

文章编号 1000-5013(2005)01-0093-04

遗传算法在制造资源优化配置中的实现

屈喜龙 孙林夫

(西南交通大学 CAD 工程中心, 四川 成都 610031)

摘要 资源优化配置是网络化制造系统中的一个重点,也是一个难点问题.文中分析传统算法在解决优化配置问题上的缺陷,提出一个基于遗传算法的资源优化配置算法模型.通过具体实例,说明该模型的有效性、先进性和可行性.最后,给出基于该模型的原型系统运行界面.

关键词 优化配置,遗传算法,网络化制造

中图分类号 TH 166 TP 391.7 F 224.31

文献标识码 A

研究区域网络化制造资源优化配置系统,利用网络技术实现资源共享和优势互补,是提高区域制造业企业整体竞争优势的重要举措,是以成德绵为核心的区域网络制造系统的核心内容之一.资源优化配置问题是组合优化问题中的一种.本文将结合成、德、绵网络化制造系统的开发实践,利用遗传算法对资源优化配置方法进行研究.

1 问题的提出与分析

网络化制造系统中的资源优化配置,是针对市场中的某一个机遇,借助于信息网络的支持,把地理上分散的、能力互补的企业资源连接起来.它们合作完成产品的设计、制造、装配,以及销售和服务整个生命周期,以最小的代价,获取最大的利益.这样,整个生产链中将有许多企业参加,各个企业在诸如设计、生产、销售和服务环节上贡献自己的核心力量.所以,如何合理地选择各个生产环节上的企业,完成资源的优化组合,就成了网络化制造资源优化配置的关键.文献[1]提出了一些伙伴关系管理原则,但只是限于定性的分析;而文献[2]虽然提出了一种设计有效动态联盟的两阶段定量框架,但这种方法存在有两个不足之处.(1)第1阶段利用DEA(数据包络分析)辨识出所谓的“有效”业务过程时,可能失去了一些尽管个体单元有效性不足但也许比较适合整个生产链的优化组合.这就造成了最终解可能是局部最优.(2)第2阶段利用(0~1)目标规划进行优化过程中,每个业务过程最多而且必须选一个企业来参加应用中,生产链的每个环节选多个企业来参加是可能的,而且是必要的.随着计算机技术的发展和遗传算法的深入研究,在解决搜索空间很大的组合优化问题时,遗传算法显得很有效^[3].文献[4]试图用遗传算法来解决这类问题,但它存在2个缺陷.(1)每种业务类型的合作伙伴必须且只能选择一个.(2)适应度函数的设计目标只有一个.但是实际运行过程中,不但要保证成本最小,也要保证完成时间最少.文献[5]采用遗传算法来解决资源优化配置问题,但它只能解决设备资源的优化配置.文献[6]的对资源优化配置的研究也仅限于对某个合作伙伴的优选,而不是针对产品全周期的.

2 算法设计

设有一个含有设计、采购、生产、装配和销售共5个业务过程的生产链,每个业务过程有若干个合作企业可以选择.整个生产链的业务过程是,设计中心设计好关键的产品部件后交给生产中心;生产出来

收稿日期 2004-08-09

作者简介 屈喜龙(1978-),男,博士研究生,主要从事网络化制造系统的研究. E-mail: quxilong@sina.com

基金项目 国家 863/CIMS 主题研究基金资助项目(2002AA414050; 2004AA414010)

后,连同采购中心所采购的配件一起交送给装配中心;装配好后提交给销售中心,进行销售.现假设,可以参与设计业务的企业伙伴有 3 家 ($D1, D2, D3$),可以参与配件采购业务的企业有 2 家 ($B1, B2$),可以参与生产的企业有 3 家 ($P1, P2, P3$),可以参与装配的企业有 2 家 ($C1, C2$),可以参与销售的企业有 2 家 ($S1, S2$).各个企业自身所需时间(t)、费用(S)数据,如表 1 所示.各企业集之间的连接费用和时间数据,如表 2 所示.表 2 对角线的上半部分表示各个企业之间的连接费用,下半部分表示连接时间.

