脑的信息处理能力与方法，特别是研究人类大脑的智能信息处理能力相关的信息处理理论方法，建立智能计算的理论与方法，为智能计算机的研究开发奠定理论基础。所谓人工神经网络是由人工神经元按照一定的拓扑结构互连而成的网络，用来模拟人的大脑通过学习获取知识并解决问题的能力。

误差反向传播算法的多层前向神经网络(BP 网络)，具有结构严谨、工作状态稳定、可操作性强等特点，采用 Sigmoid 型可微函数作为传递函数，可以实现输入和输出间的任意非线性映射，广泛地应用于函数逼近、模式识别、数据压缩等领域。本章将介绍 BP 神经网络的相关理论与方法，并给出多目标拟神经网络优化方法。

3.1 人工神经网络

3.1.1 人工神经元模型

作为人工神经网络的基本单元——人工神经元，如图 3-1 所示。它有三个基本要素：

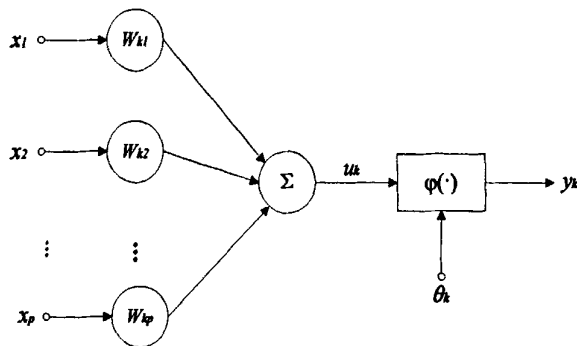


图 3-1 基本神经元模型

Fig. 3-1 Basic neuron model

1. 一组连接，连接强度由各连接上的权值表示，权值为正表示激活，为负表示抑制。

2. 一个求和单元，用于求取个输入信号的加权和(线形组合)。

3. 一个非线性激活函数，起非线性映射作用并将神经元输出幅度限制在一定范围内。

以上作用可分别以数学式表达出来：

$$u_k = \sum_{j=1}^p w_{kj} x_j, \quad v_k = net_k = u_k - \theta_k, \quad y_k = \varphi(v_k) \quad (3-1)$$

式中： x_1, x_2, \dots, x_p 为输入信号， $w_{k1}, w_{k2}, \dots, w_{kp}$ 为神经元 k 的权值， u_k 为线形组合结果， θ_k 为阈值， $\varphi(\cdot)$ 为激活函数， y_k 为神经元 k 的输出。

这里激活函数 $\varphi(\cdot)$ 可以有以下几种形式。

1. 阈值函数

$$\varphi(v) = \begin{cases} 1, & v \geq 0 \\ 0, & v < 0 \end{cases} \quad (3-2)$$

即阶梯函数。此时相应的输出

$$y_k = \begin{cases} 1, & v_k \geq 0 \\ 0, & v_k < 0 \end{cases} \quad (3-3)$$

其中 $v_k = \sum_{j=1}^p w_{kj} x_j - \theta_k$ ，常称此种神经元为 M-P 模型。

2. 分段线形函数

$$\varphi(v) = \begin{cases} 1, & v \geq 1 \\ \frac{1}{2}(1+v), & -1 < v < 1 \\ 0, & v \leq -1 \end{cases} \quad (3-4)$$

它类似于一个放大系数为 1 的非线性放大器，当工作于线性区时它是一个线性组合器，放大系数趋于无穷大时变成一个阈值单元。

3. Sigmoid 函数

最常用的函数形式为

$$\varphi(v) = \frac{1}{1 + e^{-\alpha v}} \quad (3-5)$$

参数 $\alpha > 0$ 可控制其斜率。另一种常用的是双曲正切函数

$$\varphi(v) = \tanh\left(\frac{v}{2}\right) = \frac{1 - e^{-v}}{1 + e^{-v}} \quad (3-6)$$

这类函数具有平滑和渐进性，并保持单调性。

3.1.2 神经网络模型

目前，神经网络的模型已有几十种，根据神经元之间连接的方式不同，神经网络可分为两大类：没有反馈的前向网络和具有反馈的相互结合型网络。前向网络由输入层、中间层(隐层)和输出层组成，如图 3-2 所示。中间层可以有若干层，每一层的神经元只接受来自前一层神经元的输出。反馈型网络中的任意两个神经元间都可能连接，包括神经元到自身的反馈，如图 3-3。

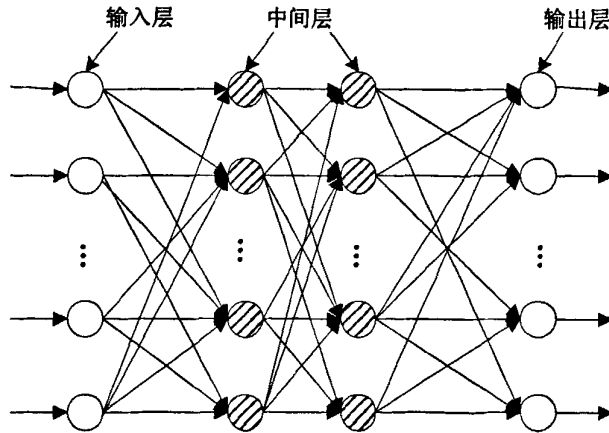


图 3-2 具有两个中间层的前馈神经网络

Fig. 3-2 Two middle layer feed forward neural network

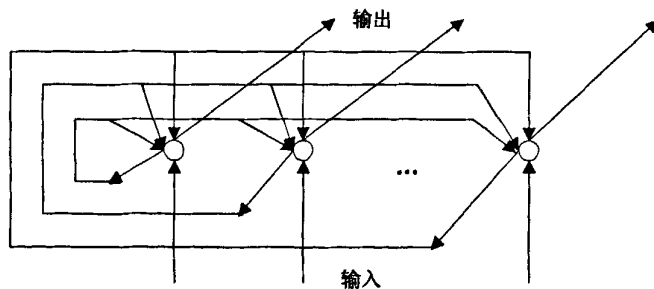


图 3-3 反馈型神经网络

Fig. 3-3 Feedback neural network

从作用效果看，前馈神经网络主要是函数映射，可用于模式识别和函数逼近。反馈神经网络按对能量函数的极小点的利用分类有两种：第一类是能量函数的所有极小点都起作用，这一类主要用作各种联想存储器；第二类只利

用全局极小点，它主要用于求解最优化问题。

3.2 BP 神经网络

3.2.1 BP 神经元模型

BP 网络是前向反馈网络的一种，也是当前应用最为广泛的一种网络。BP 网络结构如图 3-4。BP 的神经元有 n 个输入，每个输入都是通过一个适当的权值 $w_i (i=1, 2, \dots, n)$ 连接到神经元节点上。输出函数 $y=f(a, \theta)$ ， a 即为全部输入的加权求和， $a=\sum w_i x_i$ ， a 作为激活函数 f 的输入，激活函数的另外一个输入是神经元的阈值 θ 。

BP 网络中隐层激活函数通常采用 S 型对数函数；也可以采用 S 型正切函数；还可以采用线形函数。神经元模型的输出向量可表示为：

$$y = f(W \cdot X + \theta) = f\left(\sum_{i=1}^n w_i x_i + \theta\right) \quad (3-7)$$

这里权值和输入矩阵为 $W = [w_1, w_2, \dots, w_n]$ ， $X = [x_1, x_2, \dots, x_n]$ 。

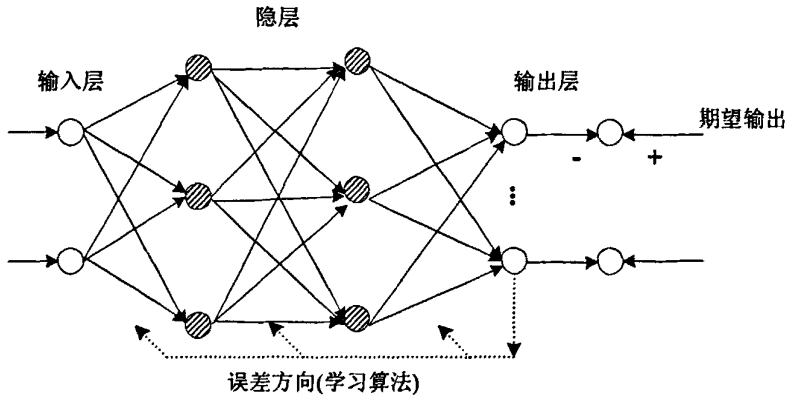


图 3-4 BP 网络模型结构

Fig. 3-4 Structure of BP network model

3.2.2 BP 网络的数学模型

输入层节点 i 的输出为 $x_i' = f'(x_i)$ ，其中 x_i 为模型的输入。为了使 BP 模型隐含层的输入规范化，函数 f' 为一规整函数，即：

$$f' = \frac{x - x^{\min}}{x^{\max} - x^{\min}} \quad (3-8)$$

其中, x^{ll} 为节点输入的下限, x^{ul} 为节点输入的上限。隐含层节点 j 的输出为 $O_j = f(I_j)$, 这里 $I_j = \sum w_{ij} x'_i - \theta_j$, I_j 称为某层节点 j 的输入的加权, 且 w_{ij} 为输入层节点 i 到隐含层节点 j 之间的连接强度(或称权值), θ_j 为隐含层节点 j 的阈值, 而在 BP 网络中 f 函数通常取 S 型函数。这样, 拟合精度较高, 即:

$$O_j = \frac{1}{1 + e^{-\sum w_{ij} x'_i - \theta_j}} \quad (3-9)$$

输出层节点 k 的输出为:

$$y'_k = \sum W_{jk} O_j - \theta_k \quad (3-10)$$

其中, W_{jk} 为隐含层节点 j 到输出层节点 k 之间的权值, θ_k 为输出层节点的阈值。模型的实际输出值 y_k 与输出层输出值 y'_k 之间的关系为:

$$y_k = (y^{\text{ul}} - y^{\text{ll}}) y'_k + y^{\text{ll}} \quad (3-11)$$

其中, y^{ll} 为系统实际输出的下限, y^{ul} 为系统实际输出的上限。

3.2.3 基于梯度下降学习算法

考虑一般多层前馈神经网络模型, 设共有 L 层。第 l 层神经元个数为 N_l , 输入为 $X^{(l)}$, 输出为 $Y^{(l)}$ 。第 l 层与第 $l+1$ 层的连接权为 $W^{(l)}$ 。其中输入层到隐藏层的激活函数采用 Sigmoid 型, 隐藏层到输出层的激活函数采用线性函数。设 $x_0^{(l)}=1$, 则各层的阈值可由 $w_{0j}^{(l)}$ 表示。网络的输出与输入之间关系可表示为

$$Y^{(L)} = f(X^{(L)}) = f(W^{(L-1)} Y^{(L-1)}) \quad (3-12)$$

若仅考虑 $L=3$ 的情况, 设用 P 个样本训练网络, 并设第 p 个样本的输出为 Y_p^* , 则网络误差函数可以定义为

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{i=1}^{N_3} (y_{pi}^{(3)} - y_{pi}^*)^2 \quad (3-13)$$

