

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/349636075>

# A Hybrid Estimation of Distribution Algorithm for Quadratic Assignment Problem

Article · March 2018

CITATIONS

0

READS

25

2 authors:



**Shoufeng Ji**

Northeastern University (Shenyang, China)

60 PUBLICATIONS 809 CITATIONS

[SEE PROFILE](#)



**Rongjuan Luo**

Yunnan University of Finance and Economics

23 PUBLICATIONS 243 CITATIONS

[SEE PROFILE](#)

# 一种求解物流设施二次分配问题的混合分布估计算法

戢守峰, 罗蓉娟, 孙琦, 朱宝琳

(东北大学 工商管理学院, 辽宁 沈阳 110169)

**摘 要:** 为了解决物流设施二次分配问题, 提出了一种混合分布估计算法(HEDA)。首先, 根据 QAP 的距离和物流量矩阵信息, 提出了一种基于假设物流中心启发式规则的种群初始化方法, 用于提高初始种群的质量和算法的搜索效率; 其次, 针对 HEDA 的概率模型, 提出了一种概率矩阵初始构型生成机制和扰动操作, 用于提高算法的全局探索能力; 最后, 在分析 QAP 的结构性质的基础上, 设计了一种基于快速评价的局部搜索策略, 用于提高算法的局部开发能力。仿真计算实验和算法比较验证了 HEDA 的优化性能。

**关键词:** 二次分配问题; 混合分布估计算法; 启发式规则; 概率模型; 快速评价

**中图分类号:** F273; TP18 **文章标识码:** A **文章编号:** 1007-3221(2018)01-0074-10 **doi:** 10.12005/orms.2018.0012

## A Hybrid Estimation of Distribution Algorithm for Quadratic Assignment Problem

Ji Shou-feng, LUO Rong-juan, SUN Qi, ZHU Bao-lin

(School of Business Administration, Northeastern University, Shenyang 110169, China)

**Abstract:** In order to solve the quadratic assignment problem(QAP), a novel estimation of distribution algorithm(EDA), namely, hybrid EDA(HEDA) is proposed. Firstly, we design a heuristic rule, i. e., hypothesis-logistic-center-based heuristic rule(HLCBHR) according to the information of distance and material flow matrixes to generate the initial population, which is helpful to enhance the quality of initial population and the HEDA's search efficiency. Secondly, focusing on the probability model of HEDA, both an initial configuration generating mechanism and a perturbation operation of probability matrix are developed in order to improve the global exploration ability. In addition, based on the sufficient analysis of QAP's structure properties, a speed-up-evaluation-based local search strategy is embedded into the HEDA to enhance the local exploitation ability. Simulation experiments and comparisons demonstrate the optimization performance of the proposed HEDA.

**Key words:** quadratic assignment problem; estimation of distribution algorithm; heuristic rule; probability model; speed-up evaluation

## 0 引言

二次分配问题(quadratic assignment problem, QAP)是企业设施布置问题(facility layout problem, FLP)中一类典型问题,在现代设施规划与物流分析领域中有着广泛的应用。随着企业物流问题的复杂化和对二次分配问题研究的深入,其应用范围已经逐步扩展到物流运输、设施规划、集成电路布线、打字机键盘设计、车间调度等诸多生产服务领

域<sup>[1]</sup>。研究表明,二次分配问题属于组合优化中 NP-hard 范畴,其求解难度会随着问题规模的增加而呈指数增长,很难在多项式时间内获得问题的最优解。相对于传统数学问题而言, NP-hard 问题使用精确型算法耗时非常大,而启发式构造型算法的性能又相对较差,因此,开发有效的基于智能优化的求解算法具有十分重要的理论意义和应用参考价值。目前,使用智能算法求解大规模的二次分配问题已经得到了学术界的广泛关注。Czapiński<sup>[2]</sup>提出了一种并行多起点机制的禁忌搜索算法

收稿日期: 2015-10-17

基金项目: 国家自然科学基金项目(71572031); 辽宁省教育厅人文社科基金项目(ZJ2013014)

作者简介: 戢守峰(1958-),男,辽宁沈阳人,东北大学教授,研究方向: 物流系统建模与优化、物流与供应链管理; 罗蓉娟(1992-),女,云南保山人,东北大学博士研究生,研究方向: 物流设施建模与智能优化。

(parallel multistart tabu search, PMTS), 同时基于 CUDA 并行计算平台对二次分配问题进行了求解。Parhizgar 等<sup>[3]</sup>提出了一种求解二次分配问题的水循环优化算法(water cycle optimization, WCO), 并将两类局部搜索方案融合到 WCO 中。Benlic 等<sup>[4]</sup>针对二次分配问题设计了一种混合搜索算法(memetic algorithm, MA), 该算法将局部搜索策略和基于种群进化策略的进化算法进行了有效的融合, 同时对二次分配问题的解空间形状(landscape)进行了研究。Tosun<sup>[5]</sup>提出了一种求解二次分配问题的并行混合算法(parallel hybrid algorithm, PHA), 该算法将遗传算法和禁忌搜索进行了有效融合。Tosun<sup>[6]</sup>针对求解二次分配问题的遗传算法提出了一种新的基因重组操作, 并基于遗传算法和禁忌搜索提出了一种混合智能算法。钟一文等<sup>[7]</sup>提出了一种离散粒子群优化算法, 该算法针对二次分配问题的特点定义了离散运算规则, 并引入了粒子间的排斥算子来保质种群的多样性。上述文献表明, 智能优化算法已经成为求解二次分配问题的有效算法, 尤其针对大规模问题, 智能优化算法的优势更为明显。

作为一种新颖的基于概率模型的群智能算法, 分布估计算法(estimation of distribution algorithm, EDA) 已经成为智能计算领域的研究热点之一<sup>[8]</sup>。EDA 没有遗传算法中的变异和交叉操作, 而是通过竞争学习机制来估计优秀个体的概率模型, 然后通过对概率模型的采样来生成新一代种群。因此, EDA 能够从更为宏观的角度对解空间中可能存在优秀解的区域进行搜索, 具有较强的全局探索能力。对于离散组合优化问题, EDA 的概率模型是多维离散型随机变量的联合概率分布函数, 由于多维离散型随机变量之间往往呈现复杂的随机相关性, 无法对联合概率分布函数进行显性表达。已有文献中, 离散 EDA 普遍采用变量无关的概率模型(如 PBIL<sup>[9]</sup>, UMDA<sup>[10]</sup>)。从本质而言, 变量无关的 EDA 的进化过程就是不断确立等位基因与各个基因位之间的概率匹配关系的过程。根据无免费午餐定理(no free lunch theorem, NFLT)<sup>[11]</sup>可知, 算法对问题的平均性能是相近的, 因此在选择优化算法时需要充分考虑到问题的特征与属性。使用智能算法求解物流设施二次分配问题, 相当于在迭代过程中不断确定设施  $i(i=1, \dots, n)$  与位置  $k(k=1, \dots, n)$  的匹配程度(匹配概率), 最终找到设施

