

文章编号: 1674-9057(2020)02-0409-06

doi:10.3969/j.issn.1674-9057.2020.02.022

基于改进遗传算法的电子产品 CTO 订单推荐

韩海峰, 叶恒舟, 张 路

(桂林理工大学 广西嵌入式技术与智能系统重点实验室, 广西 桂林 541006)

摘 要: 电子产品涉及的配件较多, 每种配件均有多种品牌与型号, 给 CTO 订单的确定带来较大的困难。针对上述问题, 以功能定位目标贴近度为优化目标, 加工成本和产品功耗为约束条件, 建立了一种电子产品 CTO 订单推荐(CTO order recommendation, CTOR)模型, 并提出一种改进的遗传算法来求解该模型。仿真结果验证了改进算法的有效性: 自适应遗传算子提高了算法的收敛精度和速度, 而种群的多样性维护策略在一定程度上降低了算法陷入局部最优的可能性。

关键词: CTO 订单推荐; 自适应; 遗传算法; 种群多样性

中图分类号: TP301; TP18

文献标志码: A

0 引 言

按订单配置(configure to order, CTO)起源于用户需求与偏好, 目标是快速、有效地满足用户的个性化需求, 提升企业的核心竞争力。产品配置是实现 CTO 的关键, 配置过程开展的前提是准确地获取客户需求。由于定制产品的复杂性不断提高, 加上客户对产品结构、性能等认识不充分, 往往仅能提出与自身体验和偏好相关的需求, 因此具有模糊性和不确定性^[1]。特别是对于电子产品, 其物料清单(bill of material, BOM)往往包含数十项核心配件, 每个配件有诸多品牌与型号可供选择, 且产品更新较快, 若采用过程式产品配置, 需要用户对这些配置非常熟悉。在 BOM 选择过程中, 难以满足全局性需求, 也易忽略企业库存等影响因素。因此, 有必要根据用户提出的局部和全局约束以及功能性需求, 为客户推荐产品配置清单, 帮助用户明确满足自身需求同时兼顾企业利益的 CTO 订单。

在需求获取方面, 但斌等研究了一种需求智能获取方法, 该方法关注用户表达需求的异质性,

提供与用户的个人特征和使用情景相适应的需求表达方式, 从而帮助用户提高需求描述的客观性和准确性^[2]。Carulli 等通过建立产品虚拟样机和虚拟现实环境, 在早期帮助客户直观感知产品, 以辅助用户与专业设计人员沟通^[3]。

以客户需求与企业资源为驱动, 研究产品配置模型及求解算法是一个研究热点。传统 CTO 产品平台仅能满足已知的客户需求, 文献 [4] 提出面向订单设计(engineering-to-order, ETO)的配置方法, 在维持个性化需求的同时增强了设计重用性, 提升了产品配置的适应能力; 文献 [5] 在 CTO 和 ETO 模型的基础上, 基于适应性开放体系结构产品平台, 以个性化自行车配置过程为例, 提出了一种个性化分布式产品配置过程; 文献 [6] 基于 ETO, 从产品系列、产品部件、产品细节 3 个层面关注产品配置的价格与利润关联; 文献 [7] 考虑了订单的紧急程度, 以综合收益最大化为目标, 将产品配置方案建模为一个多目标优化问题, 并采用蚁群优化算法求解, 该方法认为订单已经存在, 关注的是大规模个性化订单的整体调度问

收稿日期: 2019-11-11

基金项目: 国家自然科学基金项目(51365010); 广西科技重大专项(桂科 AA19046004); 广西嵌入式技术与智能系统重点实验室项目(2019-01-10)

作者简介: 韩海峰(1995—), 男, 硕士, 研究方向: 智能算法、智能计算, mirror_han@126.com。

通讯作者: 叶恒舟, 博士, 教授, 18123408@qq.com。

引文格式: 韩海峰, 叶恒舟, 张路. 基于改进遗传算法的电子产品 CTO 订单推荐[J]. 桂林理工大学学报, 2020, 40(2): 409-414.