这里设 $W = (w_{01}^{(1)}, w_{02}^{(1)}, \dots, w_{N_1 N_2}^{(1)}, w_{01}^{(2)}, w_{02}^{(2)}, \dots, w_{N_2 N_3}^{(2)})$, 即 W 为 $(N_1+1)N_2 + (N_2+1)N_3$ 维向量。极小化 $E(W)$ 的 BP 算法为

$$W(t+1) = W(t) - \alpha \nabla E(W(t)) \quad (3-14)$$

式中, $\alpha (0 < \alpha < 1)$ 为学习步长; t 表示迭代次数。

3.2.4 BP 神经网络算法

BP 算法的基本思想是最小二乘学习算法。它采用梯度搜索技术以期望使网络的实际输出值与期望输出值的误差均方值为最小，这是一种可以避免统计平均的递推算法。虽然这种算法不能界定保证收敛的步幅范围，但它能给出每一步调整的正确方向。它的学习过程是一种误差边向后传播边修正加权系数的过程，包括了正向和反向传播两个阶段。在正向传播过程中，输入信息从输入层经隐含层逐层处理并传向输出层，每一层神经元的状态只影响下一层神经元的状态。如果在输出层不能得到期望输出，则转入反向传播，将误差信号沿原来的连接通路返回，通过修改各层神经元的加权系数，使误差最小。

BP 神经网络的算法如下^[40]：

1. 输入 N 个样本集合 $\{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=1, 2, \dots, N\}$ 。

2. 建立 BP 网络结构。确定网络层数 $L > 2$ 和各层节点数，由样本集合输入向量 x_k 的长度 m 确定网络输入节点数为 m ；由样本集合输出向量 y_k 的长度为 n 确定网络输出节点数为 n ；第 l 层的节点数为 $m^{(l)}$ 。定义各层间连接权矩阵，第 l 层连接第 $l+1$ 的连接权矩阵为 $W^{(l)} = [W_{ij}^{(l)}] m^{(l)} \times m^{(l+1)} (l=1, 2, \dots, L-1)$ ，初始化各连接权矩阵的元数值。

3. 输入允许误差 ε 和学习率 η ，初始化迭代计算次数 $t=1$ ，样本集合序号 $k=1$ 。

4. 取第 k 个样本集合 $\{(x_k, y_k) | x_k = x_{1k}, x_{2k}, \dots, x_{mk}, y_k = y_{1k}, y_{2k}, \dots, y_{nk}\}$ 。

5. 由 x_k 进行正向传播计算。计算输入层各节点的输出为 $O_{jk}^{(1)} = f(x_{jk})$ ， $j=1, 2, \dots, n$ ；逐层计算各层的各节点输入和输出为

$$I_{jk}^{(l)} = \sum_{i=1}^{m^{(l-1)}} w_{ij}^{(l-1)} O_{ik}^{(l-1)} \quad l=2, 3, \dots, L; \quad j=1, 2, \dots, m^{(l)} \quad (3-15)$$

$$O_{jk}^{(l)} = f(I_{jk}^{(l)})$$

6. 计算输出层的各输出节点的误差为

$$y_{jk} = O_{jk}^{(L)}$$

$$E_{jk} = \frac{1}{2} (y_{jk} - \hat{y}_{jk})^2 \quad j=1, 2, \dots, n \quad (3-16)$$

7. 对 N 个样本集合的任一离散时间序列 k 有 $E_{jk} \leq \varepsilon, j=1, 2, \dots, n$ ，则学

习过程结束；否则，进行误差反向修改各连接权矩阵。

8. 修改第 $L-1$ 层隐含层至输出层 L 的连接权矩阵为

$$\begin{aligned}\delta_{jk}^{(L)} &= -(y_{jk} - \hat{y}_{jk}) f'(I_{jk}^{(L)}) \\ \Delta w_{ij}^{(L-1)}(t) &= \eta \delta_{jk}^{(L)} O_{ik}^{(L)} \\ w_{ij}^{(L-1)}(t+1) &= w_{ij}^{(L-1)}(t) + \Delta w_{ij}^{(L-1)}(t) \\ j &= 1, 2, \dots, n; i = 1, 2, \dots, m^{(L-1)}\end{aligned}\quad (3-17)$$

反向逐层修改联接各隐含层连接权矩阵：

$$\begin{aligned}\delta_{jk}^{(l)} &= f'(I_{jk}^{(l)}) \sum_{q=1}^{m^{(l+1)}} \delta_{qk}^{(l+1)} w_{jq}^{(l)} \\ \Delta w_{ij}^{(l-1)}(t) &= -\eta \delta_{jk}^{(l)} O_{ik}^{(l-1)} \\ w_{ij}^{(l-1)}(t+1) &= w_{ij}^{(l-1)}(t) + \Delta w_{ij}^{(l-1)}(t) \\ l &= L-1, \dots, 2, 1; j = 1, 2, \dots, m^{(l)}; i = 1, 2, \dots, n^{(l-1)}\end{aligned}\quad (3-18)$$

9. $k=(k+1)\%N$, $t=t+1$, 转步骤 4. ^[41]。

3.3 BP 算法的限制与改进

多层前馈神经网络在众多领域中得到了广泛应用。人们对非线性神经元组成的前馈网络的 BP 学习算法已做了广泛而深入的研究，然而，回顾第一章绪论知道，由于这种算法采用的是误差函数的梯度下降优化方法，因而不可避免存在局部极小问题、收敛对初值敏感问题及收敛速度慢问题。BP 算法在实际应用过程中存在：

1. 误差下降缓慢，调整时间长，迭代次数多，影响收敛速度。
2. 因存在多个极小点而使训练会因陷入某个局部极小点而不能自拔，使训练难以收敛于给定的误差。

在提高收敛速度方面，已有许多学者做出了研究并提出很多方法^[28, 38, 42]，主要措施有：

1. 附加冲量项。该方法是在每个加权调节上加上一项正比例于前次加权变化量的值。这就要要求每次调节完成后，要把该调节量记住，以便在下面的加权调节完成后，要把该调节量记住，以便在下面的加权调节中使用。

2. 改进误差函数。在式(3-16)中定义了误差函数，但这不是唯一的选择，可以选用别的函数 $f(O^*, O)^2$ 代替 $f(O^*, O)^2$ ，只要 f 函数在 $O^* = O$ 时能达到最小值就可以。这样导出的 BP 算法除输出层的 δ_j 不同以外，其他各层的方程与基本 BP 法没有什么差别。

3. 变步长的 BP 学习算法。BP 法在学习计算中, 学习率 η 和动量系数是由经验选定。对于学习率来说, η 越大, 权变化越大, 收敛越快。但 η 越大容易引起振荡, 所以 η 应在不引起振荡的情况下, 尽可能取最大值。这样可以在训练开始时, 可以给学习率 η 较大的值, 发现振荡时降低 η , 否则按原步长。

对于权值及阈值的起始值方面, 一般应选为均匀分布的小数经验值, 约为 $(-2.4/F, 2.4/F)$ 之间, 也有人建议在 $(-1.5/F, 1.5/F)$ 之间, 其中 F 为所连单元的输入端个数。

另外, 对于 N 个样本集合的离散时间序列, BP 神经网络可以完成从输入到输出的高度非线性映射, 即找到某种映射使: $F: R^m \rightarrow R^n$ 。Komogorov 定理表明, 给点任一连续函数 $f: R^m \rightarrow R^n$, $y=f(x)$, $R \in [0,1]$, f 可以精确地利用一个三层前向网络实现逼近。进而对于给定任意 ε 和在 L_2 范数下,

$\{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=1,2,\dots,N\}$ 存在一个三层 BP 神经网络, 它可以在任意

ε 平方误差精度逼近 f 。在实际应用过程中, 由于被逼近样本的性质不能精确知道, 因此即使在网络误差 E_l 为零的条件下, 也未必能保证 E_3 达到要求。往往会出现 E_l 非常小, 而 E_3 却无法达到要求。这就出现了所谓的“过拟合”现象, 从而直接影响网络的泛化能力, 使得网络最终失去实用价值。

针对上面情况, 本文提出一种改进的方法, 即把 N 个样本集合分成训练样本 δ_1 与 δ_2 和检测样本 δ_3, δ_4 :

$$\delta_1 = \{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=1,2,\dots,M, M < N\}$$

$$\delta_2 = \{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=M+1, M+2,\dots,P, P < N\}$$

$$\delta_3 = \{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=P+1, P+2,\dots,Q, Q < N\}$$

$$\delta_4 = \{(x_k, y_k) | x_k \in R^m, y_k \in R^n, k=Q+1, Q+2, \dots, N\}$$

判断 BP 网络是否具有很好的泛化能力, 主要利用训练样本通过训练建议映射关系, 再用检测样本检验网络能否给出正确的输入-输出关系。对于时间序列映射的建立方面, 可以令输入节点为 m , 输出节点为 n , 隐藏节点为 l 的三层 BP 神经网络来实现, 其中输入层到隐藏层的激活函数采用 Sigmoid 型隐藏层到输出层的激活函数采用线性函数。网络的输入与输出之间的关系如下:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3-19)$$

$$\hat{y}_k(t) = \sum_{j=1}^l v_{kj} \cdot f \left[\sum_{i=1}^m w_{ij} \cdot x_i(t) + \theta_j \right] + r_i$$

这里 $\hat{y}_k(t)$ 为网络期望输出, $y_k(t)$ 为网络的实际输出, 设定总网络误差值小于 ε_1 , 有:

$$E_1 = \frac{1}{2} \sum_{k=1}^M \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2 \leq \varepsilon_1 \quad (3-20)$$

$$E_2 = \frac{1}{P-M} \sum_{k=M}^P \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2 \leq \varepsilon_1 \quad (3-21)$$

设定检测样本平均均方误差小于 ε_2 , 有:

$$E_3 = \frac{1}{Q-P} \sum_{k=P}^Q \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2 \leq \varepsilon_2 \quad (3-22)$$

$$E_4 = \frac{1}{N-Q} \sum_{k=Q}^N \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2 \leq \varepsilon_2 \quad (3-23)$$

3.4 多目标优化的 BP 神经网络

在多目标优化过程中, 由于各个学科有各自的目标, 这些目标往往相互矛盾。在系统综合设计时, 通常不可能找到能使所有目标都同时达到最优的设计方案, 只有非劣意义上的最优。进化算法是一种以非数学模型为基础的优化方法, 快速、简便而且能够分段地找到 Pareto 最优解。为了求得多目标优化问题的有效解集, 通常的方法是把它转化为单目标优化问题, 应用标量优化技术进行求解。然而, 由于非线性标量优化问题的求解在许多情况下仍然很困难, 特别是对不可微、不连续问题更是如此。因此, 寻找求解多目标非线性优化问题有效解集的一般方法就成为学者们十分关注的问题。应用 BP 神经网络模型代替相应的子目标函数, 可以较好地解决上述问题^[43]。

3.4.1 智能函数及其逼近性能

这里智能函数具有最佳逼近和学习功能函数, 对于一个三层前馈神经网络可以看成从输入到输出的一类高度非线性映射, 其输入输出关系可表示为:

$$y = \sum_{i=1}^N c_i \varphi \left(\sum_{j=0}^{n-1} w_{ij} x_j - \theta_i \right) \quad (3-24)$$

若给定样本集 (x_i, y_i) , $i=1, 2, \dots, N$, 对网络进行训练后, 网络的权值和阈值均已确定, 此时该网络的软件所表示的输入输出关系是样本所隐含的函数关系。若所有的 c_i , w_{ij} 和 θ_i 已由给定的样本对网络训练后确定, 其相应的神经网络软件是描述所给样本函数关系的智能函数, 显然它具有容错性、自学习性和函数逼近功能, 关于其函数逼近功能有如下结果^[44]:

定理 3.1 设 $\psi(x)$ 为 D 上有界单调递增连续函数, D 为 R^n 的紧致子集, 对于任何连续函数 $F: D \subset R^n \rightarrow R^n$ 存在一个层数为 $k(k \geq 3)$, 隐层单元输出函数为 $\psi(x)$, 输入输出层单元的输出函数是线性的神经网络描述的智能函数一致逼近 $F(x)$, 其中 $\psi(x)=1/(1+e^{-x})$ 。

定理 3.2 设 D 为 R^n 的紧致子集, $Q(x): D \subset R^n \rightarrow [a, b]^n$, $(0 < a < b < 1)$ 为 D 上连续有界实值函数, 对任意给定的 $\varepsilon > 0$, 存在一个三层 BP 神经网络, 其隐层及输出层单元以 $f(u)=1/(1+e^{-u})$ 为输出函数, 输入层单元的输出函数是线性的, 该神经网络软件描述的智能函数任意逼近 $Q(x)$ 。

3.4.2 BP 神经网络的优化设计

BP 神经网络在实际应用过程中, 首先需要对网络进行训练才可以使用, 对于训练样本的获取可以有 3 种方法:

1. 线性或非线性规划法。即已知参数集和变量集, 需求目标集。
2. 计算机仿真法。可利用大量的有限元计算结果作为 BP 神经网络的训练样本。
3. 实际生产中的成功运行数据采样法。对于复杂系统, 参数和变量与目标及约束之间, 并无确定的设计规划公式, 但是可以根据实际生产中已成功运行的事例作为神经网络的训练样本, 即按一定的评价方法, 对该系统中的各项指标进行评价。

将训练后的 BP 网络用于优化求解过程中的目标函数值的仿真计算^[45]。

对于给定优化模型:

$$F(X)_{\min} = \min[f_1(x), f_2(x), \dots, f_m(x)] \quad X \in E^n \quad (3-25)$$

式中, X 为 n 维欧氏空间向量; $f_i(x)$ 为目标函数; E^n 为可行域。在多目标优化问题中, 一般由非劣解的概念来描述向量函数优化的解一个可行决策向量 $x^* \in X$ 称为非劣解, 如果不存在 $x \in X$ 使得 $f(x) \leq f(x^*)$ 及 $f(x) \neq f(x^*)$, 非劣解集由 X^* 表示。

这里使用求解多目标优化最基本的方法——评价函数法, 即将多目标优化

问题转化为单目标优化问题进行求解。先求解 m 个单目标优化问题：

$$\min f_i(X), \quad i=1,2,\dots,m \quad X \in E^n \quad (3-26)$$

设其最优值为 f_i^* ，称 $F_i^* = (f_1^*, f_2^*, \dots, f_m^*)^T$ 为值域中的一个理想点，在某种度量下，寻求距离 F^* 最近的 F 作为近似值，因此构造评价函数

$$\varphi(F) = \sqrt{\sum_{i=1}^m (f_i - f_i^*)^2} \quad (3-27)$$

这样就将式(3-18)表示的多目标优化模型转化为式(3-21)表示的单目标优化模型求解。

$$\min^* [F(X)] = \sqrt{\sum_{i=1}^m [f_i(X) - f_i^*]^2} \quad (3-28)$$

根据上面的分析，优化设计的步骤为：

1. 确定 BP 网络输入维数与输出维数。
2. 准备网络的训练集，建立 BP 神经网络模型并用得到的样本数据对网络进行训练。用训练好的神经网络模型作为目标函数值仿真计算工具。
3. 分别对目标函数值进行单目标优化，得到理想点。
4. 对式(3-28)进行单目标优化，得到式(3-25)多目标优化的结果。

3.5 本章小结

本章通过人工神经元以及人工神经网络(ANN)的引入，重点介绍了 BP 神经网络基本理论与方法。并通过对 BP 神经网络的限制与不足进行分析，提出一种改进技术。通过对训练样本的调整和处理，以两个验证误差指标作为检验函数，来避免神经网络的过拟合现象。最后分析了 BP 神经网络在多目标优化问题的应用，并给出了 BP 神经网络求解多目标优化问题的实现步骤。

第4章 改进遗传算法及其与 BP 神经网络融合

优化问题的智能计算是最近几年优化算法研究的前沿课题。遗传算法和神经网络是该领域研究的两个方向，但基于不同的机理，其研究的大部分成果和思路是在各自的领域中进行。直到近些年，这两种方法才出现互相融合的趋势，但如何融合，依然是有待于探讨的一个重要课题^[46, 47]。遗传算法与神经网络的相互补充可增强彼此的能力，从而获得更有力的表示和解决实际问题的能力。

本章将在前两章的基础上，首先重点讲解一下 NSGA-II 算法。随后，将给出本文所提出的改进的并行混合非劣分类遗传算法 (Intercurrent Mixed Non-dominated Sorting Genetic Algorithm II, IMNSGA-II)。然后，研究遗传算法与 BP 神经网络的融合技术，充分结合遗传算法和 BP 神经网络的优势并提出了一种遗传-神经混合算法。

4.1 改进的多目标遗传算法

4.1.1 NSGA-II 算法原理

在本文第二章第二节我们提到 Deb.et 等学者在 2000 年对原始的 NSGA 进行改进，提出改进型非劣分类遗传算法(Nondominated Sorting Genetic Algorithm II, NSGA-II)，算法流程如图 4-1 所示。

首先，用父代种群 P_n 产生子代种群 Q_n ，并将两个种群联合在一起形成大小为 $2n$ 的种群 R_n ，并对具有 $2n$ 规模的种群 R_n 进行非劣分类操作。计算每一非劣等级所有个体的拥挤距离。按照拥挤选择算子的原则产生下一代种群 P_{n+1} ，进化代数计数器 n 加 1 并判断 n 是否大于最大进化代数。如果是，那么该算法结束；否则，继续进化。如此循环，直到进化到指定的最大进化代数。

该算法中对联合的父代种群和子代种群进行非劣分类并填充新一代种群 P_{n+1} 的过程如图 4-2 所示。对 R_n 中的第一非劣等级判断并决定该非劣等级能否被新种群接受。如果能，则将该非劣等级的所有个体填充到新种群中，继续判断下一非劣等级能否被新种群接受。如此反复，直到不能容纳该非劣等级的所有个体。对最后不能被完全容纳的非劣组中的个体计算拥挤距离并选择分布最广的 $N-|P_{n+1}|$ 个个体填充新种群。

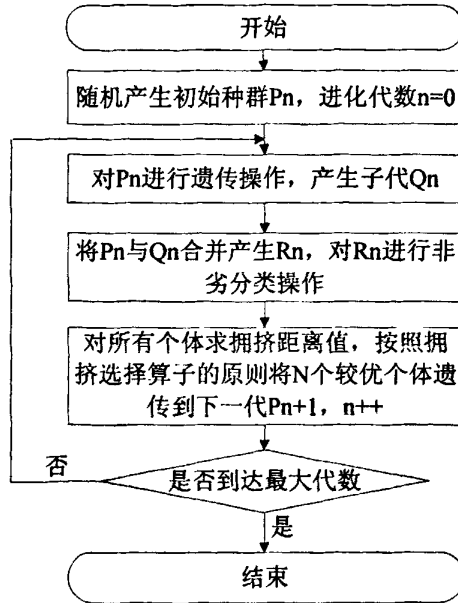


图 4-1 NSGA-II 算法流程

Fig. 4-1 Flow chart of NSGA-II

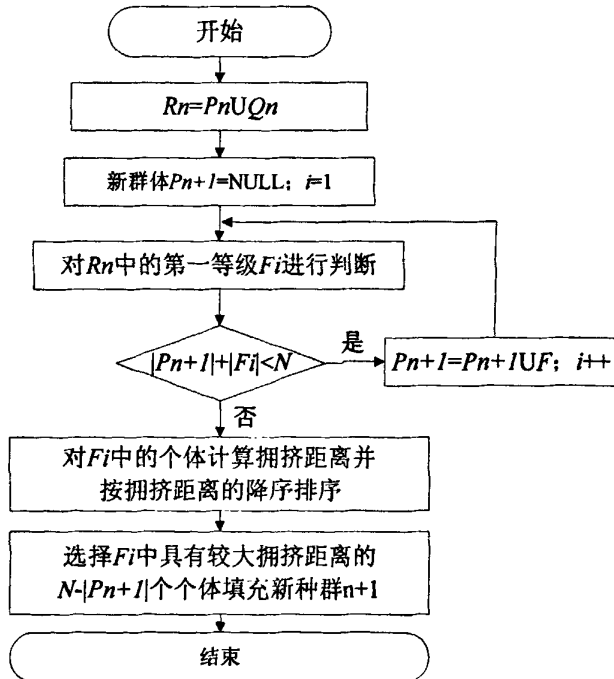


图 4-2 NSGA-II 算法中新一代种群的生成算法流程

Fig. 4-2 Flow chart of creating the new generation in NSGA-II

4.1.2 并行混合进化机制

精英策略已经被认为是一种改进多目标遗传算法效率的有效手段,当前在多目标优化领域的一些研究也显示该策略能有效改善 MOEA 算法的效率^[38]。但是通过研究 NSGA-II 和其它一些采用精英策略算法发现存在如下问题:精英保持也就意味着强调子代中的最优解。在程序运行到一定代数后种群中的所有解都是精英,随后各代的操作都将在精英解中进行,非精英解无法参与其中,降低了解的多样性。这样由于非精英解不能参与遗传操作,使得全局解的搜索速度减慢并最终导致种群过早收敛到局部 Pareto 解。通过分析本章第二节的算法,本文提出一种改进的并行混合非劣分类遗传算法 (IMNSGA-II)。

并行混合进化处理如下,其结构如图 4-3 所示:

1. 产生初始群体 P 和一个空的外部存档集 P' 。

2. 父代种群 P_n 产生子代种群 Q_n , 产生种群 $R_n = P_n \cup Q_n$, 并对 R_n 进行快速非支配排序,并对每个个体计算拥挤距离。选择当前种群中最好的非劣解,即精英解。随着等级数 i 值的增大,各个等级中最好的非劣解数目依次减少,为了保持非精英解能参与下一代的进化,此时我们限制当前精英解集的范围从而使非精英解集也能平等的参与到下一代进化中,即限制当前等级中允许的最大个体数。同时将 R_n 中精英解的副本保持到 P' 。

3. 此时 P' 来自于 t 代精英解(第一代直接将 P' 复制到 P'),而 P 来自于 $t+1$ 代的精英解,相对于 P' 来说此时 P 为外部种群,此时 $P' + P$ 种群综合运用联赛选择和共享函数的思想来选择当前群体中优良的个体遗传到下一代群体中,从而保持外部存档集种群数目不超过预定的数 N' 。

共享函数的方法如下:

1. 从群体中随机选取 k 个个体组成个体比较集合 C , 其中 k 为预先指定参数。

2. 从群体中随机选择两个个体组成个体联赛集合 T 。

3. 分别比较个体联赛集合 T 中的两个个体与个体比较集合 C 中各个个体之间的优胜关系,按下述方法从 T 中选择一个个体遗传到下一代群体。

4. 如果集合 T 中的一个个体 X 比集合 C 中的所有个体优越,而集合 T 中的另一个个体都不比 C 中的所有个体优越,则将个体 X 遗传到下一代群体中。如果未能选择出一个个体,则利用共享函数的概念从集合 T 中选出一个小生境较小的个体遗传到下一代。