与位置之间的最优或近似最优匹配关系, 这与分布估计算法的进化机理是完全一致的。迄今为止, 关于使用分布估计的算法来求解二次分配问题的研究十分鲜见, 只有 Zhang 等<sup>[12]</sup>提出了一种基于 2-opt 局部搜索策略的分布估计算法来求解二次分配问题。因此, 开发用于求解二次分配问题的分布估计算法, 对于拓展二次分配问题的新型求解方法和开拓分布估计算法的应用领域均具有十分重要的理论意义和应用价值。

提出了一种基于混合分布估计算法(hybrid estimation of distribution algorithm, HEDA), 用于求解复杂分布情况下的物流设施二次分配问题。首先, 合理利用了 QAP 的距离和物流量矩阵信息, 设计了一种基于假设物流中心的启发式规则来生成初始种群, 用于提高初始种群的质量和 HEDA 的搜索效率; 其次, 提出了一种概率矩阵初始构型生成机制, 用于提高初始概率矩阵的信息量, 进而有效指导算法的搜索行为; 第三, 针对概率矩阵在进化过程中的元素不断向 0 或 1 收敛的缺陷, 提出了一种概率矩阵扰动操作, 用于延缓概率矩阵的过快收敛, 从而提高种群多样性和全局搜索的宽度; 最后, 在分析 QAP 的结构性质的基础上, 设计了一种基于快速评价的局部搜索策略, 用于提高算法的局部开发能力。基于 Benchmark 算例与企业设施布置实例对 HEDA 的有效性和灵敏度进行了验证。这些均是已有研究未涉及到的。

## 1 二次分配问题的数学描述

设二次分配问题的位置间的距离矩阵为  $D = [d_{kh}]_{n \times n}$ , 设施间的物流量矩阵  $C = [c_{\pi(k)\pi(h)}]_{n \times n}$ ;  $\pi = [\pi(1), \pi(2), \dots, \pi(k), \dots, \pi(h), \dots, \pi(n)]$  表示一种分配方案, 其中  $\pi(k)$  和  $\pi(h)$  分别表示位置  $k$  和位置  $h$  上所放置的设施。则二次分配问题问题可以用如下式子描述:

$$f(\pi) = \sum_{k=1}^n \sum_{h=1}^n d_{kh} c_{\pi(k)\pi(h)} \quad (1)$$

$$f(\pi^*) = \min_{\pi \in \Pi} f(\pi) \quad (2)$$

$$\pi^* = \arg\{f(\pi)\} \rightarrow \min, \forall \pi \in \Pi \quad (3)$$

式(1)是目标函数(即总物流费用)的计算公式, 式(2)和式(3)表示在所有分配方案所组成的集合  $\Pi$  中找到最优排序  $\pi^*$ , 使  $f(\pi)$  最小。

## 2 混合分布估计算法( HEDA) 设计

### 2.1 编码

编码是算法进行迭代寻优的前提。将问题方案(问题解)转换成符合算法搜索要求的编码排列(算法解)的过程称为编码,通过编码操作可以建立问题解空间与算法搜索空间之间的映射关系。根据二次分配问题的特点,HEDA 采用基于设施的编码方式,即基于各个设施标号来进行编码,譬如, $\pi = [2, 4, 1, 3, 5]$ 可以表示问题的一个编码排列,其含义是:设施 2 分配到位置 1,设施 4 分配到位置 2,设施 1 分配到位置 3,设施 3 分配到位置 4,设施 5 分配到位置 5。

### 2.2 种群初始化

EDA 的搜索过程本质上是一种概率学习过程,通过概率模型的不断更新来积累优秀个体的历史信息,并通过对概率模型的采样生成新一代种群。因此,质量较高的初始化种群有利于算法获得较好的搜索起点,进而能够更加高效率地进行后续的搜索。根据距离矩阵  $D = [d_{kh}]_{n \times n}$  和物流量矩阵  $C [c_{\pi(k) \pi(h)}]_{n \times n}$  的信息,提出了一种基于假设物流中心启发式规则(hypothesis-logistic-center-based heuristic rule, HLCBHR),并用于生成初始化种群。

通过距离矩阵  $D$  第  $k$  行所有元素和  $\sigma_k$ ,可计算出位置  $k$  到所有位置的距离之和,即

$$\sigma_k = \sum_{h=1}^n d_{kh}, \quad k = 1, 2, \dots, n \quad (4)$$

同理,使用物流量矩阵  $C$  可以计算出设施  $\pi(k)$  与所有设施之间的物流量总和  $\tilde{\sigma}_k$ ,即

$$\tilde{\sigma}_k = \sum_{h=1}^n c_{\pi(k) \pi(h)} + \sum_{h=1}^n c_{\pi(h) \pi(k)}, \quad k = 1, 2, \dots, n \quad (5)$$

基于上述分析,可根据  $\sigma_k$  和  $\tilde{\sigma}_k$  的值来生成合法的个体。首先,使用  $\sigma_k$  的值来近似估计位置  $k$  与假设物流中心之间的距离, $\sigma_k$  的值越小则认为位置  $k$  与假设物流中心越接近,通过对  $\sigma_1, \dots, \sigma_k, \dots, \sigma_n$  进行非降排序,便可以估计出所有设施与假设物流中心的接近程度;然后,将  $\tilde{\sigma}_1, \dots, \tilde{\sigma}_k, \dots, \tilde{\sigma}_n$  进行非升排序,将值最大的一个设施放置在离假设物流中心最近的位置上,依次类推。

设  $gen$  为进化代数, $PS$  为种群规模,那么使用该启发式规则可以构造出初始化种群中的第 1 个个体  $\pi_{1,gen=0} = [\pi_{1,gen=0}(1), \pi_{1,gen=0}(2), \dots, \pi_{1,gen=0}(n)]$ ,随后通过对  $\pi_{1,gen=0}$  执行随机互换操作就可以得到初始种群中的其余  $PS - 1$  个体。

HEDA 的种群初始化方法的伪代码如图 1 所示。值得注意的是,如果  $\sigma_k$  或  $\tilde{\sigma}_k (k = 1, 2, \dots, n)$  中的元素相等时,此初始化种群方法失效,需要以随机方式生成初始化种群。