表 1 各个企业自身所需时间和费用数据

企业	$D1$	$D2$	$D3$	$B1$	$B2$	$P1$	$P2$	$P3$	$C1$	$C2$	$S1$	$S2$
t/d	3.0	4.0	3.5	4.5	3.5	1.5	5.0	2.5	3.0	3.5	7.0	6.0
$S/\text{万元}$	7.6	5.4	7.3	5.9	5.6	6.4	6.1	8.3	9.6	8.7	4.4	4.8

表 2 各个企业之间的连接费用和连接时间数据

企业	$D1$	$D2$	$D3$	$B1$	$B2$	$P1$	$P2$	$P3$	$C1$	$C2$	$S1$	$S2$
$D1$	0	2.6	1.7	0.3	0.7	4.8	5.3	8.1	1.6	1.8	3.7	3.1
$D2$	3	0	4.5	0.8	0.5	5.5	10.0	1.7	1.5	3.5	1.8	1.0
$D3$	4.5	8.0	0	0.6	0.9	7.7	14.0	12.0	1.8	2.8	4.1	0.7
$B1$	4	10.0	8.5	0	4.1	4.1	2.9	1.7	0.6	0.7	3.1	3.5
$B2$	7.0	14.0	12.0	17.0	0	2.1	2.6	8.6	1.2	1.5	2.0	1.0
$P1$	14.0	10.0	5.0	7.0	13.0	0	7.2	1.3	5.9	4.0	3.9	3.4
$P2$	4.0	10.0	3.5	4.5	5.0	7.0	0	3.1	15.0	39.0	3.6	3.9
$P3$	10.0	5.0	5.5	10.0	7.5	11.0	2.0	0	14.0	10.0	4.6	2.3
$C1$	12.0	9.5	5.5	9.0	7.0	12.0	6.0	2.5	0	7.7	31.0	4.9
$C2$	10.0	12.0	18.0	14.0	16.0	7.0	13.0	3.5	7.0	0	21.0	4.6
$S1$	13.0	5.0	4.5	3.0	10.0	8.5	6.0	10.0	10.0	9.0	0	18.0
$S2$	10.0	19.0	25.0	12.0	14.0	5.5	7.5	13.0	4.0	6.5	3.5	0

2.1 编码

本文采用遗传算法的编码方式为二进制编码.因为在网络化制造系统中,采用这种编码有 5 点优势. (1) 简单. 每 1 个合作伙伴可以很自然地用一串二进制代码表示. 1 表示选中, 0 表示没选中, 代码串的长度取决于所考察的候选企业的个数. (2) 完备. 问题空间中的所有点(侯选解)都能作为遗传算法空间中的染色体表现. (3) 健全. 遗传算法空间中的染色体能对应所有问题空间中的侯选解. (4) 非冗余. 染色体和侯选码一一对应. (5) 采用了二进制编码. 就可以采用很成熟的交叉、变异算子. 设所有的企业个数为 L , 那么, 代码长 L 位, 0 表示相应的企业没被选中参与, 而 1 表示被选中. 如 1101010110011 表示 $D1, D2, B1, P1, P3, C1, S2$ 和 $S3$ 被选中.

2.2 适应度函数的构造与选择方式

通常, 时间(T)、成本(C)、质量(Q)、服务(S)是企业 在市场中竞争中成功的关键因素. 网络化制造系统也离不开这个准则. 所以, 对于整个生产链的资源优化配置, 适应度函数的设计目标是: 整体运行时间最小和整体运行成本最少.