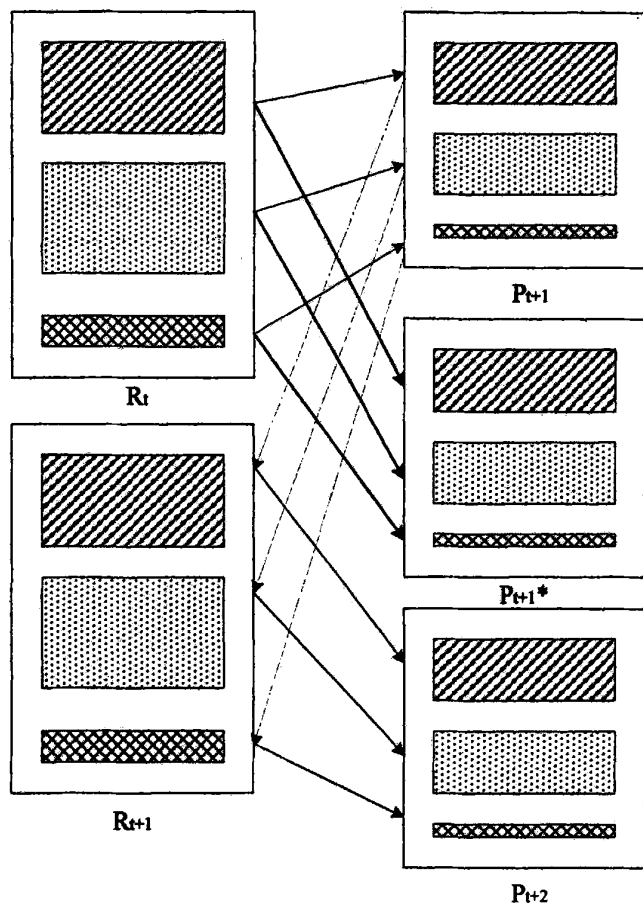


图 4-3 并行混合进化框图

Fig. 4-3 Parallel hybrid evolutionary diagram

4.1.3 改进的并行混合遗传算法实现过程

下面给出本算法：

1. 初始化控制调节参数：进化代数 GEN ，生成初始种群 P_0 ，规模为 N ，初始外部存档集为 $P' = NULL$ 及预定规模 N' ，交叉概率 P_c ，变异概率 P_m 。
2. 用父代种群 P_t 产生子代种群 Q_t ，并将两个种群联合在一起形成大小为 $2n$ 的种群 R_t ，并对具有 $2n$ 规模的种群 R_t 进行非劣分类操作，按照拥挤选择算子的原则与分级策略产生下一代种群 P_{t+1} 。并将副本复制到 P' 。
3. 与 NSGA-II 类似，按照 2. 的方法将 N 个符合条件的解放入下一代中，继续对其执行选择、交叉、变异等遗传操作。同时将 P' 与 P 进行联赛选

择和共享函数方法进行并行进化。

4. 如果达到规定的最大进化代数则停止运行，否则转步骤 2.。

算法的流程图如图 4-4 所示。

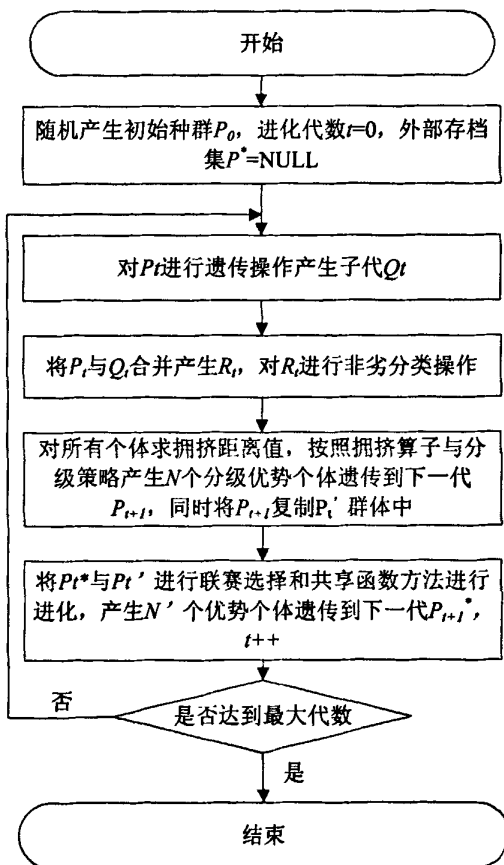


图 4-4 IMNSGA-II 算法流程图

Fig. 4-4 Flow chart of IMNSGA-II

在第 5 章中，改进的并行混合非劣分类遗传算法将与 NSGA-II 在 5 个困难多目标问题上进行性能的比较。

4.2 遗传算法与 BP 神经网络融合技术

4.2.1 问题的提出

神经网络和遗传算法都是人们仿效生物处理模式的方法，并从中获得的用

于处理复杂问题的方法。然而,随着 BP 神经网络研究的不断深入发现在实际应用过程中 BP 算法存在以下两个主要问题:

1. 从数学角度上,多层感知器算法是一个非线性优化问题,由于采用了传统的无约束最小化方法来实现误差函数的极小化,因此不可避免地存在有局部极小问题。

2. 神经网络的设计主要依据设计者的经验在大样本空间反复实验来进行选取,尚无理论上的指导,因此在神经网络的初始连接权以及网络结构的选择上具有很大的随机性,很难选取具有全局性的初始点,因而网络求得全局最优的可能性小。

遗传算法虽然对于寻优问题有很好的自适应优化搜索能力,但是它不具备自适应学习能力,难以单独有效地作为一种控制方法研究,但由于遗传算法能够收敛到全局最优解,且遗传算法的鲁棒性强,将遗传算法与神经网络融合起来不仅能发挥神经网络的泛化映射能力,而且使神经网络具有很快的收敛性以及较强的学习能力。

遗传算法具有全局搜索的特性且不依赖梯度信息,也不需要求解函数可微,只需要求解函数在约束条件下可解,用它优化神经网络的连接权和网络结构,可以较好地克服 BP 神经网络的问题并且有效提高神经网络的泛化性能。遗传算法和神经网络的融合对于寻求全局最优解其效果要优于单个的使用遗传算法或神经网络^[47, 48, 49]。

4.2.2 GA-BP 神经网络融合

遗传算法的 BP 神经网络的融合数学描述如下^[50]:

$$\begin{cases} \min E(w, v, \theta, r) = \frac{1}{2} \sum_{k=1}^M \sum_{i=1}^n [y_k(t) - \hat{y}_k(t)]^2 \\ s, t: w \in R^{m \times p}, v \in R^{n \times p}, \theta \in R^p, r \in R^n \end{cases} \quad (4-1)$$

其中 E_1 为网络总的误差, $y_k(t)$ 为期望输出, $\hat{y}_k(t)$ 为网络实际输出

$$\hat{y}_k(t) = f \left\{ \sum_{j=1}^p v_{kj} \cdot f \left[\sum_{i=1}^m w_{ij} \cdot x_i(t) + \theta_j \right] \right\} + r_k \quad (4-2)$$

设 $E_2 = \frac{1}{N - N_1} \sum_{k=N_1}^N \sum_{i=1}^n [y_k(t) - \hat{y}_k(t)]^2$ 为检测样本平均均方误差,表示对网络输出数据可靠性的估计, E_2 小,网络输出的可靠性大一些,否则网络输出就不大可靠。

在 BP 神经网络中,权值的初始化强烈地影响着最终的解。不同的初始权

值设置可能会对训练时间、收敛性等造成巨大的差异。为了更好地解决网络初始权值与阈值的确定问题，可采用基于遗传算法的权值与阈值，利用遗传算法的全局搜索能力来确定初始权值和阈值。神经网络融合遗传算法结构图如图 4-5 所示。

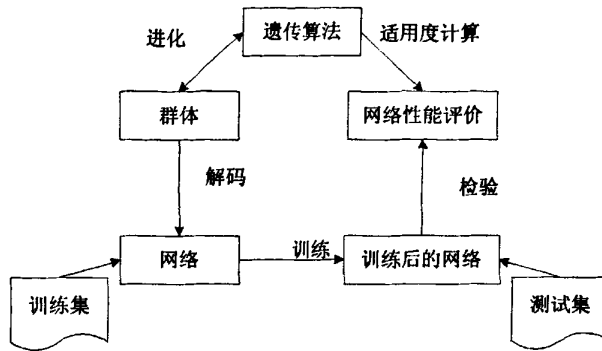


图 4-5 遗传算法与 BP 神经网络的融合

Fig. 4-5 Combination of genetic algorithm and BP neural network

我们先采用三层 BP 网络来初步确定解空间，设定网络的训练次数以及网络的训练误差。在训练样本误差和检测样本都比较满意时，把连接权值中的最大值和最小值分别记为 w_{\max} 与 w_{\min} ，设连接权的基本解空间为 $[w_{\min} - \delta_1, w_{\max} + \delta_2]$ ， (δ_1, δ_2) 为调节参数。由于染色体表示权值，则适应度函数便是神经网络的误差计算，且误差越大，适应度越小。设定输入种群规模、交叉概率(P_c)、变异概率(P_m)、网络层数、每层的神经元数，使用 GA 反复优化神经网络的权值，直到平均值不再有意义地增加为止，此时解码得到的参数组合已经充分接近最佳参数组合，再用 BP 算法在这些小的解空间中对网络的连接权和阈值再次寻优，搜索出最优解。

4.2.3 算法实现的关键技术

1. 个体的编码形式与初始化 个体的编码采用实数编码方案，Michalewicz 指出，对于实数值优化问题，浮点表示比二值表示运行得更好，这是因为它们具有更好的一致性、更准确，导致更快的执行。将决策变量用实数编码，群体中每一个个体的表示形式 $x = (x_1, x_2, \dots, x_n)$ 。对于各个个体 x_i ，初始值按式(4-3)产生均匀分布在 $[a_{i\min}, b_{i\max}]$ 的随机数。基于实数编码的遗传算法流程如图 4-6。

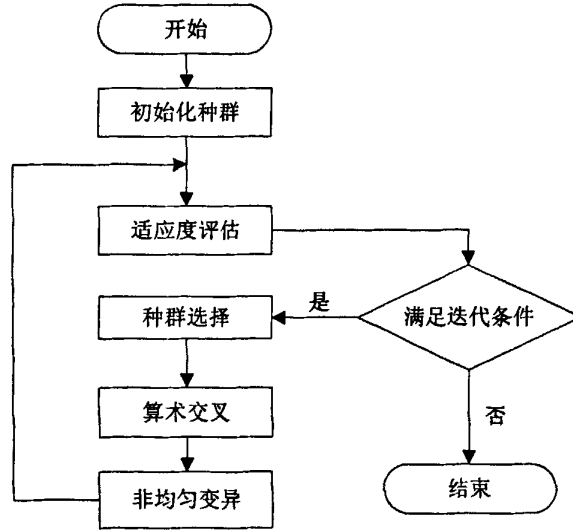


图 4-6 基于实数编码的遗传算法

Fig. 4-6 Genetic algorithm based on real-coded

$$x_i = ai_{\min} + (bi_{\max} - ai_{\min}) \cdot rand(1) \quad (4-3)$$

式中, ai_{\min} 为第 i 个目标参数的最小值, ai_{\max} 为第 i 个目标参数的最大值。

对于权重的初始化问题, 普通的 BP 算法网络的初始权重一般取 0~1.0 之间均匀分布的随机数, 而在初始染色体集中, 网络的各权重是以概率分布 $e^{-|w|}$ 来随机确定的, 这一点是有别于 BP 算法。遗传算法的这种随机分布的取法是通过以往的大量实验得出的, 可以发现当网络收敛后, 权重的绝对值一般都较小。