算法1: 种群初始化

```

for i:= 1 to PS do
begin
  if (i=1) then
  begin
     $\pi_{1,gen=0} = [\pi_{1,gen=0}(1), \pi_{1,gen=0}(2), \dots, \pi_{1,gen=0}(n)]$ ;
  end//end if
  else
  begin
     $t\_arr = \pi_{1,gen=0}$ ;
    p0: integer, //随机交换操作
    rp_1=random[1,n];
    rp_2=random[1,n]\rp_1;
    p0 = t_arr(rp_1);
    t_arr(rp_1) = t_arr(rp_2);
    t_arr(rp_2) = p0;
     $\pi_{i,gen=0} = t\_arr$ ;
  end//end else
end// end for i

```

图 1 HEDA 种群初始化伪代码

### 2.3 概率矩阵及更新方法

对于离散组合优化问题,通常采用概率矩阵作为 EDA 的概率模型<sup>[13]</sup>。概率矩阵的具体形式如式(6)所示。

$$M(gen) = \begin{pmatrix} p_{11}(gen) & \cdots & p_{1n}(gen) \\ \vdots & \ddots & \vdots \\ p_{n1}(gen) & \cdots & p_{nn}(gen) \end{pmatrix}_{n \times n} \quad (6)$$

实际上,概率矩阵所揭示的是一种多变量相互独立的概率图模型结构,概率矩阵的第  $k$  列则是第  $k$  维离散型随机变量的边缘分布律,因此概率矩阵满足列归 1 性,即

$$\sum_{i=1}^n p_{ik}(gen) = 1, \quad k = 1, 2, \dots, n \quad (7)$$

对于不同离散组合优化问题,概率矩阵都有着不同的内涵。对于二次分配问题而言,元素  $p_{ik}$  表示设施  $i (i = 1, \dots, n)$  与位置  $k (k = 1, \dots, n)$  的匹配程度(匹配概率)。对于初始化的概率矩阵,其任意元素  $p_{ik}(gen=0) = 1/n$ 。

设  $LR$  为学习速率,  $\pi_{gb,gen} = [\pi_{gb,gen}(1), \pi_{gb,gen}(2), \dots, \pi_{gb,gen}(n)]$  为算法在第  $gen$  代的历史最优个体(best so far),则概率矩阵的更新操作的伪代码如图 2 所示。

```

算法2: 概率矩阵更新
for  $i = 1$  to  $n$  do
  begin
     $l$ : integer;
     $l = \pi_{gb,gen}(i)$ ; //等位基因位置
     $p_{li} = p_{li} + LR$ ; //更新第 $i$ 列
    for  $j = 1$  to  $n$  do
      begin
         $p_{ji} = p_{ji} / (1 + LR)$ ; //第 $i$ 列归一化处理
      end; //end for  $j$ 
    end; //end for  $i$ 
  
```

图 2 概率矩阵更新操作伪代码

## 2.4 概率矩阵初始构型生成机制

设  $genMax$  为 EDA 的最大迭代次数,  $o$  为非负无穷小的数, 那么随着概率矩阵的不断迭代更新, 当极限  $\lim_{gen \rightarrow genMax} p_{ik}(gen) + o = 1$  时, 则可以认为设施  $i$  与位置  $k$  为最佳匹配关系。由于初始化的概率模型, 概率矩阵的任意元素  $p_{ik}(gen = 0) = 1/n$ , 因此初始化的概率模型不能表征任何匹配关系, 即可以认为  $M(gen = 0)$  所包含的历史信息量为 0。

由上述分析可知, EDA 的概率矩阵所包含的历史信息量是随着迭代的进行而不断增加的, 是一个渐进的过程。在 EDA 进化初期, 概率矩阵缺乏用于指导搜索的历史信息量, 这时的概率矩阵是十分不成熟的, 对其进行采样所得到的解的质量往往偏低。由此, 需要采取一定的策略来生成概率矩阵的初始构型, 为 EDA 的采样操作做好充足的准备, 待初始构型构造完成后才对概率模型进行采样。

在生成 HEDA 概率矩阵的初始构型时, 采用进化初期不采样的策略, 即当进化代数不大于采样代数阈值  $genSAM$  时, 仅执行概率矩阵更新操作, 不进行采样操作。也就是说, HEDA 的采样操作是从第  $genSAM + 1$  代开始执行的。

## 2.5 概率矩阵的扰动操作

研究表明, 随着 HEDA 的不断迭代, 概率矩阵的元素会逐渐向 0 或 1 逼近, 这是算法进化过程中无法回避的事实。若元素过快地向 0 或 1 逼近, 将导致采样生成的种群的多样性降低, 进而可能诱发算法的早熟收敛, 并制约算法的全局搜索能力。对概率矩阵进行适当的扰动, 有利于延缓概率矩阵的元素向 0 或 1 收敛的速度。因此, 对 HEDA 的概率矩阵设计并添加了扰动操作。设  $PM$  为扰动概率,  $MR$  为扰动速率, 概率矩阵扰动策略的伪代码如图 3 所示。

```

算法3: 概率矩阵扰动操作
for  $i = 1$  to  $n$  do
  begin
     $r = \text{random}[0, 1]$ ;
    if  $r \leq PM$  then
      begin
         $l = \text{random}[1, n]$ ; //随机行
         $p_{li} = p_{li} + MR$ ; //扰动
        for  $j = 1$  to  $n$  do
          begin
             $p_{ji} = p_{ji} / (1 + MR)$ ; //第 $i$ 列归一化
          end;
        end; //end if
      else
        begin
          continue;
        end; //end else
    end; //end for  $i$ 
  
```

图 3 概率矩阵的扰动操作伪代码

## 2.6 概率矩阵的采样

HEDA 通过采样操作生成新一代种群, 采样与概率矩阵更新是 EDA 类算法最为核心的操作。对于概率矩阵  $M(gen)$ , 可以根据第  $k$  列元素 (即第  $k$  维随机变量的边缘分布律) 构造出第  $k$  维随机变量的边缘分布函数, 即

$$F_{xk}(gen) = \begin{cases} 0 & x < 1 \\ p_{\delta k} + \sum_{j=0}^{\delta-1} p_{jk} & (\delta \leq x \leq \delta+1) \wedge (1 \leq \delta < n) \wedge (p_{0k} = 0) \\ 1 & x \geq n \end{cases} \quad (8)$$

由式 (8) 可以定义边缘分布函数  $F_{ik}(gen)$  的广义反函数, 即

$$F_{xk}^{-1}(gen) = \begin{cases} \delta+1 & \text{如果 } (F_{\delta k}(gen) \leq x \leq F_{\delta+1 k}(gen)) \wedge (1 \leq \delta < n) \\ 1 & \text{其他} \end{cases} \quad (9)$$

```

算法4: 概率矩阵的采样
for  $i = 1$  to  $PS$  do
  begin
    for  $j = 1$  to  $n$  do
      begin
         $\text{CalculateMPD}(j)$ ; //构造边缘分布
         $r = \text{random}[0, 1]$ ;
         $\pi_{i,gen}(j) = F_{rj}^{-1}(gen)$ ; //生成设施标号
        for  $k = j+1$  to  $n$  do
          begin
             $p_{\pi_{i,gen}(j)k} = 0$ ;
          end; //end for  $k$ 
        end; //end for  $j$ 
      end; //end for  $i$ 
    
```