题;文献[8]探讨了多情境及需求的不确定性情况下大规模定制订单的优化配置问题。考虑到为用户定制一些复杂产品繁琐费时,文献[9]在产品配置过程中,采用基于基尼指数的问答导航属性选择策略,帮助设计人员更好地获取用户需求与偏好;文献[10]讨论了云制造环境下个性化产品配置,重点是基于关联规则挖掘的产品初步配置以及基于蚁群算法的产品完整配置;文献[11]在产品配置时重点关注了用户的感性需求;文献[12]也根据用户需求为用户推荐产品配置结果,将其应用到多功能液压千斤顶的配置设计,但它强调的是规则匹配,而所采用的交互式遗传算法需要用户来评价种群个体适应度,交互过程复杂冗长。

综上所述,现有关于 CTO 产品配置的研究大多是考虑如何在用户个性化需求及生产成本的基础上,将大量的 CTO 订单进行整合获取 BOM,而忽略了 CTO 订单本身如何根据用户的个性化需求而明确化。针对这一问题,本文根据用户的个性化需求描述,通过一种半交互过程帮助用户高效确定 CTO 订单,其核心是构建一个针对电子产品的 CTO 订单推荐(CTO order recommendation, CTOR)模型,并采用一种支持自适应多样性维护特性的改进遗传算法(adaptive diversity maintenance of genetic algorithms, ADM-GA)求解该模型。

1 电子产品 CTOR 模型

设某电子产品的 BOM 包含 n 个配件,每个配件有若干个品牌,每个品牌有若干个型号可供选择。CTO 订单推荐就是要为每个配件选择某个品牌下的某个型号,以满足用户的个性化需求,并兼顾企业库存等信息。

用户的个性化需求可以归结为 3 类:1)功能定位性需求,如定位为商务型;2)针对配件或可分解为针对配件的局部需求,如内存不小于 4 G、CPU 是 Intel、配色为银白等;3)针对全部或多个配件的全局约束,如对价格、功耗、重量等的要求。由于第 2 类的局部需求只需要对各个配件的候选集作筛选,而企业库存则可以通过调整配件价格来体现(如对于库存中没有的配件适当提高价格),因此,要满足用户的个性化需求,可以考虑以功能定位目标贴近度为优化目标,价格及功耗等为

约束条件构建配件清单推荐模型,并通过求解该模型为用户推荐满足其个性化需求的订单。

1.1 功能定位目标贴近度

假设描述某种功能定位涉及 r_1, r_2, \dots, r_u 等 u 个指标,指标 $r_i(i=1, 2, \dots, u)$ 的既定目标属于 $\{a_i, \dots, b_i\}$ 。设指标 r_i 的可选集 $CS_i = \{v_{i1}, v_{i2}, \dots, v_{id}\}$,即该指标可以有 d 个不同的选择值(或区间),而用户为该指标选择值(或区间)为 v_i ,记集合 $SS_i = \{v | v \in CS_i \wedge ((v_i \leq v \wedge v < a_i) \vee (b_i < v \wedge v \leq v_i))\}$,则用户对指标 r_i 的选择的目标贴近度 μ_i 可描述为

$$\mu_i = 1 - \frac{|SS_i|}{|CS_i|}, \tag{1}$$

其中, $|SS_i|$ 、 $|CS_i|$ 分别表示集合 SS_i 、 CS_i 中元素的个数。

例如,设商务手机的标准 GPU 型号定位在集合 $\{530, 540\}$ 中,而 GPU 型号可选集为 $\{430, 530, 540, 630, 640\}$ 。若用户选择的型号为 430,则 SS 集为 $\{430\}$,GPU 功能定位目标贴近度为 $1-1/5=0.8$;若用户选择为 530,则 SS 集为空,GPU 功能定位目标贴近度为 1;若用户选择 640,则 SS 集为 $\{630, 640\}$,GPU 功能定位目标贴近度为 $1-2/5=0.6$ 。

总体的功能定位目标贴近度 μ 为

$$\mu = \sum_{i=1}^u w_i \cdot \mu_i, \tag{2}$$

其中, u 为产品功能定位目标贴近度的指标个数; w_i 为指标 r_i 的权重, $\sum_{i=1}^u w_i = 1$ 。

1.2 CTOR 模型

设某电子产品的选配清单涉及 n 个配件,第 i 个配件有 p_i 种选择(每种选择对应某个品牌下的某个型号), $x_{ij}(i=1, 2, \dots, n; j=1, 2, \dots, p_i)$ 为 0-1 变量, $x_{ij}=1$ 表示第 i 个配件作出第 j 种选择,则可以将该电子产品的 CTOR 模型构建为如下的多维多选择背包问题(multidimensional multiple-choice knapsack problem, MMKP)^[13]:

目标: $\max \mu$

约束条件:

$$prcl \leq \sum_{i=1}^n \sum_{j=1}^{p_i} x_{ij} \cdot pr_{ij} + pr_0 \leq prcu; \tag{3}$$

$$\sum_{i=1}^n \sum_{j=1}^{p_i} x_{ij} \cdot pc_{ij} \leq pcc; \tag{4}$$

$$v_r = f_r(x_{ij}), r = r_1, r_2, \dots, r_u; \tag{5}$$

$$\sum_{j=1}^{p_i} x_{ij} = 1, i = 1, 2, \dots, n; \quad (6)$$

$$x_{ij} \in \{0, 1\}, i = 1, 2, \dots, n; j = 1, 2, \dots, p_i. \quad (7)$$

其中, 式(3)描述价格约束, pr_{ij} 为第 i 个配件的第 j 种选择的价格, pr_0 为加工等成本, 认为是一个常数, $[prcl, prcu]$ 为用户约定的产品价格区间; 式(4)描述功耗约束, pc_{ij} 为第 i 个配件的第 j 种选择的配件功耗(有些配件的功耗很小, 可以忽略不计, 即取值为 0), pcc 为用户约定的产品功耗上限; 式(5)描述根据用户的选择计算功能定位中某种指标值的方法。

2 改进遗传算法求解 CTOR 模型

CTOR 模型是一个 MMKP 问题, 诸如蚁群算法^[14]、粒子群优化算法^[15] 等进化算法可以有效求解这类问题。其中遗传算法^[16-17] 具有分布式的全局搜索能力, 对于多维组合优化问题具有良好的应用前景, 不足在于收敛速度较慢以及存在局部收敛问题, 通过调整遗传算子、构建多样性维护策略可以改善上述问题。

2.1 简单遗传算法

简单遗传算法(simple genetic algorithm, SGA) 主要涉及编码、设定初始种群、适应度函数、种群遗传操作(选择、交叉、变异)等核心要素。种群个体采用实数编码方式。染色体中的每个基因位对应一种配件, 基因的编码值代表该配件的选项序号(按照价格参数排序)。种群的初始化随机产生, 仅保留满足约束条件的个体, 初始化种群个体的数量为 N 。个体的适应度由功能定位目标贴近度确定。种群选择采用回放式等概率采样的方式, 即轮盘赌。第 i 个个体被选中的概率为

$$p_i(t) = \frac{f_i(t)}{\sum_{i=1}^N f_i(t)}, i = 1, 2, \dots, N. \quad (8)$$

其中, $f_i(t)$ 为第 i 个个体的适应度。

SGA 中交叉变异最主要的两点: 概率和方式。种群的交叉、变异概率为固定值(p_c, p_m)。交叉和变异方式均采用多点形式。多点交叉中, 每个基因座的交叉起始点设在编码的起始(上一个基因座的末尾), 交叉结束点在该基因座编码的末尾, 每个个体的基因座都以概率 p_c 发生交叉行为。变异方式类似交叉。

以实数编码为例, 染色体 A 的基因编码为[03 11 32 08 13 26], 染色体 B 的编码为[19 25 07 15 02 22], 发生交叉互换的位置分别发生在基因座 1、4、6 上, 那么交叉过后的新染色体 A 编码为[19 11 32 15 13 22], 新染色体 B 编码为[03 25 07 08 02 26], 如图 1 所示。多点变异类似于多点交叉, 若图 1 中染色体 A 的基因座 1、4、6 发生变异, 则新染色体 A 的编码为[** 11 32 ** 13 **], ** 为从相对应的基因座编码空间中随机选取的编码。