本文将 BP 网络权值与阈值按一定的顺序级联起来, 形成一个数组, 作为遗传算法的一个染色体。遗传操作在这样的染色体种群中进行。个体的长度为神经网络的权值的个数和阈值个数之和, 即:

$$n = H \cdot (X + Y) + H + Y \quad (4-4)$$

式(4-4)中 H 为隐含层数, X 为输入层数, Y 为输出层数。

2. 适应度分配 在基于 Pareto 最优概念的多目标遗传算法中, 较常见的是基于个体排序的适应度赋值方式。与单目标优化不同是, 多目标优化优劣性与支配关系并非定义目标向量之间的整体有序关系, 只是给出部分有序关系, 因而种群的级别排序不具有唯一性。这里我们将利用基于个体排序的适应度赋值方式。假设第 n 代种群中的个体 x_u 其对应的目标向量为 u , 定义 $r_u^{(n)}$ 为当前种

群中支配目标向量 u 的个体数目, x_u 在第 n 代种群个体排序中的位置设为 $\text{rank}(x_u, n)$, 取 $\text{rank}(x_u, n) = r_u^{(n)}$, 则所有当前种群中非劣最优个体的排序均预定为 0。其操作步骤:

1. 首先需要对目标向量的每个子目标取值进行计算。
2. 基于 $\text{rank}(x_u, n)$ 的数值将种群中所有个体进行分级别排序。
3. 利用线性或非线性的插值方式在最低序号(非劣最优个体)与最高序号($\leq N$)之间进行插值。
4. 具有相同序号的个体进行适应度共享算子操作, 即通过除以相同序号的个体数目得到新的适应度值。

BP 网络的评估标准就是输出值与期望输出值之间的误差。其平方和越小性能越好。遗传算法利用适应度来评估染色体的生存能力。将均方误差函数作为适应度函数可以充分地把两者的评估标准融为一体, 提高网络的优化性能。

4.2.4 GA-BP 算法设计

基于遗传算法的 BP 神经网络融合算法(GA-BP)的具体步骤如下:

1. 初始化 BP 网络模型, 确定输出与输入个数 X, Y , 网络的层数 L 、网络的训练次数 N 等; 初始化学参数精度 $\varepsilon \geq 0$; 调节因子 σ_1, σ_2 ; 种群规模 P 与最大迭代次数 T ; 交叉概率 P_c 与变异概率 P_m ; 锦标赛规模 K ; 累计迭代次数 $t=0$ 。

2. 将 BP 网络的权值和阈值按输入层与隐含层权值和阈值、隐含层与输出层的权值与阈值顺序级联起来, 随机产生 n 个染色体, 形成初始化种群 P_t 。

3. 将均方误差函数作为适应度函数, 计算每一个染色体的适应度。用父代种群 P_t 产生子代种群 Q_t , 并将两个种群联合在一起形成大小为 $2n$ 的种群 R_t , 并对具有 $2n$ 规模的种群 R_t 进行非劣分类操作, 按照拥挤选择算子的原则产生下一代种群 P_{t+1} 。并将副本复制到 P' 。

4. 与 NSGA-II 类似, 按照步骤 3 的方法将 n 个符合条件的解放入下一代中, 继续对其执行选择、交叉、变异等遗传操作。同时将 P^* 与 P' 进行联赛选择和共享函数方法进行并行进化。

5. 判断是否到达最大迭代次数, 如果符合或转向步骤 6, 否则转向步骤 3。

6. 将 P^* 中最优个体按编码顺序解码, 作为 BP 网络的初始权值和阈值。

7. 进行 BP 网络的前向传播, 计算全局误差, 并判断是否符合要求, 如果

符合则结束网络学习。

8. 如果小于循环次数则进行 BP 网络反向传播, 修改阈值, 返回步骤 6; 否则结束网络学习。

在第 5 章中, 我们将把 GA-BP 算法应用到多目标车间作业动态调度问题实例中, 进而验证 GA-BP 算法的有效性与可行性。

4.3 本章小结

本章根据前两章对遗传算法与 BP 神经网络的分析 and 提出的不足, 以及加以改进的基本思想, 在非劣分类遗传算法基础上提出一种新的改进精英保持并行混合遗传算法, 并给出该改进多目标遗传算法的设计原理, 算法流程及具体的方案, 实现了将遗传算法与 BP 算法融合。随后对其结构和具体细节以及主要算法进行了描述, 并将其应用到多目标优化问题的求解中。遗传-神经算法主要特点是将遗传算法全局搜索最优和传统的 BP 神经网络模型局部寻优结合起来, 取长补短, 既可以减小遗传算法的搜索空间、提高搜索效率, 又可以较容易地收敛到最优解, 为求解多目标优化问题提供了新的策略。

第5章 改进算法的性能测试与应用实例

在多目标遗传算法的研究领域关于如何选择恰当的 MOP 问题进行算法性能测试已有不少的讨论。本文为了测试改进的并行混合非劣分类遗传算法的性能,我们用 C++ 语言与 Matlab 将改进的并行混合非劣分类遗传算法(IMNSGA-II)完全实现,用此程序产生多目标优化的结果,并将 IMNSGA-II 与 NSGA-II 在 5 个困难多目标问题(ZDT1, ZDT2, ZDT3, ZDT4, ZDT6)上进行性能的比较。

通过将改进的遗传算法与 BP 神经网络融合后的算法应用于多目标车间作业动态调度实例中,验证本文提出的 GA-BP 算法的可行性与有效性。

5.1 改进多目标遗传算法的数值实验

5.1.1 针对不同问题特性设计测试函数

选择问题测试集的目的在于检验一种多目标进化算法的性能和效率。为了验证改进算法的性能,本文选择文献[38]中的 5 个标准测试函数作为测试用例,文献中 DEB 指出了多目标优化中比较困难的几个问题:

1. 收敛到 Pareto 最优阵面,在单目标遗传优化中的多峰函数,欺骗函数和孤立最优函数都是很著名的问题。
2. 为了获得好的 Pareto 最优阵面的分布特性,在群体中维持个体多样性。

然而, Pareto 最优阵面的某些特征会不利于多目标遗传算法找到各种各样的 Pareto 最优解:凸或非凸的,离散的,非均匀分布的。针对具有这些特性的 Pareto 最优解,DEB 提出了很多函数实例以方便使用 f 、 g 、 h 函数为此通过这些函数实例的组合变化可以使测试的多目标问题具有各种特征。为了简化测试,每个测试函数只限于两个目标。实际上,二维问题已经反映了多目标优化中的实质问题。这些供组合的函数实例的定义如表 5-1 所示。

在 NSGA-II 和 IMNSGA-II 算法中,均采用实数编码方式采用实数编码,进化算子模拟二进制交叉(SBX)、实参数变异、基于小生境的锦标赛与共享保持选择。

算法参数设置如表 5-2 所示,其中变异概率的 n 为决策因子的数目,这里

我们采用基于优胜和 Pareto 级别排序适应度赋值。

表 5-1 常用多目标优化问题

Table 5-1 Common multi-objective optimization problems

MOP	定义	约束条件
ZDT1: PF _{true} 为凸性	$F = (f_1(x_1, f_2(x))),$ 其中 $f_1(x_1) = x_1,$ $f_2(x) = g(1 - \sqrt{(f_1/g)}),$ $g(x) = 1 + 9 \sum_{i=2}^m x_i / (m-1)$	$m = 30$ $0 \leq x_i \leq 1$
ZDT2: PF _{true} 为非凸性	$F = (f_1(x_1, f_2(x))),$ 其中 $f_1(x_1) = x_1,$ $f_2(x) = g(1 - (f_1/g)^2),$ $g(x) = 1 + 9 \sum_{i=2}^m x_i / (m-1)$	$m = 30$ $0 \leq x_i \leq 1$
ZDT3: PF _{true} 为非连续	$F = (f_1(x_1, f_2(x))),$ 其中 $f_1(x_1) = x_1,$ $f_2(x) = g(1 - \sqrt{(f_1/g)} - (f_1/g) \sin(10\pi f_1)),$ $g(x) = 1 + 9 \sum_{i=2}^m x_i / (m-1)$	$m = 30$ $0 \leq x_i \leq 1$
ZDT4: 局部 PF _{true}	$F = (f_1(x_1, f_2(x))),$ 其中 $f_1(x_1) = x_1,$ $f_2(x) = g(1 - \sqrt{(f_1/g)}),$ $g(x) = 1 + 10(m-1) + \sum_{i=2}^m (x_i^2 - 10 \cos(4\pi x_i))$	$m = 10$ $0 \leq x_i \leq 1$
ZDT6: PF _{true} 为非幺正	$F = (f_1(x_1, f_2(x))),$ 其中 $f_1(x_1) = 1 - \exp(-4x_1) \sin^6(6\pi x_1),$ $f_2(x) = g(1 - (f_1/g)^2),$ $g(x) = 1 + 9((\sum_{i=2}^m x_i) / (m-1))^{0.25}$	$m = 10$ $0 \leq x_i \leq 1$

表 5-2 算法参数设置

Table 5-2 Parameter settings for algorithms

算法名称	进化代数	种群大小	变异概率	交叉概率	二进制交叉算子 η_c	实参变异算子 η_m
NSGA-II	300	100	1/n	0.9	20	20
IMNSGA-II	300	100	1/n	0.9	20	20

性能度量值方面，与单目标不同，多目标优化性能的度量有两个标准，即收敛到 Pareto 优解集和保持 Pareto 优解集的多样性。所以，这两个指标不能用一个性能指标来评测。为了说明改进算法的有效性，并与其他算法进行比较，本文采用文献[38]提出的两个评价标准。

5.1.2 试验结果与分析

图 5-1 到图 5-5 为两种算法分别对测试函数 ZDT1、ZDT2、ZDT3、ZDT4 和 ZDT6 运行 300 次后得到最好的非支配曲面的比较图，从比较图中可以看出，算法 IMNSGA-II 得到 Pareto 解的数量更多，分布更均匀，所得的结果优于算法 NSGA-II。