图 4 概率矩阵采样伪代码

在对概率矩阵进行采样时,按照从左到右的次序顺序采样,进而生成个体,重复  $PS$  次操作便可生成新一代的种群。由于二次分配问题的编码中,每个设施标号只能出现一次,因此,当某个设施标号被采样选中后,则将概率矩阵对应的所有行元素置零,以免被重复选中。设  $CalculateMPD(j)$  表示构造第  $j$  维随机变量的边缘分布函数的操作,则采样操作的伪代码如图 4 所示。

## 2.7 基于快速评价的局部搜索

与遗传算法不同,标准 EDA 更加侧重于从宏观层面对解空间进行搜索,缺乏对问题的局部信息的利用,这导致其局部搜索能力较差。若要提高标准 EDA 的性能,必须增强其局部搜索能力。在 HEDA 中,首先通过标准 EDA 实现对解空间的全局探索 (exploration),然后通过局部搜索机制进行相对细致局部开发 (exploitation),进而实现算法全局探索能力和局部开发能力的合理平衡。

对于离散组合优化问题而言,解空间中的解往往服从近邻假设,即优秀的解的邻域内一般也存在优秀的解。因此,HEDA 的局部搜索致力于对历史最优个体 (best so far) 的邻域展开搜索并找到更加优秀的个体,进而提高算法的局部搜索性能。邻域搜索的核心之一是邻域结构的选择,HEDA 使用的是最为简单的互换型邻域结构 (Interchange),即通过对两个设施标号的互换来构造历史最优个体的邻域。值得注意的是,在进行邻域搜索时采用“首次改进跳出原则”,即只要在邻域中找到了更加优秀的个体就停止搜索。

现就二次分配问题的互换型邻域结构进行分析。如图 5 所示,如果对历史最优个体  $\pi_{gb,gen}$  的  $i, j$  两个位置的元素进行互换,就可以得到其邻域中的一个个体  $\pi_{NI,gen}$ 。

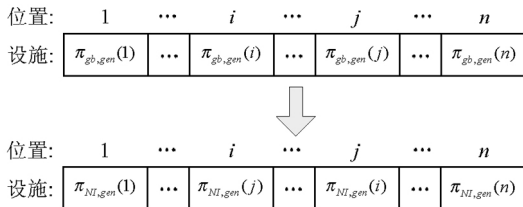


图 5 Interchange 邻域结构示意图

由于互换型邻域结构仅仅是两个位置上的元素的互换,未参与互换操作的其余  $n-2$  个元素仍处于原来的位置,由式 (1) 可知,  $\pi_{NI,gen}$  的目标函数值的变化仅仅取决于参与互换操作的  $i, j$  两个位置上的元素。将式 (1) 进行分解,可以得到  $\pi_{NI,gen}$  的

目标函数值的另一种表达形式,即

$$\begin{aligned} f(\pi_{gb,gen}) &= \sum_{k=1}^n d_{ki} c_{\pi_{gb,gen}(k)} \pi_{gb,gen}(i) + \\ &\quad \sum_{k=1 \setminus \{i\}}^n d_{ik} c_{\pi_{gb,gen}(i)} \pi_{gb,gen}(k) + \\ &\quad \sum_{k=1}^n d_{kj} c_{\pi_{gb,gen}(k)} \pi_{gb,gen}(j) + \\ &\quad \sum_{k=1 \setminus \{j\}}^n d_{jk} c_{\pi_{gb,gen}(j)} \pi_{gb,gen}(k) + \\ &\quad \sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{gb,gen}(k)} \pi_{gb,gen}(h) \\ &= TC_{ij}^A + \sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{gb,gen}(k)} \pi_{gb,gen}(h) \quad (10) \end{aligned}$$

而互换操作的  $i, j$  两个位置上的元素后,  $\pi_{NI,gen}$  的目标函数值也可以用类似于式 (10) 的方式表示,即

$$\begin{aligned} f(\pi_{NI,gen}) &= \sum_{k=1}^n d_{ik} c_{\pi_{NI,gen}(k)} \pi_{NI,gen}(i) + \\ &\quad \sum_{k=1 \setminus \{i\}}^n d_{ik} c_{\pi_{NI,gen}(i)} \pi_{NI,gen}(k) + \\ &\quad \sum_{k=1}^n d_{kj} c_{\pi_{NI,gen}(k)} \pi_{NI,gen}(j) + \\ &\quad \sum_{k=1 \setminus \{j\}}^n d_{jk} c_{\pi_{NI,gen}(j)} \pi_{NI,gen}(k) + \\ &\quad \sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{NI,gen}(k)} \pi_{NI,gen}(h) \\ &= TC_{ij}^B + \sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{NI,gen}(k)} \pi_{NI,gen}(h) \quad (11) \end{aligned}$$

由式 (1) 可知

$$\begin{aligned} &\sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{gb,gen}(k)} \pi_{gb,gen}(h) \\ &= \sum_{k=1 \setminus \{i,j\}}^n \sum_{h=1 \setminus \{i,j\}}^n d_{kh} c_{\pi_{NI,gen}(k)} \pi_{NI,gen}(h) \quad (12) \end{aligned}$$

易知,进行互换操作后所得到的新个体的目标函数值的变化量为

$$\Delta TC = TC_{ij}^B - TC_{ij}^A \quad (13)$$

因此,位置  $i, j$  的元素进行互换后,所得到的  $\pi_{gb,gen}$  邻域中的一个个体  $\pi_{NI,gen}$  的目标函数为

$$f(\pi_{NI,gen}) = f(\pi_{gb,gen}) + \Delta TC \quad (14)$$

由上述分析可知,在计算互换型邻域中个体的目标函数值时,只需计算互换操作后目标函数值的变化量,在计算该变化量时所需要的乘法和加法操作次数为  $(n-1+n+n-2+n-1) \times 2 \times 2 = 16n-16$ ,即采用此方法计算目标函数值的算法复杂度为  $O(n)$ ;然而,使用式 (1) 计算  $\pi_{NI,gen}$  的目标函数值时所需要的乘法和加法操作次数为  $(n-1+n) \times n + n-1 = 2n^2-1$ ,即算法复杂度为  $O(n^2)$ 。因此,