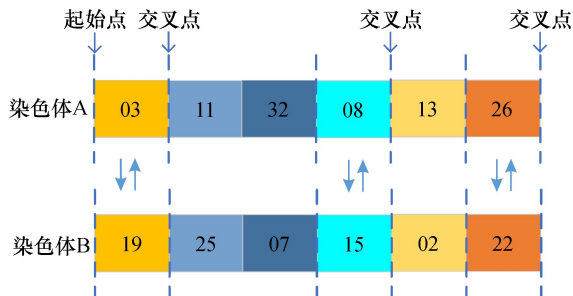


图 1 染色体交叉

Fig. 1 Chromosome crossings

2.2 自适应遗传算子设计

在 SGA 中, 遗传算子(选择算子、交叉算子、变异算子)是影响算法收敛速度的重要因素, 本文主要在交叉算子、变异算子上采用自适应变化策略。

2.2.1 交叉算子设计 在 SGA 中, 任意两个父代染色体的交叉变异概率是相同且固定的。如果采用差异化的交叉变异概率, 即父代染色体的适应度越高, 它们交叉变异的概率越低, 有望使得适应度高的染色体更多地保留在下一代种群中, 从而提高新种群的总体适应度。另一方面, 对于适应度较低的染色体, 在迭代前期的交叉变异概率应比迭代后期大; 而对于适应度较高的染色体, 在迭代前期的交叉变异概率应比迭代后期小。如此, 在迭代前期可以扩大搜索范围, 提升种群多样性, 以避免早收敛; 而在迭代后期, 可以避免破坏种群收敛, 提升寻优效果。基于以上理念, 以种群的适应度均值为参照, 设计了确定交叉算子的计算式:

$$p_{ij}(t+1) = \left(1 - \beta \cdot \frac{f_i(t) + f_j(t) - 2\bar{f}(t)}{\bar{f}(t)}\right) \cdot p_c, \quad (9)$$

其中, $p_{ij}(t+1)$ 为两个父代染色体的等位基因发生交叉的概率; p_c 为标准遗传的交叉概率; $f_i(t)$ 、 $f_j(t)$ 分别为选择交叉的两个父代染色体的

适应度； $\overline{f(t)}$ 为父代种群的适应度均值； β 为可调系数。

2.2.2 自变异算子设计 类似于交叉算子的设计理念，由第 i 个染色体的自变异概率为

$$m_i(t+1) = \left(1 - \gamma \cdot \frac{f_i(t) - \overline{f(t)}}{\overline{f(t)}}\right) \cdot p_m, \quad (10)$$

其中， p_m 为标准遗传的变异概率； γ 为可调系数。

2.3 种群多样性检测与维护

遗传算法中的多样性可从染色体或基因多样性角度考虑。维持种群多样性是降低种群陷入局部最优的有效策略^[18-19]。为避免破坏其收敛，每隔 10 代进行一次种群多样性检测，并根据检测结果判定是否进行维护。

第 t 代种群的多样性 $D(t)$ 可度量为

$$D(t) = \frac{\sum_{i=1}^l d_i(t)}{N \cdot l}, \quad (11)$$

其中， $d_i(t)$ 表示 t 代中所有个体的第 i 位上的等位基因出现频率最多的基因的次数； N 为种群中染色体的个数； l 为染色体的基因串的长度。这种度量方式是从基因多样性的角度测量种群多样性。假设存在种群矩阵

$$M_{4 \times 4} = \begin{bmatrix} 5 & 5 & 1 & 2 \\ 3 & 6 & 5 & 2 \\ 3 & 7 & 5 & 2 \\ 4 & 3 & 5 & 2 \end{bmatrix}. \quad (12)$$

矩阵中的每一行代表一个个体，每一列代表相同基因座上的等位基因。因为所有第 1 位等位基因中，基因编码相同的最大数量是 2（基因编码为 3 的数量最大），故 $d_1(t) = 2$ ；类似可知， $d_2(t) = 1$ 、 $d_3(t) = 3$ 、 $d_4(t) = 4$ 。

由种群的多样性 $D(t)$ 和 $g/step(g, step)$ 分别为当前迭代次数、目标总迭代次数共同确定种群更替占比 ρ （表 1）。每次迭代后需要更新的个体数为 $\varepsilon \cdot \rho \cdot N$ （ ε 为可调参数）。这些个体以一种新轮盘赌的方式选取，并向种群中添加新生成的个体：

$$p_k(t) = \frac{f_{best}(t) - f_k(t)}{f_{best}(t)}, \quad (13)$$

式中， $p_k(t)$ 表示 t 代种群中第 k 个染色体的中间概率； $f_k(t)$ 表示 t 代种群中第 k 个染色体的适应度； $f_{best}(t)$ 表示 t 代种群中最佳的个体适应度。然后，利用中间概率作为染色体被选中的概率，进行轮盘赌选择。