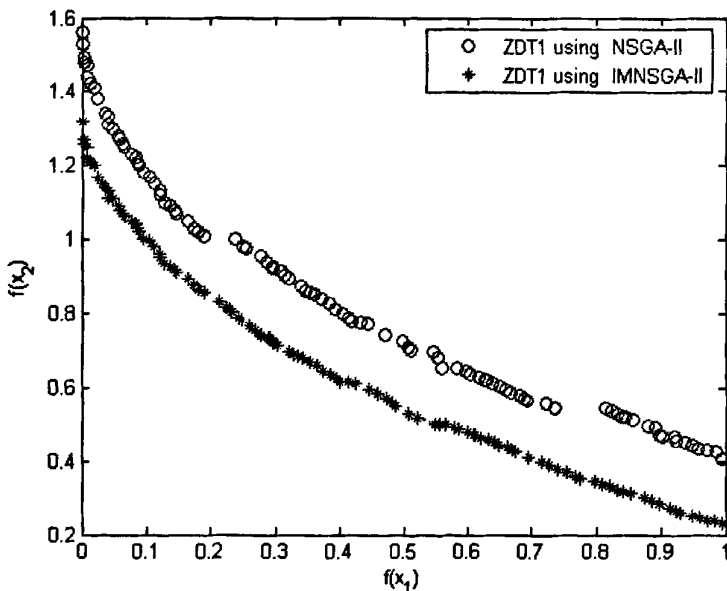


图 5-1 ZDT1 在 NSGA-II 与 IMNSGA-II 运行结果图

Fig. 5-1 The result of ZDT1 using NSGA-II and IMNSGA-II

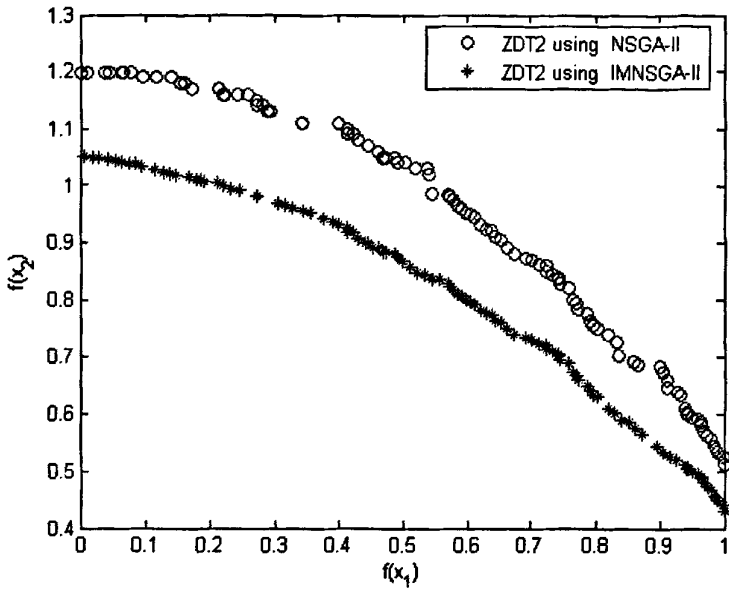


图 5-2 ZDT2 在 NSGA-II 与 IMNSGA-II 运行结果图

Fig. 5-2 The result of ZDT2 using NSGA-II and IMNSGA-II

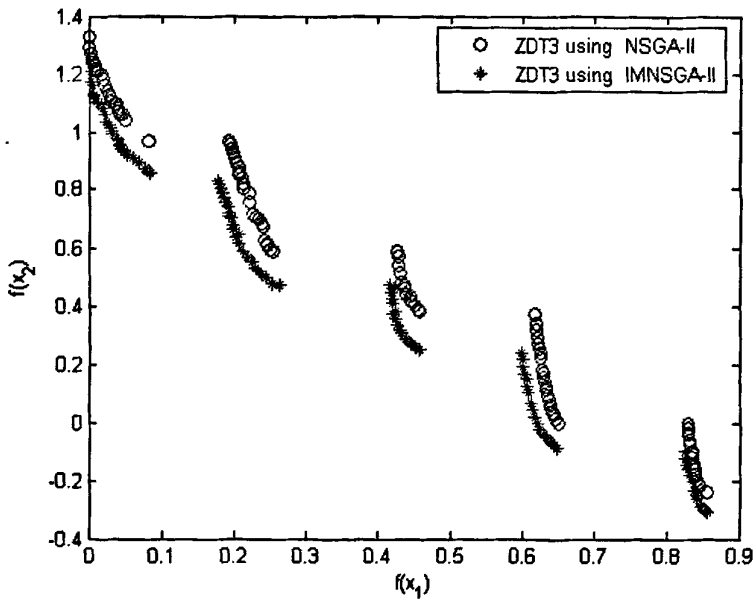


图 5-3 ZDT3 在 NSGA-II 与 IMNSGA-II 运行结果图

Fig. 5-3 The result of ZDT3 using NSGA-II and IMNSGA-II

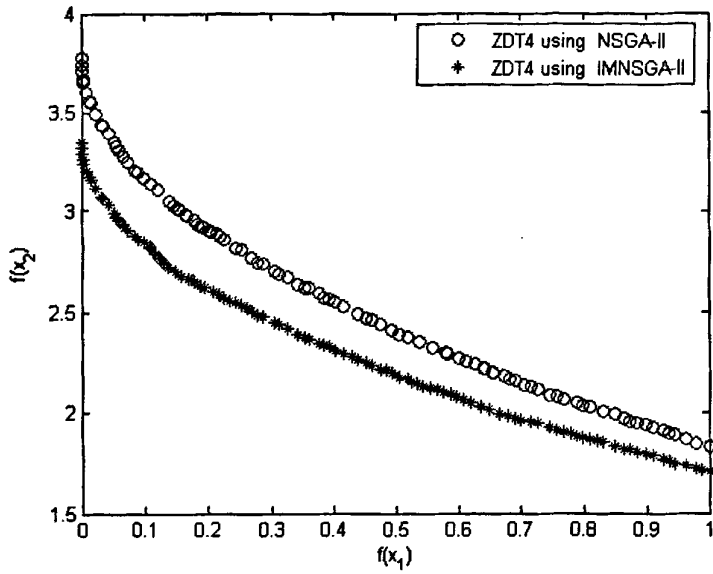


图 5-4 ZDT4 在 NSGA-II 与 IMNSGA-II 运行结果图

Fig. 5-4 The result of ZDT4 using NSGA-II and IMNSGA-II

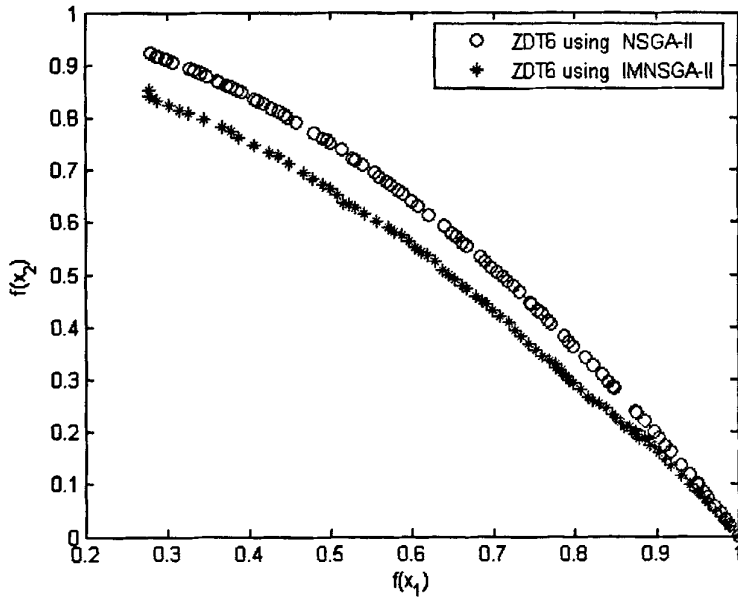


图 5-5 ZDT6 在 NSGA-II 与 IMNSGA-II 运行结果图

Fig. 5-5 The result of ZDT6 using NSGA-II and IMNSGA-II

1. 收敛性 γ 收敛性用来评价算法所得到的非劣解集 Q 与问题的真正 Pareto 前沿近似集 P^* 之间的逼近程度:

$$\gamma = (\sum_{i=1}^{|Q|} d_i) / (|Q|) \quad (5-1)$$

式中, d_i 表示 Q 中第 i 个个体与 P^* 之间在目标空间的最小欧氏距离。 γ 值越小, 表明算法所得非劣解集 Q 越接近所求问题的真正 Pareto 前沿, 算法收敛性越好。表 5-3 显示了采用 NSGA-II 和 IMNSGA-II 两种算法在 300 次运行后所获得的收敛性度量值 γ 的平均值(上)和偏移值(下), 从表 5-3 可以看出改进的算法具有很好的收敛性。

表 5-3 各算法收敛性比较

Table 5-3 Comparison of each algorithm's convergence

算法	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.034483	0.078291	0.114502	0.513052	0.297576
	0.004767	0.034679	0.007940	0.118378	0.013134
IMNSGA-II	0.009042	0.008243	0.074855	0.067214	0.085720
	0.000279	0.002858	0.006446	0.027338	0.005764

2. 多样性 Δ 多样性用于评价算法的分布广度和均匀程度:

$$\Delta = (d_f + d_i + \sum_{i=1}^{|Q|-1} |d_i - d|) / (d_f + d_i + (|Q| - 1)d) \quad (5-2)$$

其中, d_i 是非劣最优目标域 Q 中连续两个非劣解向量间的欧氏距离; d 是所有 d_i 的均值; d_f 和 d_i 分别表示真正 Pareto 前沿近似集 P^* 中边界点和非劣解集 Q 边界点间的距离。 Δ 值越小, 表明算法的多样性越好。表 5-4 显示了采用 NSGA-II 和 IMNSGA-II 两种算法在 300 次运行后所获得的多样性度量值 Δ 的平均值(上)和偏移值(下)。

表 5-4 各算法多样性比较

Table 5-4 Comparison of each algorithm's diversity

算法	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.391308	0.440778	0.738542	0.703622	0.668020
	0.001896	0.004732	0.019706	0.065648	0.009923
IMNSGA-II	0.257731	0.251325	0.656543	0.525207	0.536751
	0.001254	0.003604	0.014214	0.005223	0.007924

5.1.3 结果讨论

从最终 Pareto 解集的图示中可以看出本文第四章提出的改进的并行混合非劣分类遗传算法具有不错的性能。同时进一步验证了本文提出的并行混合优化机制的可行性与有效性。在改进的多目标遗传算法中, 并行混合进化机制不仅

保存了从上一代至今最优的非劣解，使精英解的作用代数得以延续，而且外部同步进化机制的引入保持非精英解能参与到下一代的进化，我们限制了当前精英解的范围，保留部分非精英解，同时利用小生境的思想不断混入隔代因子，保持种群的多样性维持了种群的多样性。通过分析性能度量值 γ 与 Δ ，改进的算法在收敛性方面有较大改观，在多样性及收敛性之间达到了一个较好的平衡。同时，我们所使用的量化性能度量值已经足以表明改进的算法和 NSGA-II 算法性能上的差异。

5.2 GA-BP 算法在多目标车间作业动态调度问题应用

5.2.1 问题描述

车间作业调度是针对一项可分解的工作(如产品生产)，探讨在一定的约束条件下(如工艺约束，交货期约束)，如何安排其组成部分(操作、所占用的资源、加工时间及先后顺序)，以获得产品制造时间或成本的最优化^[51]。这里我们提出的车间作业动态调度问题主要针对参考文献[52]提出的多品种、小批量生产企业车间作业调度问题。基于多品种、小批量车间作业调度主要生产状态参数即设备的生产能力、设备的利用率以及队列情况，订单加工的紧迫度、仓库存储费用。

1. 机床待加工任务队列数，机床的加工任务队列数反应了机床负荷率的大小，队列数越多表明机床负荷越大，机床的加工任务较重。

2. 机床利用率 u ：机床加工利用率是指机床在一定时间 T 内的加工总时间与 T 的比值，一般 T 选取所有零部件中最大交货期与现在时刻的差值 $T = d_{\max} - t$ ，其中 T_{ijk} 表示机床 K 上的队列任务加工时间。机床利用率越大表明机床在这段时间内的使用频率高，反之机床使用频率较低，这时就需要加大该机床的加工任务，提高机床利用率。

3. 交货延期的平均惩罚系数 λ ：交货延期的平均惩罚费用计算公式： $D = \lambda t$ ， λ 、 t 分别表示交货延期惩罚系数和延期时间；不同的产品具有不同惩罚系数， r 越大表明交货延期的代价越高，因此就需要考虑尽量缩短延期的时间，安排提前加工。

4. 零部件的加工紧迫度 CR ：零部件的加工紧迫度是将剩余时间与需要加工的时间(计划提前期)对比， $CR = L/R$ ，其中 CR 表示工件的加工紧迫系数， L 、 R 分别表示加工剩余时间和计划提前期。