使用上述快速评价方法,可以有效降低计算目标函数值的算法复杂度,进而提高局部搜索的效率。

此外,由于局部搜索时采用了“首次改进跳出原则”,故可根据  $\Delta TC$  与 0 的关系来预测进行元素互换操作是否有意义,如果  $\Delta TC$  的值小于 0,可以直接判断相应  $\pi_{NI,gen}$  的互换操作是有效的,在得到相应的编码排列和目标函数值后,可以直接跳出局部搜索。因此,采用这种快速评价机制不仅有利于提高局部搜索的效率,而且还可以隐式地避免局部搜索中无效和迂回的搜索。设  $Interchange(t\_arr, i, j)$  表示对个体  $t\_arr$  的  $i, j$  两个位置的元素执行互换操作,  $CalculateSP(\pi_{NI,gen})$  表示使用快速评价机制计算目标函数值,则 HEDA 的局部搜索策略可用图 6 所示的伪代码表示。

```

算法5: 基于快速评价机制的局部搜索
for i:=1 to n-1 do
begin
  for j:=i+1 to n do
  begin
    t_arr =  $\pi_{gb,gen}$ ;
     $\pi_{NI,gen} = Interchange(t\_arr, i, j)$ ;
    if ( $\Delta TC < 0$ ) then
    begin
      CalculateSP( $\pi_{NI,gen}$ );
       $\pi_{gb,gen} = \pi_{NI,gen}$ ;
       $f(\pi_{gb,gen}) = f(\pi_{NI,gen})$ ;
      break; //退出局部搜索并更新  $\pi_{gb,gen}$ 
    end
  end
  else
  begin
    continue; //继续进行局部搜索
  end
end; //end for j
end; //end for i

```

图 6 基于快速评价的局部搜索的伪代码

由图 6 可知, HEDA 所采用的局部搜索策略属于基于互换型邻域结构的大规模邻域搜索, 当采用“首次改进跳出原则”时, 最坏情况下需要进行  $n \times (n-1)/2$  次邻域搜索。因此, 基于快速评价机制的局部搜索在最坏情况下评价所有个体所需的乘法和加法次数为  $(16n-16) \times n \times (n-1)/2 = 8n^3 - 16n^2 + 8n$ , 即算法的复杂度为  $O(n^3)$ ; 如果不采用快速评价策略进行局部搜索, 可知在最坏情况下评价所有个体所需要的乘法和加法次数为  $(2n^2-1) \times n \times (n-1)/2 = (2n^4 - 2n^3 - n^2 + n)/2$ , 即算法的复杂度为  $O(n^4)$ 。综上分析, 所提出的快速评价机制有利于提高局部搜索的效率, 同时这种大规模邻域搜索机制与惯用的随机邻域搜索(随机选择两个位置进行交换操作)相比要细致得多, 有利

于对可能存在优秀解的  $\pi_{gb,gen}$  邻域进行精细的搜索。

## 2.8 HEDA 的算法流程

结合 2.1 至 2.7 节的内容, 现给出 HEDA 的算法流程(如图 7 所示), 具体步骤如下:

步骤 0 初始化;

步骤 0.1 令  $gen = 0$ ;

步骤 0.2 按照 2.2 节所述的方法进行种群初始化;

步骤 0.3 按照 2.3 节所述的方法进行概率矩阵初始化;

步骤 1  $gen = gen + 1$ ;

步骤 2 基于快速评价机制的局部搜索, 按照 3.7 所述的方法进行局部搜索, 并使用  $\pi_{NI,gen}$  更新  $\pi_{gb,gen}$ ;

步骤 3 概率矩阵的更新;

步骤 3.1 按照 2.3 节所述的方法使用历史最优个体更新概率矩阵;

步骤 3.2 概率矩阵归一化处理;

步骤 4 概率矩阵的扰动操作, 按照 2.5 节所述的方法对概率矩阵进行扰动操作;

步骤 5 概率矩阵的采样, 如果  $gen > gen_{SAM}$ , 则按照 2.6 所述的方法对概率矩阵进行采样, 否则, 不执行采样操作并转向步骤 3, 以便生成 2.4 节所述的概率矩阵初始构型;

步骤 6 如果  $gen < gen_{MAX}$ , 转向步骤 1, 否则, 算法终止并输出结果。

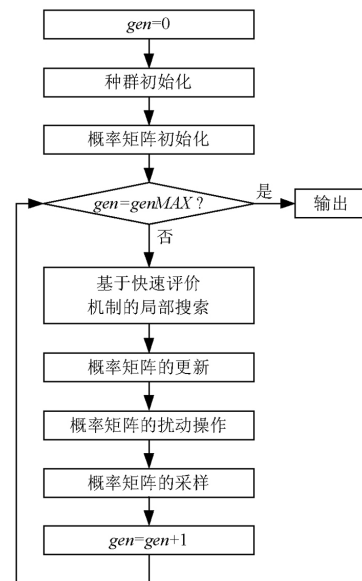


图 7 HEDA 的算法流程

由上述 HEDA 的流程可知, 步骤 0.2 采用基于假设物流中心的启发式规则, 充分利用了二次分配

问题本身的结构性质,此种初始化种群方法有利于提高算法的搜索效率;步骤5采取了进化初期不采样的原则,用于生成概率矩阵的初始构型,提高了用于采样的概率矩阵的历史信息量,有利于提升采样的质量和效率;步骤4中的概率矩阵扰动操作,可以有效延缓概率矩阵元素向0或1收敛的速度,有利于提高 HEDA 进化过程中的种群多样性;步骤2中的局部搜索策略可以显著地提高算法的局部搜索能力,有利于克服 EDA 类算法局部搜索能力较差的缺陷。因此,HEDA 的全局探索能力,局部开发能力和搜索效率均得到了合理的平衡,算法可有效获得二次分配问题的优质解。

### 3 仿真计算实验与灵敏度分析

仿真计算实验从 QAPLIB<sup>[14]</sup> 中随机选择了问题规模为 12 ~ 256 的 12 组算例。实验硬件环境为 Core i5 3.3 GHz, RAM 3.46 GB; 软件环境为 Visual Studio 2008( C++ 语言), Windows 7。所有算法均独立重复运行 10 次,使用 *AVG* 表示均值, *BEST* 表示求得的最佳值, *SD* 表示标准差, *f(BKS)* 表示相应算例目前已知最好解的目标函数, *BRE* 表示最优相对百分误差, *ARE* 表示平均相对百分误差。

为了确保算法比较的公平性,设置 HEDA 的进化代数为 2000 代(平均运行时间为  $T(s)$  /秒),所有算法的运行时间均取为  $T(s)$ ,如表 5 所示。

