

## 新冠肺炎疫情条件下的企业复工复产优化规划方法\*

郑宇军<sup>1,†</sup> 吴晨昕<sup>1</sup> 陈恩富<sup>2</sup> 卢雪琴<sup>3</sup> 张敏霞<sup>3</sup>

**摘要** 新冠病毒肺炎疫情对整个经济社会发展造成了很大冲击,如何在不放松疫情防控的前提下科学规划企业复工复产,这是地方政府面临的一个重要挑战。基于浙江省在统筹疫情防控和经济社会发展工作中的有关经验,本文建立了一个疫情条件下企业复工复产规划问题的整数规划模型,其目的是要在不违反疫情传播风险等约束下,从大批申请企业中选择一部分批准复工复产并安排优先顺序,以尽可能满足社会对相关产业产能的需求。为有效求解该问题,本文提出了一个改进的禁忌搜索算法,它使用贪心策略来构造一个初始解,并不断通过可变规模的邻域搜索来探寻更优的解,在多个地区企业复工复产规划问题实例上的计算结果验证了该算法的效率。

**关键词** 复工复产, 疫情防控, 优化, 整数规划, 禁忌搜索

中图分类号 O221.4, O229

2010 数学分类号 90C10, 68W20

## An optimization method for production resumption planning under COVID-19 Epidemic\*

ZHENG Yujun<sup>1,†</sup> WU Chenxin<sup>1</sup> CHEN Enfu<sup>2</sup>

LU Xueqin<sup>3</sup> ZHANG Minxia<sup>3</sup>

**Abstract** The outbreak of the novel coronavirus pneumonia (COVID-19) has caused a great impact on the whole economic and social development. It is an important challenge for local governments to plan the production resumption of enterprises without relaxing the epidemic prevention and control. Based on the experiences of Zhejiang Province in overall planning of epidemic prevention and control and economic development, in this paper, we formulate a production resumption planning problem, which selects a subset of enterprises from a large number of candidates that apply for production resumption and determines their order of resumption under epidemics, so as to satisfy the social demand for industrial capacities as much as possible without violating the constraints such as epidemic spreading risk. To efficiently solve this problem, we propose an improved tabu search algorithm, which uses a greedy strategy to construct an initial solution and continually explores a better solution based on variable neighborhood search. Computational results on enterprise production resumption planning in several regions demonstrates the efficiency of our method.

收稿日期: 2020-03-17

\* 基金项目: 国家自然科学基金(No. 61872123), 浙江省自然科学基金(No. LR20F030002)

1. 杭州师范大学信息科学与工程学院, 杭州 311121; School of Information Science and Engineering, Hangzhou Normal University, Hangzhou 311121, China

2. 浙江省疾病预防控制中心, 杭州 310051; Zhejiang Provincial Center for Disease Prevention and Control, Hangzhou 310051, China

3. 浙江工业大学计算机科学与技术学院, 杭州 310023; College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

† 通信作者 E-mail: yujun.zheng@computer.org

**Keywords** production resumption, epidemic prevention and control, optimization, integer programming, tabu search

**Chinese Library Classification** O221.4, O229

**2010 Mathematics Subject Classification** 90C10, 68W20

新冠病毒肺炎疫情对整个经济社会发展造成了很大冲击。随着疫情防控形势逐步好转, 平稳有序地组织企业复工复产, 是化解疫情不利影响、保持经济社会发展的关键。但在疫情条件下, 企业复工复产不能像开闸放水那样“一起上”, 而是要应对企业员工疫情传播风险控制、企业复工复产资源保障、产业链上下游企业协同复工复产等一系列挑战。因此, 政府管理部门需要优化安排企业复工复产, 在不放松疫情防控的前提下, 最大限度地满足社会对相关产业的产能恢复需求。

面对本次新冠肺炎疫情, 浙江省不仅是最早启动重大突发公共卫生事件一级响应的省份, 也是最早统筹规划企业复工复产的省份。自2020年2月9日起, 全省工作重心从集中精力防控疫情向疫情防控和复工复产“两手硬、两战赢”转变, 充分运用基于大数据和人工智能的“精密智控”机制, 因地因时、分类有序地推进企业复工复产。截至2月25日, 浙江省规模以上工业企业复工44370家, 复工率达99.8%, 员工到岗率60.1%, 产能恢复率50%以上<sup>[1]</sup>。