令 C_{ik} 表示第 i 种业务类型的中的第 k 个企业完成业务所需要的时间, C_{ik} 表示第 i 种业务类型中的第 k 个企业完成业务所需要的费用. $CONT_{ikjl}$ 表示第 i 种业务类型的中的第 k 个企业与第 j 种业务类型中的第 l 个企业之间所需的连接时间(如运输时间); $CONC_{ikjl}$ 表示第 i 种业务类型中的第 k 个企业与第 j 种业务类型中的第 l 个企业之间所需的连接成本(如运输费用). T 为整个产品生命周期中所要求的完工时间么

$$(ik, j1) = \begin{cases} 1, & \text{第 } i \text{ 种业务类型的第 } k \text{ 个企业与第 } j \text{ 种业务类型的第 } l \text{ 个企业被选中,} \\ 0, & \text{其它情况.} \end{cases} \tag{3}$$

$$(ik) = \begin{cases} 1, & \text{如果第 } i \text{ 种业务过程的第 } k \text{ 个企业被选中,} \\ 0, & \text{其它情况.} \end{cases} \tag{4}$$

如果以成本最小为目标函数, 那么有 $\min C = \sum (ik) C_{ik} + \sum (ik, jl) CONC_{ikjl}$; 如果是 以总的时间为目标函数, 那么有 $\min T = \sum (ik) C_{ik} + \sum (ik, jl) CONC_{ikjl}$, 且满足约束条件 $\forall (ik) > 0$. (ik) 为第 i 种业务类型的第 k 个企业, 即每 1 种业务类型至少要选一个参与. 适应度函数 $f(t) = \frac{1}{T} \times C(t) + \frac{1}{C} \times$

$T(t)$, 其中 t 为遗传算法中的代数, T , C 根据对产品的时间和费用要求的程度不同而取的不同权值. 考虑到 $C(t)$ 与 $T(t)$ 可能数量级不同, 会造成低数量级的那个因素对整个适应度影响被减低. 即 $f(t) = T \times C(t) / E_C + C \times T(t) / E_T$, E_C , E_T 分别表示整个生产链运行的最低期望值. 本文采用最基本的轮盘选择法. 采用轮盘选择法最到的优点是简单容易实现. 因为我们的目标是追求的是最小值, 所以在进行选择操作时, 必须选取适应度 $f(t)$ 值尽量小的染色体进入下一代.

2.3 交叉和变异与终止条件

本文采用单点交叉, 交叉概率选 0.6, 变异概率选 0.1, 以 50 代作为终止条件. 文献 [7] 中的类似问题的参数选取已进行了详细的论证.

3 系统实现与运行结果

当种群大小为 8, T 为 0.6, C 为 0.4, E_C 为 100, E_T 为 60 时, 可得到最优解. $f(010101010101) = 0.223$. 本实例是在 Matlab 6.5 的遗传算法工具箱的基础上实现的. 这样, $D2$, $B1$, $P1$, $P3$, $C2$ 和 $S2$ 被选中, 从而最优地对网络化资源进行配置. 当然, 可以根据实际情况, 对时间和费用的权值进行调整. 网络化制造资源优化配置是成、德、绵区域网络化制造系统的核心部分之一, 基于本遗传算法模型的原型系统已经实现. 系统基于 .NET 平台开发的, 用 Visual Studio .NET 作为开发工具, 开发语言为 C#, 数据库采用 SQL SERVER 2000. 采用 N 层架构, 分为 4 层实现, 其关键函数为

```
private binary[] init(int seedgroupsize,int chromlength)
{ // 初始化种群, seedgroupsize 为种群大小, chromlength 为编码长度
private float[][] gettimedate(binary[] seedgroup)
{ // 根据染色体编码, 从数据库中获取时间的相关数据, 以供 calfitvalue 使用
private float[][] getcostdate(binary[] seedgroup)
{ // 根据染色体编码, 从数据库中获取成本的相关数据, 以供 calfitvalue 使用
private float[] calfitvalue(binary[] seedgroup)
{ // 计算种群中的每个染色体的适应度值
private void select(binary[] seedgroup, string selectmethod)
{ // 根据 calfitvalue 计算的适应度值, 采用某种选择算法选择染色体进入下一代, 把被选中的染色体依然保存在 seedgroup 中
private void cross(binary[] seedgroup, string crossmethod, float crossratio)
{ // 对选择后的种群采用单点或多点交叉法进行交叉运算, 交叉概率为 crossratio 把进入下一代的染色体依然保存 seedgroup 中
private void mutate (binary[] seedgroup, float mutateratio)
{ // 对交叉过后的种群, 进行变异. 变异的概率为: mutateratio, 把进入下一代的染色体依然保存 seedgroup 中. 优化配置的实现过程
private void Bt _ Configure _ Click(object sender, System. EventArgs)
{
    binary[] seedgroup = init(8,12) ; // 种群大小为 8, 编码长度为 12
    int i = 1 ; 记录进化的代数
    while ( i < 50)
    {
        float[][] timedata = gettimedate(seedgroup) ; float[][] costdata = getcostdate(seedgroup) ;
        float[] fitvalue = calfitvalue(seedgroup) ; // 根据 gettimedate, getcostdate 得到的数据, 计算种群中每个染色体的适应度
        select(seedgroup, turntable) ; // 根据 calfitvalue 的结果, 根据轮盘选择法进行选择
        cross(seedgroup, singlepoint, 0.60) ; // 对种群进行单点交叉运算, 交叉概率为: 0.6
        mutate(seedgroup, 0.10) ; // 对种群进行变异运算, 变异概率为: 0.1
        i++ ; // 进化的代数增 1 代
    }
}
```