#### 3.1 参数设置与灵敏度分析

为了对 HEDA 的参数进行合理配置,采用实验设计<sup>[15]</sup> (design of experiment, DOE) 来确定 HEDA 的参数。在开展 DOE 时,种群规模 *PS*, 学习速率 *LR*, 概率矩阵扰动概率 *PM*, 概率矩阵扰动速率 *MR* 和采样代数阈值 *genSAM* 这 5 个参数均取 4 个水平(如表 1 所示),选择规模为  $L_{16}(4^5)$  的正交实验,对各种参数组合下的所有测试问题独立重复

运行 10 次,取 HEDA 求解 12 个算例的 *AVG* 的平均值 *AVG* 作为评价指标,所有正交实验的参数组合及 *AVG* 如表 2 所示。

表 1 参数水平设置

参数	水平			
	1	2	3	4
<i>PS</i>	30	50	80	100
<i>LR</i>	0.02	0.04	0.06	0.08
<i>PM</i>	0.1	0.2	0.3	0.4
<i>MR</i>	0.005	0.01	0.015	0.02
<i>genSAM</i>	100	150	200	250

表 2 参数组合与评价指标

参数组合	水平						<i>AVG</i>
1	1	1	1	1	1	1	4659782.95
2	1	2	2	2	2	2	4673179.90
3	1	3	3	3	3	3	4725992.80
4	1	4	4	4	4	4	4776625.45
5	2	1	2	3	4	4	4821090.85
6	2	2	1	4	3	3	4633654.68
7	2	3	4	1	2	2	4617529.75
8	2	4	3	2	1	1	4614046.40
9	3	1	3	4	2	2	4900006.15
10	3	2	4	3	1	1	4794121.00
11	3	3	1	2	4	4	4613451.27
12	3	4	2	1	3	3	4600981.37
13	4	1	4	2	3	3	4846451.77
14	4	2	3	1	4	4	4620324.67
15	4	3	2	4	1	1	4620747.13
16	4	4	1	3	2	2	4600979.48

由表 2 的实验结果,可以得到各参数的极差及在不同水平下的响应值(如表 3 所示),由表 3 可以绘制各参数的响应曲线,如图 8 所示。

表 3 各参数的极差及响应值

水平	响应值				
	<i>PS</i>	<i>LR</i>	<i>PM</i>	<i>MR</i>	<i>genSAM</i>
1	4708895.28	4806832.93	4626967.10	4624654.68	4672174.37
2	4671580.42	4680320.06	4678999.81	4686782.33	4697923.82
3	4727139.95	4644430.24	4715092.50	4735546.03	4701770.15
4	4672125.76	4648158.18	4758681.99	4732758.35	4707873.06
极差	55559.53	162402.69	131714.90	110891.35	35698.69
等级	4	1	2	3	5

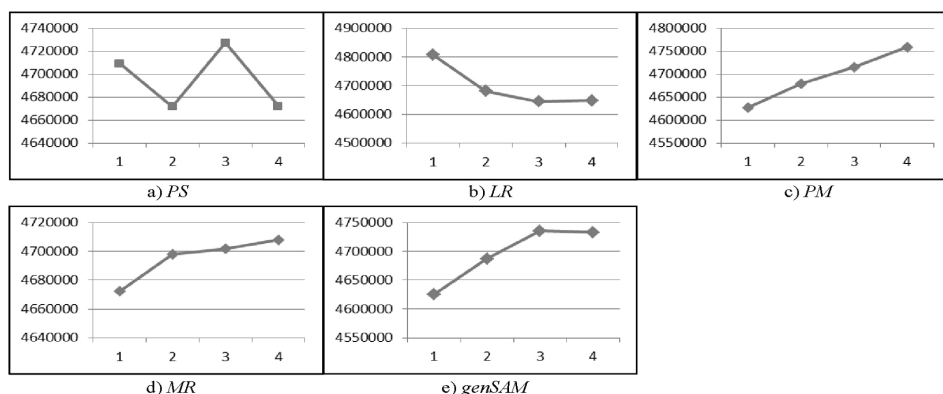


图 8 各参数响应曲线



基于上述分析, HEDA 的参数取值如下:  $PS = 50$ ,  $LR = 0.06$ ,  $PM = 0.1$ ,  $MR = 0.005$ ,  $genSAM = 100$ 。由极差可知, HEDA 对学习速率  $LR$  和概率矩阵扰动概率  $PM$  最为敏感, 而采样代数阈值  $genSAM$  对算法的影响最弱。

表 4 启发式规则生成的初始种群与随机初始种群比较

名称	规模	随机初始种群		启发式初始种群	
		BEST	AVG	BEST	AVG
Had12	12	1752	1890.43	1698	1774.10
Had20	20	7450	7760.70	7374	7498.03
Nug30	30	7680	8124.45	7180	7467.93
Lipa40b	40	610462	622130.22	594106	591085.24
Wil50	50	54364	55612.04	53660	53953.25
Lipa60a	60	109821	110110.23	109821	108752.30
Lipa70a	70	173358	173728.63	173358	171567.21
Lipa80a	80	258099	258433.76	257908	255242.56
Lipa90a	90	367054	367452.60	366859	362876.88
Wil100	100	296746	299834.81	290580	287968.78
Tho150	150	9646826	9796964.13	9467312	9398120.05
Tai256c	256	50207038	53630212.68	51454336	53028932.73
均值		5145054.17	5444354.56	5232016.00	5356269.92

表 5 HEDA 与 HEDA-V1, HEDA-V2 比较结果

名称	规模	HEDA_V1		HEDA_V2		HEDA		$T(s)$
		AVG	SD	AVG	SD	AVG	SD	
Had12	12	1743.80	17.47	1699.20	9.13	1668.01	4.89	0.35
Had20	20	7389.80	39.78	7331.80	64.72	7009.98	37.41	0.95
Nug30	30	7286.60	73.33	7147.80	108.88	6228.54	54.57	2.28
Lipa40b	40	590011.40	2246.09	588677.90	1889.09	476581.00	0.00	4.39
Wil50	50	53442.60	335.85	52763.40	257.07	49295.86	244.55	7.07
Lipa60a	60	109731.00	36.63	109418.50	91.74	108386.68	51.77	11.15
Lipa70a	70	173212.40	121.76	172828.30	95.56	171335.42	54.78	15.91
Lipa80a	80	257825.70	133.70	257253.60	124.07	255250.94	124.37	21.48
Lipa90a	90	366577.00	195.30	365824.90	190.61	363392.43	134.99	27.01
Wil100	100	290120.40	384.60	287969.20	640.44	276284.42	417.19	26.99
Tho150	150	9377536.40	25081.79	9046563.00	44119.59	8630592.62	38357.49	51.08
Tai256c	256	49837443.80	317688.27	46000758.60	185363.30	44945045.07	24419.65	451.61
均值		5089360.10	28862.88	4741519.70	19412.85	4607589.20	5325.14	51.69