2.4 求解 CTOR 模型

为了快速有效地求解 CTOR 模型，针对 SGA 收敛速度慢、收敛精度不足等缺点，进行了以下改进：①采用 2.2 节的方法设计自适应交叉算子与自变异算子，以提升遗传进化速度；②采用 2.3 节的方法加入种群多样性检测与维护机制，以保持种群多样性，缓解局部收敛问题。图 2 为 ADM-GA 求解 CTOR 模型的流程。

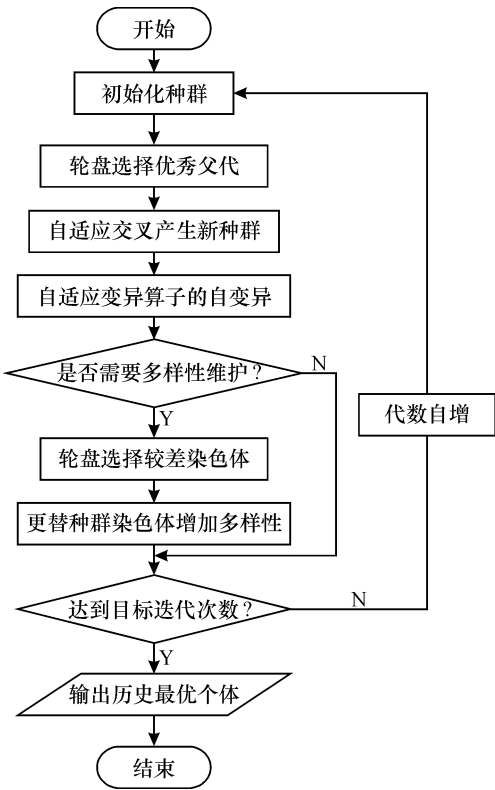


图 2 ADM-GA 算法求解 CTOR 模型流程

Fig. 2 Solving CTOR model process with ADM-GA algorithm

表 1 种群更替占比 ρ 的确定

Table 1 Determination of population replacement percentage ρ

$g/step$	$D(t)/\%$							
	0~30	30~40	40~50	50~60	60~70	70~80	80~90	90~100
0~0.3	0	0.05	0.1	0.1	0.2	0.2	0.3	0.4
0.3~0.5	0	0	0.05	0.1	0.1	0.2	0.2	0.3
0.5~0.8	0	0	0	0.05	0.05	0.1	0.1	0.1
0.8~1.0	0	0	0	0	0	0	0	0

3 实验仿真与分析

3.1 实验环境与参数取值

实验 PC 机主要配置为：Intel(R) Core(TM) i7-9750H、2.6 GHz CPU、16 GB 内存，64 位 Windows 10 操作系统，编程语言为 Java 1.7。实验时，考虑 PC 机的个性化订单问题。从公开的网络平台爬取了涉及显示器、CPU、GPU、主板、内存、硬盘、电源、键盘、鼠标、耳机等 10 个配件共计约 348 条记录，其中单个配件的最小记录条数为 23 条，最高为 52 条。

经过多次参数调节和统计数据发现，采用如表 2 所示的参数设置时，实验效果最佳。

表 2 最佳参数设置组合								
Table 2 Parameter settings								
n	N	l	$w_i(i = 1, 2, \dots, l)$	p_m	p_c	γ	β	ε
10	40	10	0.1	0.1	0.6	20	1	1

3.2 实验结果分析

实验对比了标准遗传算法(SGA)、在 SGA 的基础上添加 2.2 节设计的自适应遗传算子后的遗传算法(DGA)、本文算法(在 DGA 的基础上加入 2.3 节多样性检测与维护策略后的遗传算法，记为 ADM-GA)3 种算法的性能。为提升实验数据的可靠性，实验结果为测试 50 次后的各相关指标数值(适应度、遗传代数、迭代时间)的均值或标准差。

图 3 对比了各代种群的历史最佳适应度。可见，3 种算法在迭代前期(<200 代)都能较快地趋近最优解，200 代之后逐渐缓慢。在到达相同的迭代次数时，ADM-GA 比 SGA 和 DGA 可以获得更好的适应度。