其运行的真实界面 ,如图 1 所示.



图 1 原型系统的运行界面

4 结束语

资源优化配置是成德绵区域网络化制造系统近期实施的重点. 基于本文提出的算法模型开发的原型系统不但运行稳定 ,而且取得很好的实际效果. 通过实例的数据结果可以看出 ,该模型克服了传统方法的缺陷 ,对生产链的每个业务过程没有只能选一个的限制 ,也不会因为识别所谓有效的业务过程而漏掉有效的最优组合. 在描述和表达问题时 ,清晰直观 ,而且具有全局的搜索能力 ,对类似的问题求解亦具有可借鉴性.

参 考 文 献

1 Zhan Su , Poulin D. Partnership management with in the virtual enterprise in a network[J] . IEMC , 1996 : (1) , 645 ~ 650
2 Talluri S , Baker R C . A quantitative framework for designing efficient business process alliances[J] . IEMC , 1996 : (1) 656 ~ 661
3 王小平 ,曹立明. 遗传算法理论、应用、与软件实现[M]. 西安 :西安交通大学出版社 ,2002. 6 ~ 97
4 李培根 ,张 洁. 敏捷化智能制造系统的重构与控制[M]. 北京 :机械工业出版社 ,2003. 79 ~ 96
5 马雪芬 ,戴旭东 ,孙树栋. 面向网络化制造的制造资源优化配置研究[J] . CMIS ,2004 ,10 (5) :523 ~ 527
6 赵 刚 ,江平宇. 网络化制造中资源优化配置的方法研究[J] . 新工艺新技术新设备 ,2003 ,33 (8) :60 ~ 64
7 冯尉东. 基于遗传算法的动态联盟伙伴选择过程及优化模型[J] . 清华大学学报 (自然科学版) ,2000 ,40 (10) :120 ~ 124

Implementation of Genetic Algorithm to the Optimal Configuration of Manufacture Resources

Qu Xilong Sun Linfu

(CAD Engineering Center , Southwest Jiaotong University , 610031 , Chengdu , China)

Abstract Optimizing configuration of resources is a focal point and also a difficult point in networking manufacture. After analysing the shortcomings of conventional algorithm in settling optimizing configuration , the authors propose an algorithmic model of optimizing configuration of resources which is based on genetic algorithm ; and illustrate validity , advancement and feasibility of this model by specific examples ; and finally , give an interface for the operation of prototypc system based on this model.

Keywords optimizing configuration , genetic algorithms , networking manufacture