由表 5 可知, HEDA\_V2 的性能占优于 HEDA\_V1, 这表明了概率矩阵扰动策略和概率矩阵初始构型生成机制是有效的; 其次, HEDA 明显占优于 HEDA\_V1 和 HEDA\_V2, 这表明所提出的基于快速评价机制的局部搜索策略的有效性; 第三, HEDA 的标准差在多数大多数情况下小于其余两种对比算法, 这表明所提出的关键操作有利于提高算法的鲁棒性。

### 3.4 HEDA 与已有最好解的比较结果

为了对 HEDA 的有效性进行验证, 考虑将 HEDA 与 12 组算例的已有的最好解 (best known solution, BKS) 的目标函数值进行比较, 比较结果

### 3.2 基于假设物流中心启发式规则的性能验证

为了验证所提出的基于假设物流中心的启发式规则的有效性, 将启发式规则所生成的初始种群与随机初始种群进行质量比较, 使用初始种群中所有个体的目标函数值的均值 (AVG) 及最好值 (BEST) 作为评价指标, 比较结果如表 4 所示。

由表 4 可知, 使用所提出的启发式规则所生成的初始种群的质量明显优于随机初始化种群, 这表明使用所提出的启发式规则可以有效提高初始种群的质量, 进而有效提高 HEDA 的搜索效率。

### 3.3 HEDA 全局性能验证

为了对所提出的 HEDA 全局性能进行验证, 考虑将 HEDA 与如下两种变异算法进行比较, 即 HEDA\_V1: 不含概率矩阵扰动操作和局部搜索操作, 其余操作与 HEDA 相同; HEDA\_V2: 不含局部搜索操作, 其余操作与 HEDA 相同。在对 HEDA 的关键操作进行验证时, 主要考察对比算法的均值 AVG 和标准差 SD, 实验结果如表 5 所示。

如表 6 所示。HEDA 所求得的最佳解的排列及目标函数值可参。

表 6 HEDA 与 BKS 比较结果

名称	规模	$f(BKS)$	BRE (%)	ARE (%)
Had12	12	1652	0.85	0.97
Had20	20	6922	0.12	1.27
Nug30	30	6124	0.75	1.70
Lipa40b	40	476581	0.00	0.00
Wil50	50	48816	0.37	0.98
Lipa60a	60	107218	0.94	1.09
Lipa70a	70	169755	0.85	0.93
Lipa80a	80	253195	0.74	0.81
Lipa90a	90	360630	0.71	0.77
Wil100	100	273038	0.89	1.19
Tho150	150	8133398	5.51	6.11
Tai256c	256	44759294	0.36	0.42

由表 6 可知,在绝大多数情况下  $BRE$  和  $ARE$  可以控制在 1% 之内,显然, HEDA 的解的质量是可以接受的。这也充分表明了 HEDA 是求解二次分配问题的有效算法。

3.5 HEDA 与其他算法比较结果

鉴于所选用的算例及性能评价指标与现有文献中的有效算法不尽相同,这里仅与两类遗传算法

(genetic algorithm, GA) 的变种算法进行比较,比较结果如表 7 所示。其中,GA: 标准遗传算法,使用 HEDA 的启发式规则初始化种群,不含基于快速评价机制的局部搜索策略,交叉概率为 0.7,变异概率为 0.3,采用基于轮盘赌(比例)选择法的选择操作; GA\_LS: 在 GA 中加入基于快速评价机制的局部搜索操作,其余设置与 GA 相同。

表 7 HEDA 与 GA\_noLS, GA\_LS 比较结果

名称	规模	GA		GA_LS		HEDA	
		$BRE$ (%)	$ARE$ (%)	$BRE$ (%)	$ARE$ (%)	$BRE$ (%)	$ARE$ (%)
Had12	12	1.82	1.89	0.00	0.00	0.85	0.97
Had20	20	1.50	2.61	0.93	1.77	0.12	1.27
Nug30	30	7.77	10.67	1.60	3.50	0.75	1.70
Lipa40b	40	21.62	22.91	16.63	19.81	0.00	0.00
Wil50	50	5.27	6.22	2.90	4.69	0.37	0.98
Lipa60a	60	1.83	1.92	1.09	1.45	0.94	1.09
Lipa70a	70	1.64	1.72	0.96	1.45	0.85	0.93
Lipa80a	80	1.54	1.59	1.24	1.34	0.74	0.81
Lipa90a	90	1.40	1.48	1.14	1.30	0.71	0.77
Wil100	100	5.39	5.71	4.35	4.83	0.89	1.19
Tho150	150	12.71	13.42	12.24	13.10	5.51	6.11
Tai256c	256	2.59	3.98	0.42	0.73	0.36	0.42
均值		5.42	6.18	3.63	4.50	1.01	1.35

由表 7 可知, HEDA 的性能明显占优于两类遗传算法,这也进一步表明了 HEDA 求解二次分配问题的优势。值得注意的是,即使在遗传算法中加入基于快速评价机制的局部搜索,其性能仍然劣于 HEDA,这表明所提出的基于快速评价机制的局部搜索策略与 HEDA 进行混合是合理的。综上, HEDA 可以有效地获取二次分配问题的优质解,是一种有效的优化算法。

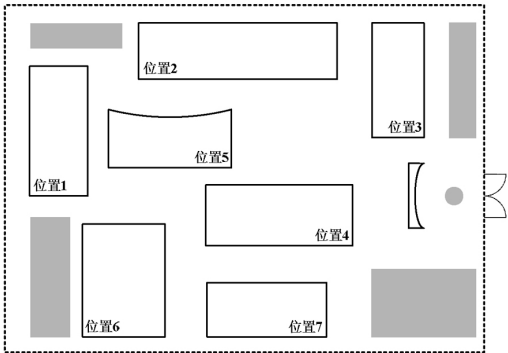


图 9 辽阳第地嘉交通物流园区可利用位置示意图

3.6 基于企业实例的求解

以辽阳第地嘉交通物流园区为例,检验 HEDA 算法的性能及灵敏度。该园区由辽阳第地嘉仓储物流有限公司建设营运,主要从事电商与 O2O 产品的存储与加工业务。园区内的 7 个功能设施为: (1) 包装材料生产与存储区, (2) 电商与 O2O 产品存储与加工区, (3) 综合品管理服务区, (4) 电商与

O2O 产品包装区, (5) 电商与 O2O 产品质检 I 区, (6) 电商与 O2O 产品质检 II 区, (7) 出园公共服务区。园区内现有 7 个可利用的位置,如图 9 所示。