5. 零件单位库存费用率 i ：库存零部件占用费用的计算公式为： $a = nCi$ ，其中 n 和 C 分别表示存储的零件数和单位成本。

6. 车间作业调度优化目标方面文章选择最短生产周期 $f_1(x)$ 、最少工件平均驻留时间 $f_2(x)$ 、最小平均延滞 $f_3(x)$ 三个指标作为车间作业调度模式的优化目标。这里考虑车间作业调度的调度策略主要有：最早交货期策略 EDD 、优先选择等待加工时间最短的机床加工的调度策略 W_k 和优先选择利用率最低的机床加工的调度策略 U_k ^[52, 53]。

5.2.2 车间作业动态调度算法的实现

在解决车间作业动态调度问题，本文从实际的、已知正确的钻床车间选取 100 对样本，其中 70 对样本作为训练样本，30 对作为测试样本，每一对样本包括 11 个数据，分别为机床待加工任务队列数、机床加工利用率、零部件的加工紧迫度、交货延期的平均惩罚系数、存储的平均费用系数以及三个优化（即目标最短生产周期、最少工件平均驻留时间、最小平均延滞）八种参数作为输入层神经元；以车间作业的最早交货期策略、等待加工时间最短的机床加工的调度策略、利用率最低的机床加工的调度策略为输出层参数。部分样本数据如表 5-5 所示。

表 5-5 部分训练样本数

Table 5-5 Partial data of training samples

样本	队列数	机床利用率	加工紧迫度	延期惩罚费用	存储费用系数	$f_1(x)$	$f_2(x)$	$f_3(x)$	EDD	W_k	U_k
1	3	0.782	0.365	0.15	0.20	0.20	0.20	0.60	0.10	0.90	0.10
2	1	0.892	0.982	0.20	0.28	0.60	0.10	0.30	0.10	0.90	0.10
3	2	0.862	0.892	0.30	0.40	0.20	0.60	0.10	0.90	0.10	0.10
4	1	0.568	1.278	0.15	0.14	0.20	0.20	0.60	0.10	0.10	0.90
5	3	0.514	0.952	0.25	0.17	0.3	0.2	0.5	0.10	0.10	0.90
6	2	0.617	0.568	0.50	0.45	0.30	0.60	0.10	0.90	0.90	0.10

训练 GA-BP 网络时，采用三层网络结构，输入层的神经元个数 Y 为 8，输出层神经元个数 X 为 3，隐含层的神经元个数按公式 $H = (Y + X) \times 0.5 + C$ 公式选取，这里 C 是 1~10 之间的常数。如 $C=3$ 则 BP 网络模型结构为 8-9-3 型

式。此时个体长度按 $n = H \cdot (X + Y) + H + Y$ 计算得 $n=111$ 。网络训练最小均方误差为 $1e-3$ ，网络学习步长设为 0.05，训练次数为 7000，动量项系数为 0.9。

遗传算法参数设置为：种群规模 P 为 100，进化代数 T 为 200 代，实数编码方式，交叉概率 P_c 为 0.9，变异概率 P_m 为 $1/n$ (n 为决策变量的个数)，二进制交叉算子 $\eta_c = 20$ ，实参变异算子 $\eta_m = 20$ 。

5.2.3 试验结果与分析

为了验证多目标GA-BP算法的有效性，本文分别利用传统的BP神经网络和GA-BP去训练样本。训练BP网络时，BP权值和阈值通过随机函数来初始化；训练GA-BP网络时，通过GA算法训练过的最优个体来初始化权值和阈值。两种网络的收敛曲线如图5-6所示。

由网络曲线图可以明显看出，相同的训练参数和训练次数下，BP神经网络在训练6305次后终止。GA-BP网络在训练2753次终止。GA-BP网络较BP网络，明显提高了收敛精度，同时也加快了网络的训练次数。

为了验证输出模型的合理性，这里考虑6个工件需要在3台不同的机床上加工，工件加工路线与时间如表5-6，各工件的交货等信息见表5-7。

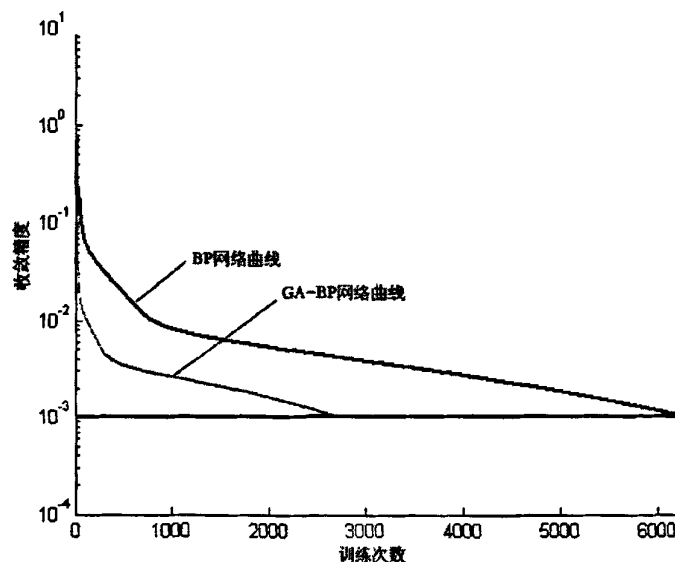


图 5-6 BP 网络与 GA-BP 的收敛曲线比较

Fig. 5-6 Comparison of convergence curves between BP network and GA-BP

表 5-6 各工件加工路线与时间

Table 5-6 The workpiece processing routes and time

工件	1	2	3	4	5	6
加工路线	M1-M2-M3	M3-M2-M1	M2-M1-M3	M1-M3-M2	M2-M3-M1	M3-M1-M2
加工时间/h	2-3-5	2-6-7	1-6-4	10-2-5	8-1-4	5-6-8

表 5-7 工件交货信息

Table 5-7 Workpiece delivery information

工件	1	2	3	4	5	6
交货期/h	17	21	22	30	18	21
惩罚系数	0.35	0.15	0.25	0.35	0.45	0.40
存储费用系数	0.20	0.15	0.45	0.35	0.25	0.15

这里在加工开始时，对于机床M1，工件1，4将要被加工，此时网络模型的输入为 $X=\{2, 0.548, 1.278, 0.35, 0.20, 0.8, 0.1, 0.1\}$ 网络输出 $Y=\{0.9014, 0.1002, 0.0948\}$ ；此时我们按最早交货期策略加工工件1；对于机床M2，工件3，5将要被加工，此时网络输入为 $X=\{2, 0.617, 0.586, 0.25, 0.45, 0.7, 0.1, 0.2\}$ ，网络输出为 $Y=\{0.1005, 0.9024, 0.0933\}$ ，此时选择等待加工时间最短的机床加工策略，此时选择工件3。由此可以看出网络输出的合理性。

5.2.4 结果讨论

遗传-神经算法充分利用了遗传算法的全局搜索能力和 BP 网络的局部搜索能力。在实验仿真中，通过分析车间作业主要生产状态参数和车间调度优化目标，得到最优的车间作业动态调度策略。通过结果分析可以看出本文提出的遗传-神经混合算法的有效性与可行性。同时，我们通过网络曲线图可以明显看出遗传-神经算法和纯 BP 算法性能上的差异。

5.3 本章小结

本文通过多个算例测试结果中的性能度量值及最终优化后的 Pareto 最优解的组成图看出，改进的并行混合非劣分类遗传算法无论在解的多样性方面还是解的收敛性方面都有较大改观，在多样性及收敛性之间达到了一个较好的平衡，避免了解过早收敛到局部 Pareto 解集，其性能明显优于 NSGA-II 算法。另外，本文成功地将 GA-BP 算法应用到多目标车间作业动态调度问题实例中，通过实验结果分析可以看出 GA-BP 算法的有效性与可行性。

结论

多目标优化一直是优化界中的一大难题,对该问题的研究一直富有挑战性和吸引力,然而研究多目标优化一直缺乏一种高效实用的求解方法。本文研究提出的基于遗传算法的 BP 神经网络算法,既利用了 BP 神经网络的通过样本学习完成任意空间映射的特点,又发挥了遗传算法可以对复杂的、非线性及不可微的函数实现全局搜索的优势,在解决多目标优化问题中,取得了良好的效果。本文的主要研究成果如下:

1. 通过对目前现有的几种多目标遗传算法的研究与分析,本文提出并实现了一个改进的并行混合非劣分类遗传算法(IMNSGA-II)。IMNSGA-II 是对 NSGA-II 的改进,即利用两种不同的选择机制并行进行进化操作,采用精英策略保存从上一代至今最优的非劣解,使得精英解的作用代数得以延续,为保持非精英解能参与到下一代的进化,我们限制了当前精英解的范围,保留部分非精英解,同时利用小生境的思想不断混入隔代因子,保持种群的多样性。多个算例测试结果中的性能度量值及最终优化后的 Pareto 最优解的组成图表明,该算法无论在解的多样性方面还是解的收敛性方面都有较大改观,在多样性及收敛性之间达到了一个较好的平衡,避免了解过早收敛到局部 Pareto 解集,其性能明显优于 NSGA-II 算法。

2. 提出一种解决 BP 网络过拟合的学习方法,通过对训练样本的调整和处理,以两个验证误差指标作为检验函数,来避免在实际应用过程中因被逼近样本的性质不能精确知道所引起的过拟合现象。

3. 在研究 BP 神经网络与遗传算法的基础上,将具有优良的逼近、分类和自学习功能的前馈神经网络和基于自然选择的遗传算法进行融合,建立一种多目标 GA-BP 算法,使用遗传算法在全局解空间内对网络拓扑结构和网络参数进行自适应的动态调整,从而获得网络的最优设计。另外,多目标遗传算法所求得的 Pareto 最优解集为设计决策提供了很大方便。将 GA-BP 算法应用到多目标车间作业动态调度问题实例中,通过实验结果分析可以看出 GA-BP 算法的有效性与可行性。

虽然本文提出的算法与传统进化算法相比,具有上述优良的特性,但是仍有许多不尽人意的地方。如当问题相当复杂时,计算工作量很大,计算时间较长。今后还可以在以下方面进一步展开深入的研究工作:

一方面,本文提出改进的遗传算法虽然提高了最优解的多样性和收敛性,

但对算法的效率并无明显改进。因此，仍可继续通过对其他遗传算子如并行选择机制的改进来提高算法的求解效率。

另一方面，在基于遗传算法的 BP 神经网络融合技术上的研究尚停留在比较试验结果层次，还没有形成一套完善的理论体系。算法的终止方面具有较强的人为因素，其科学性与合理性有待于进一步完善。

目前智能算法融合技术正处在发展阶段，从多目标优化的发展以及人类改造自然的优化原则不难看出，把智能算法应用在多目标优化领域的研究成果是令人瞩目的，开展进一步的研究具有深远意义。智能算法的融合技术为求解多目标优化问题提供了新的发展思路。当然算法的融合技术还有待进一步的研究学习，不断发展和完善，使其更加成熟。

参考文献

- [1] 崔逊学. 多目标进化算法及其应用[M]. 北京: 国防工业出版社, 2006: 4-6.
- [2] 谢涛, 陈火旺, 康立山. 多目标优化的演化算法[J]. 计算机学报, 2003, 26(8): 997-1003.
- [3] 王志良. 人工心理[M]. 北京: 机械工业出版社, 2007: 205-207.
- [4] 史忠植. 智能科学[M]. 北京: 清华大学出版社, 2005: 50-75.
- [5] 钟珞, 饶文碧, 邹承明. 人工神经网络及其融合应用技术[M]. 北京: 科学出版社, 2007: 69-90.
- [6] 陈国良, 王熙法, 庄镇泉, 等. 遗传算法及其应用[M], 北京: 人民邮电出版社, 1996: 38-57.
- [7] TOM M MITCHELL. 机器学习[M]. 曾华军, 张银奎, 等, 译. 北京: 机械工业出版社, 2003: 179-193.
- [8] BERGEY P K, RAGSDALE C T, HOSKOTE M. A Simulated Annealing Genetic Algorithm for the Electrical Power Districting Problem [J]. Annals of Operations Research, 2003, 121(1-4): 33-35.
- [9] 武兆慧, 张杜娟, 刘希玉. 基于模拟退火遗传算法的关联规则挖掘 [J]. 计算机应用, 2005, 25(5): 1009-1011.
- [10] 吕军, 冯博琴, 李波. 免疫遗传算法及其应用研究[J]. 微电子学与计算机, 2005, 22(6): 221-224.
- [11] WEI LINGYUN, ZHAO MEI. A Niche Hybrid Genetic Algorithm for Global Optimization of Continuous Multimodal Functions[J]. Applied Mathematics and Computation(New York), 2005 , 160(3): 649-661.
- [12] YUN YOUNGSU, GEN MITSUO. Performance Analysis of Adaptive Genetic Algorithms with Fuzzy Logic and Heuristics[J]. Fuzzy Optimization and Decision Making , 2003, 2(2): 161-175.
- [13] 邓莉, 鲁瑞华. 一种改进的抑制早熟收敛的模糊遗传算法[J]. 计算机科学, 2007, 34(11): 150-153.
- [14] ROSENBERG, RICHARD S. Simulation of Genetic Populations with Biochemical Properties[D]. Ann Harbor, Michigan: University of Michigan(Doctor), 1967: 15-38.