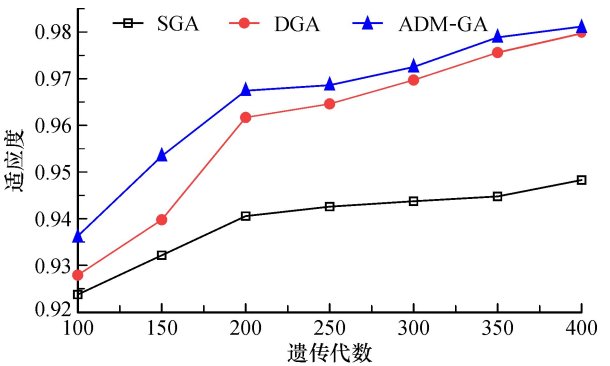


图 3 3 种算法的历史最佳适应度值

Fig. 3 Historical optimal fitness values of the three algorithms

图 4 对比了 3 种算法所获得的最佳适应度的标准差。可见，标准差越小，稳定性越好。3 种算法的稳定性总体上为：ADM-GA>SGA>DGA。DGA 的稳定性弱于 SGA，是因为引入自适应交叉和自变异算子后，虽然在收敛速度上有所加快，但更容易陷于局部收敛；而 ADM-GA 在 DGA 的基础上，加入了多样性检测与恢复机制，较好地克服了局部收敛问题，稳定性较好。

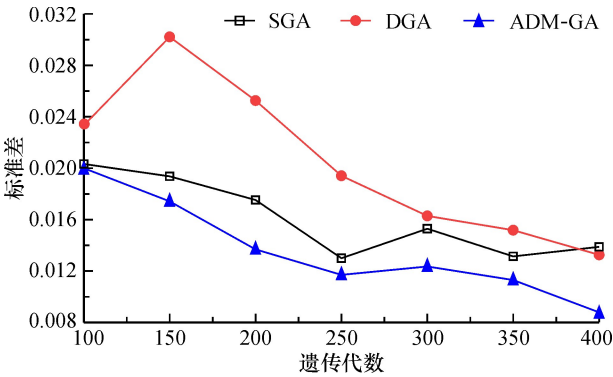


图 4 3 种算法的适应度标准差

Fig. 4 Fitness standard deviation of the three algorithms

表 3 对比了 3 种算法达到相应的目标适应度值时所需要的遗传代数及时间开销。其中，时间开销仅统计了种群初始化、选择、交叉、变异、适应度计算、多样性检测与维护等操作，而没有统计编码解码的时间开销(因为该过程的时间开销较大，且 3 种算法所花时间是相同的)。可见，3 种算法的时间开销均随着目标适应度的增加而增加，但 DGA 与 ADM-GA 的增速比 SGA 缓慢很多；达到同一目标适应度，ADM-GA 所需要的迭代次数与时间开销均是最低的，DGA 次之，SGA 最高。

表 3 3 种算法迭代次数及时间开销						
Table 3 Iteration times and time cost of the three algorithms						
目标适应度值	SGA		DGA		ADM-GA	
	次数	时间/ms	次数	时间/ms	次数	时间/ms
0.93	180	254	133	187	86	122
0.94	253	325	158	221	116	174
0.95	602	781	189	259	156	233
0.96	1 529	1 970	225	322	202	280
0.97	4 230	5 337	282	386	245	343
0.98	44 894	55 430	370	510	313	412

4 结 论

确定电子产品的 CTO 订单是实现个性化设计与生产的重要基础，其目标是根据用户的个性化

需求确定产品所涉及的每个配件的品牌与型号。CTO 订单推荐可辅助用户与专业设计人员沟通,以快速、准确地确定 CTO 订单。通过形式化度量功能定位目标贴进度指标,电子产品 CTO 订单推荐被建模为一个多维多选择背包问题。为了克服经典遗传算法存在局部收敛及收敛速度较慢等不足,在用遗传算法求解该问题时,提出了新的遗传算子自适应调节方法及种群多样性检测与维护策略。当前的 CTO 订单推荐,仅从用户的角度进行了优化,同时从用户与生产商角度(如优先利用现有库存、推荐和优化现有生产线加工产品等)进行优化,是下一阶段的研究方向。

参考文献:

- [1] 王建,黎荣,武浩远,等. 按单设计型复杂产品的配置设计技术综述 [J]. 计算机集成制造系统, 2020, 26 (2): 410-425.
- [2] 但斌,经有国,孙敏,等. 在线大规模定制下面向异质客户的需求智能获取方法 [J]. 计算机集成制造系统, 2012, 18 (1): 15-24.
- [3] Carulli M, Bordegoni M, Cugini U. An approach for capturing the voice of the customer based on virtual prototyping [J]. Journal of Intelligent Manufacturing, 2013, 24(5): 887-903.
- [4] Levandowski C E, Jiao J R, Johannesson H. A two-stage model of adaptable product platform for engineering-to-order configuration design [J]. Journal of Engineering Design, 2015, 26 (7-9): 220-235.
- [5] Zheng P, Xu X, Yu S Q, et al. Personalized product configuration framework in an adaptable open architecture product platform [J]. Journal of Manufacturing Systems, 2017, 43: 422-435.
- [6] Haug A, Shafiee S, Hvam L. The costs and benefits of product configuration projects in engineer-to-order companies [J]. Computers in Industry, 2019, 105: 133-142.
- [7] 杜浩明,张爱新,阮文华,等. 大规模定制中产品配置的蚁群优化算法 [J]. 上海电机学院学报, 2012, 15 (3): 177-183.
- [8] 陈淮莉,李丹妮,郑建华. 大规模定制型企业需求不确定下订单优化配置 [J]. 上海海事大学学报, 2013, 34 (4): 71-75.
- [9] Wang Y, Tseng M. Attribute selection for product configurator design based on Gini index [J]. International Journal of Production Research, 2014, 52 (20): 6136-6145.
- [10] 童晓薇,曾思通,刘艳斌. 云制造环境下个性化产品配置研究 [J]. 图学学报, 2018, 39 (6): 1192-1199.
- [11] 韩煜东,刘伟,伊辉勇. 考虑客户感性需求下的产品模块化配置方法 [J]. 计算机辅助设计与图形学学报, 2015, 27 (7): 1320-1326.
- [12] 朱佳栋,苏少辉,陈昌,等. 面向产品配置设计的改进交互式遗传算法 [J]. 中国机械工程, 2018, 29 (20): 2474-2478.
- [13] Islam M I, Akbar M M. Heuristic algorithm of the multiple-choice multidimensional knapsack problem (MMKP) for cluster computing [C] // Proceedings of 2009 12th International Conference on Computers & Information Technology. IEEE, 2010: 157-161.
- [14] 沈记全,罗常委,侯占伟,等. 基于改进蚁群优化算法的服务组合与优化方法 [J]. 计算机工程, 2018, 44 (12): 68-73.
- [15] 胡震,邹德旋,张旭. R-dPSO 算法及其在 ATO 控制策略中的应用 [J]. 计算机工程与应用, 2018, 54 (24): 212-220.
- [16] 葛继科,邱玉辉,吴春明,等. 遗传算法研究综述 [J]. 计算机应用研究, 2008, 25 (10): 2911-2916.
- [17] 陈天伟,卢献健,杨子江. 基于遗传算法的杨赤中插值权系数解算 [J]. 桂林理工大学学报, 2015, 35 (3): 514-517.
- [18] 刘芳,马玉磊,周慧娟. 基于种群多样性的自适应遗传算法优化仿真 [J]. 计算机仿真, 2017, 34(4): 250-255.
- [19] 邢小军,闫建国. 一种基于多样性保持的遗传算法及其仿真 [J]. 计算机仿真, 2010, 27 (4): 206-209.

CTO order recommendation of electronic products based on improved genetic algorithm

HAN Hai-feng, YE Heng-zhou, ZHANG Lu

(Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, Guilin 541006, China)

Abstract: Configure-to-order (CTO) is an effective way to meet the personalized needs of users. There are many accessories involved in electronic products, and each kind of accessories has a variety of brands and models, thus very difficult to determine the order of CTO. In order to solve the above problems, a CTO order recommendation (CTOR) model for electronic products is established with function proximity as optimization target in processing cost and product power consumption as constraints. An improved genetic algorithm is proposed to solve the model. The simulation results verify the effectiveness of the improved algorithm. Adaptive genetic operator improves the convergence accuracy and speed of the algorithm. The diversity maintenance strategy of the population reduces the possibility of the algorithm falling into the local optimum to some extent.

Key words: CTO order recommendation; adaptive; genetic algorithm; population diversity