采用每对位置几何中心之间的欧氏距离来获取距离矩阵(表 8),再根据园区内各个功能设施之间的工艺要求,度量出功能设施之间的物流量关系(表 9)。

表 8 距离矩阵

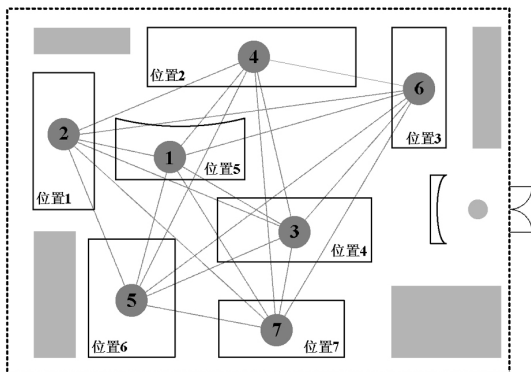
位置	距离						
	1	2	3	4	5	6	7
1	0	210	380	279	90	173	296
2	210	0	160	174	110	286	280
3	380	160	0	180	266	388	385
4	279	174	180	0	130	170	80
5	90	110	266	130	0	125	205
6	173	286	388	170	125	0	64
7	296	280	385	80	205	64	0

表 9 物流量矩阵

设施	物流量						
	1	2	3	4	5	6	7
1	0	50000	8000	5300	32000	200	10000
2	4500	0	1400	1300	1200	800	1400
3	5900	500	0	300	850	10	35
4	1500	85	12000	0	640	8000	120
5	200	350	9000	830	0	40	8000
6	0	140	5000	0	0	0	0
7	0	0	3500	0	0	0	0

获取距离矩阵与物流量矩阵信息后,采用所提

出的 HEDA 算法独立重复运行 10 次,所获的结果如表 10 所示。最终所获得的功能设施分配方案如图 10 所示。



注:  
1: 包装材料生产与存储区 2: 电商与O2O产品存储与加工区 3: 综合品管服务区  
4: 电商与O2O产品包装区 5: 电商与O2O产品质量检I区 6: 电商与O2O产品质量检II区  
7: 出园公共服务区

图 10 辽阳第地嘉交通物流园区功能设施分配方案

表 10 求解结果

BEST	AVG	SD	T(s)	分配方案
22709740	22709740	0.00	<0.20	[2-4-6-3-1-5-7]

由求解结果可知,使用 HEDA 可以在极短的时间内获取实际问题的设施分配方案。当算法独立重复运行 10 时的标准差为 0,这表明了 HEDA 求解实际问题时具有较强的鲁棒性。这再次验证了 HEDA 具有良好的优化能力,在求解物流设施二次分配问题时具有较高的灵敏度。

#### 4 结论

在提出的 HEDA 算法中,首先,为了提高初始种群的质量和算法的搜索效率,提出了一种基于距离和物流量矩阵信息的启发式规则用于生成初始种群;其次,提出了一种概率矩阵初始构型生成机制,用于提高概率矩阵的初始信息积累量,进而提高算法的采样质量与采样效率;再次,提出了一种概率矩阵扰动操作,用于延缓概率矩阵元素向 0 或 1 收敛的速度,进而提高种群的多样性与全局搜索的深度;最后,在分析 QAP 的结构特征的基础上,设计了一种基于解的快速评价机制的局部搜索策略,用于提高算法的局部开发能力。仿真实验和算法比较验证了所提出的 HEDA 的有效性和鲁棒性。基于企业设施布置的案例对 HEDA 求解实际问题的灵敏度进行了验证。下一步的工作是设计并开发基于 EDA 的有效混合算法,并用于求解多目标二次分配问题。

#### 参考文献:

- [1] Loiola E M, Abreu N M M D, Boaventura-Netto P O. A survey for the quadratic assignment problem [J]. European Journal of Operational Research, 2007, 176 (2): 57-690.
- [2] Czapinski M. An effective parallel multistart tabu search for quadratic assignment problem on cuda platform [J]. Journal of Parallel & Distributed Computing, 2013, 73 (11): 461-468.
- [3] Parhizgar M, Shiri F M. Solving quadratic assignment problem using water cycle optimization algorithm [J]. International Journal of Intelligent Information Systems, 2014, 3(6-4): 75-79.
- [4] Benlic U, Hao J K. Memetic search for the quadratic assignment problem [J]. Expert Systems with Applications, 2015, 42(1): 584-595.
- [5] Tosun U. On the performance of parallel hybrid algorithms for the solution of the quadratic assignment problem [J]. Engineering Applications of Artificial Intelligence, 2015, 39: 267-278.
- [6] Tosun U. A new recombination operator for the genetic algorithm solution of the quadratic assignment problem [J]. Procedia Computer Science, 2014, 32: 29-36.
- [7] 钟一文,蔡荣英. 求解二次分配问题的离散粒子群优化算法 [J]. 自动化学报, 2007, 33(8): 871-874.
- [8] Liang X, Chen H, Lozano J. A Boltzmann-based estimation of distribution algorithm for a general resource scheduling model [J]. IEEE Transactions on Evolutionary Computation, Published online: DOI: 10.1109/TEVC, 2014. 238-2135.
- [9] Li Z C, Qian B, Hu R. A self-adaptive hybrid population-based incremental learning algorithm for m-machine reentrant permutation flow-shop scheduling [J]. Lecture Notes on Computer science, 2013, 7995: 8-20.
- [10] Zhao J, Han C Z, Wei B. A novel univariate marginal distribution algorithm based discretization algorithm [J]. Statistics and Probability Letters, 2012, 82 (11): 2001-2007.
- [11] Wolpert D H, Macready W G. No free lunch theorems for optimization [J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 67-82.
- [12] Zhang Q, Sun J, Tsang E. Estimation of distribution algorithm with 2-opt local search for the quadratic assignment problem [J]. Studies in Fuzziness and Soft Computing, 2006, 192: 281-292.
- [13] 李作成,钱斌,胡蓉,罗蓉娟. 遗传-分布估计算法求解化工生产中一类带多工序的异构并行机调度问题 [J]. 2014, 65(03): 981-992.
- [14] Burkard R E, Karisch S E, Rendl F. QAPLIB-a quadratic assignment problem library [J]. Journal of Global optimization, 1997, 10(4): 391-403.
- [15] 李子辉,钱斌,胡蓉. 自适应混合 EDA 求解一类三阶段装配流水线调度问题 [J]. 计算机集成制造系统, 2015, 21(7): 1829-1845.



知网查重限时 7折 最高可优惠 120元

本科定稿，硕博定稿，查重结果与学校一致

立即检测

免费论文查重: <http://www.paperyy.com>

3亿免费文献下载: <http://www.ixueshu.com>

超值论文自动降重: [http://www.paperyy.com/reduce\\_repetition](http://www.paperyy.com/reduce_repetition)

PPT免费模版下载: <http://ppt.ixueshu.com>

---