- [15] FONSECA C M, FLEMING P J. Genetic Algorithm for Multi-objective Optimization: Formulation, Discussion and Generalization[C]. In Proceedings of the 5th Int. Conf. on Genetic Algorithms, Sao Mateo, CA, 1993, 416-423.
- [16] SRINIVAS N, DEB K. Multi-objective Optimization Using Nondominated Sorting in Genetic Algorithms[J]. Evolutionary Computation, 1995, 2(3): 221-248.
- [17] HORN J, NAFLPLOTIS N, GOLDBERG D E. A Niche Genetic Algorithm for Multi-objective Optimization[C]. Proceedings of the 1st IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence. Piscataway, NJ: IEEE Service Center, 1994, 1: 82-87.
- [18] LI YANWEN, LI MENG, LU YINGHUA, et.al.. A New Text Detection Approach Based on BP Neural Network for Vehicle License Plate Detection in Complex Background [J]. Lecture Notes in Computer Science, 2007, 4492: 842-850.
- [19] LI SHOUYI, WANG RONGGUI, HU WEIWEI, et.al.. A New QPSO Based BP Neural Network for Face Detection[J]. Advances in Soft Computing, 2007, 40: 355-363.
- [20] 罗四维. 大规模人工神经网络理论基础[M]. 北京: 清华大学出版社: 北方交通大学出版社, 2004: 14-25.
- [21] CHEN WEIWEI, WEI XIAOPENG, ZHAO TINGTING. Product Schemes Evaluation Method Based on Improved BP Neural Network [J]. Lecture Notes in Computer Science, 2008, 5227: 99-106.
- [22] 李建珍. 基于遗传算法的人工神经网络学习算法[J]. 西北师范大学学报(自然科学版), 2002, 38(2): 33-37.
- [23] JINHUA XU. GA Optimized Wavelet Neural Networks[J]. Lecture Notes in Computer Science, 2006, 4099: 1176-1180.
- [24] 王勇, 蔡自兴, 周育人, 等. 约束优化进化算法研究及其进展[J]. 软件学报, 2007, 11(18): 2691-2706.
- [25] KUHN H W, TUCKER A W. Nonlinear Programming Proceedings of the Second Berkeley[J]. Symposium on Mathematical Statistical and Probability. U.California Press, 1951: 481-491.

- [26] 徐磊. 基于遗传算法的多目标优化问题的研究与应用[D]. 湖南: 中南大学(硕士学位论文), 2007: 2.
- [27] FONSECA C M., FLEMING P J. Genetic Algorithms for Multi-objective Optimization: Formulation, Discussion and Generalization[C]. In S. Forrest Ed., Proceedings of the Fifth International Conference on Genetic Algorithms , Ljniversity of Illinois at Urbana Champaign: Morgan Kaufman Publishers, 1993: 416-423.
- [28] RODRIGUEZ VAZQUEZ K, FONSEA C M, FLEETING P J . Multi-objective Genetic Programming: A Nilinear System Identification Application[C]. In J.R. Koza Ed. Late Breaking Papers at the Genetic Programming 1997 Conference, California Stanford Bookstore, 1997: 207-212.
- [29] TODD D S, SEN P A . Multiple Criteria Genetic Algorithm for Container-ship Loading[C]. In T Back Ed.. Proceedings of the Seventh International Conference on Genetic Algorithms. Michigan State University: Morgan Kaufman Publishers, 1997: 674-681
- [30] TAKAHAMA T, SAKAI S. Constrained Optimization by Applying the Constrained Method to the Nonlinear Simplex Method with Mutations [J]. IEEE Trans. on Evolutionary Computation, 2005, 9(5): 437-451.
- [31] RUNARSSON TP, YAO X. Search Biases in Constrained Evolutionary Optimization[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2005, 35(2): 233-243.
- [32] PASCOLETTI, P SERAFINI.DIFFERENTIAL. Conditions for Constrained Nonlinear Programming Via Pareto Optimization[J]. Journal of Optimization Theory and Applications, 2007, 134(3): 399-411.
- [33] PRADYUMN KUMAR SHUKLA. On the Normal Boundary Intersection Method for Generation of Efficient Front[J]. Lecture Notes in Computer Science, 2007, 4487: 310-317.
- [34] EL GHAZALI TALBI, SANAZ MOSTAGHIM, TATSUYA OKABE, et.al.. Parallel Approaches for Multiobjective Optimization[J]. Lecture Notes in Computer Science, 2008, 5252: 349-372.
- [35] HAMIDREZA ESKANDARI, CHRISTOPHER D. GEIGER. A fast Pareto genetic algorithm approach for solving expensive multiobjective optimization problems[J]. Journal of Heuristics, 2008, 14(3): 203-241.

- [36] VANVELDHUIZEN D A, LAMONT G B. Multiobjective Evolutionary Algorithms : Analyzing the State of the Art[J] . IEEE Transaction Evolutionary Computation, 2000, 8: 125-147.
- [37] 阎平凡, 张长水. 人工神经网络与模拟进化计算 [M]. 第二版. 北京: 清华大学出版社, 2005: 549-552.
- [38] DEB K, PRATAP A, AGARWAL S, et.al.. A Fast and Elitist Multi-objective Genetic Algorithm : NSGA-II[J] . IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197.
- [39] 江瑞, 罗予频, 胡东成, 等. 一种协调勘探和开采的遗传算法: 收敛性及性能分析[J]. 计算机学报, 2001, 24(12): 1233-1241.
- [40] SHAO YUXIANG, CHEN QING, ZHANG DONGMEI. The Application of Improved BP Neural Network Algorithm in Lithology Recognition [J]. Lecture Notes in Computer Science, 2008, 5370: 342-349.
- [41] 高隽. 人工神经网络原理及仿真实例[M]. 北京: 机械工业出版社, 2003: 10-15.
- [42] LUO ZHONG, LIU LISHENG, ZOU CHENGMING, et.al.. The Application of Neural Network in Lifetime Prediction of Concrete[J]. 2002, 17(1): 79-81.
- [43] 王向慧, 张国强, 连志春. 基于神经网络和 Pareto 最优的多目标联合优化算法[J]. 计算机应用, 2008, 28(10): 2517-2520.
- [44] 钟守楠, 蔡伟. 多目标拟神经优化方法[J]. 武汉水利电力大学学报, 2000, 33(5): 102-106.
- [45] 林广平, 黄卫, 方么生, 等. 基于 BP 神经网络的正交异性钢桥面系多目标结构优化设计[J]. 公路交通科技, 2005, 22(5): 83-86.
- [46] 李敏强, 徐博艺, 寇纪淞. 遗传算法与神经网络的结合[J]. 系统工程理论与实践, 1999, 17(4): 37-40.
- [47] ZOU CHENGMING, TONG QI WEI, YANG HONG YUN. A study on Neural Network Based on Contractive Mapping Genetic Algorithm[C]. Guo Qingping . International Sysposium on Distributed Computing and Application to Business, Wuhan, 2004: 777-779.
- [48] SCHAFFER J DAVID, WHITLEY DARRELL, ESHEL MAN LARRY-J. Combinations of Genetic Algorithms and Neural Networks: a Survey of the State of Art[C], Proceedings of International Workshop on Combinations

- of Genetic Algorithms and Neural Networks, Baltimore, 1992: 1-37.
- [49] KITANO H. Empirical Studies on the Speed of Convergence of Neural Network Training Using Genetic Algorithms[C]. Proceedings 8th JMIT National Conference in Artificial Intel-ligence. Boston MASS, 1990: 789-796.
- [50] 柴毅, 尹宏鹏, 李大杰, 等. 基于改进遗传算法的 BP 神经网络自适应优化设计[J]. 重庆大学学报(自然科学版), 2007, 30(4): 91-96.
- [51] TAMER F ABDELMAGUID. Permutation Induced Acyclic Networks for the Job Shop Scheduling Problem[J]. Science Direct, 2008, 33: 1560-1572.
- [52] 阳岁红, 鄢萍, 刘颖. 基于 BP 网络和模拟退火算法的多品种、小批量车间作业动态调度模型研究[J/OL]. 中国科技论文在线, 2008, http://www.paper.edu.cn/paper.php?serial_number=200812-1029.
- [53] JAYAMOHAN M S, RAJENDRAN C. New Dispatching Rules for Shop Scheduling: a Step Forward [J]. International Journal of Production Research, 2000, 38(3): 563-586.

攻读硕士学位期间发表的学术论文

- [1] 朱文龙, 丁华福. 遗传算法在多目标柔性 Job-Shop 调度中应用[J], 计算机技术与发展, 2009, 19(4)(已录用)

致谢

走过了两年多研究生生活，得到了许多老师和朋友的帮助和关怀，对自己有幸得到的这一切帮助、鼓励、关心，一直心存感激。

首先，感谢我的导师丁华福教授。在整个攻读硕士学位期间，从论文的选题、开题、撰写直至最后的答辩，都得到了导师的关心、帮助和指导。导师在学术上敏锐的洞察力、开阔活跃的学术思维、不懈进取的科研精神、严谨的治学态度、崇高的敬业精神、渊博的学识都给我留下深刻的印象，这些将使我终身受益。此外，导师为我提供了良好的学习和研究条件。谨此毕业来临之际，向关心和培养我的导师丁华福教授表示衷心的感谢和诚挚的敬意。

感谢我亲密的舍友，鄢章华、李智、蒋金华，和他们一起度过了二年多的快乐的时光，无论在学习方面还是在生活方面他们都尽力帮我排忧解难；此外感谢本实验室的同学，张中华、郭明、于水英、梁建中，在学习上给予我的帮助。我们共同营造的那份轻松与欢乐将被永远回忆与珍惜，衷心祝你们学业、事业有成！

最后，感谢论文的各位评审专家能在百忙之中抽出时间对我的硕士论文进行评审，并提出宝贵的建议，在此表示衷心的感谢！