企业生产规划问题在文献中有着广泛的研究, 常用的问题模型包括线性规划<sup>[2,3]</sup>、整数规划<sup>[4,5]</sup>、混合整数规划<sup>[6~8]</sup>、随机规划<sup>[9,10]</sup>、目标规划<sup>[11,12]</sup>等, 其中绝大多数是NP-难题<sup>[13]</sup>。但目前还没有关于疫情条件下企业复工复产规划问题的研究报道。基于浙江省在统筹疫情防控和经济社会发展工作中的有关经验, 本文提出了一个疫情条件下企业复工复产规划的模型, 它要从大批申请企业中选择一部分批准复工复产, 并在规定的决策周期内安排其复工复产时间, 在不违反疫情传播风险等约束下, 使得所有相关产业的综合产能需求满足率最大化。该问题被建模为一个非线性整数规划问题模型。由于每个决策周期(目前浙江省为一周)内申请复工复产的企业可能有成千上万家, 问题的解空间非常庞大, 传统的数学规划方法难以有效求解该问题。为此, 本文提出了一个改进的禁忌搜索算法<sup>[14,15]</sup>, 它使用贪心策略来构造一个初始解, 而后不断通过邻域搜索来探寻更优的解, 并利用禁忌表来降低重复搜索的可能性。该算法还通过动态改变邻域大小来平衡全局搜索和局部搜索, 并在搜索陷入停滞时重新初始化当前解。在杭州、绍兴、金华等地区的复工复产规划问题实例上的计算结果验证了该算法的效率。

## 1 企业复工复产规划问题模型

本文针对疫情条件下企业复工复产规划问题建立的数学模型描述如下。地方政府管理部门需要处理 $n$ 个企业 $\{E_1, E_2, \dots, E_n\}$ 的复工复产申请, 在疫情防控不放松的前提下, 尽可能地恢复经济发展。考虑到疫情风险评估的动态性, 本文假定以7天为一个周期进行决策。问题的决策变量记为 $x = \{x_1, x_2, \dots, x_n\}$ , 其中每个 $x_i$ 是一个整数变量, 取值为0时表示不批准企业 $E_i$ 在本周期复工复产, 取值为1~7时表示批准 $E_i$ 自第 $x_i$ 日起启动复工复产。企业申报内容可分为复工人员信息和生产信息两部分, 其中人员信息主要用于评估疫情传播风险, 生产信息用于评估经济发展效果。

设企业 $E_i$ 的复工人员来源于 $K_i$ 个地区, 其中第 $k$ 个地区的员工数量为 $s_{ik}$ ; 在统计分析中, 基于浙江省“五色疫情图”评估策略, 获取每个地区的疫情风险等级 $r_{ik}$ (取值为1~5,

分别对应低风险、中低风险、中风险、中高风险、高风险), 以及从该地区到企业所在地的交通时长 $t_{ik}$ , 据此估算该地区的人员返程传播风险( $1 \leq k \leq K_i$ ):

$$R_{ik}^{(T)} = b^{r_{ik} \log_2(1+s_{ik} t_{ik})} - 1, \quad (1)$$

其中 $b$ 是一个大于1的常数(本文设为1.18)。这是我们从实践中分析提炼出的一个经验公式, 它表示传播风险 $R_{ik}^{(T)}$ 随 $r_{ik}$  和 $\log_2(1+s_{ik} t_{ik})$ 的增长而呈指数级增长。图1(a)~(d)分别给出了在1~4风险等级下, 估算的传播风险值随人数(0~100)和交通时长(0~50 h)的变化情况(高风险地区人员不允许返程复工)。可见对于低风险地区, 返程传播风险随人数和时长的增加而增长较慢; 对于中高风险地区, 返程传播风险随人数和时长的增加而增长较快。不过, 在实际应用中, 如果某个企业从某个地区的返程传播风险值超过阈值(本文设为100), 则要求该地区人员必须分批(间隔至少1天)返回, 每批的风险单项值不超过阈值, 并以各批风险单项值之和作为 $R_{ik}^{(T)}$ 的计算结果, 以免按式(1.1)对员工众多的大企业计算出的风险值超高。

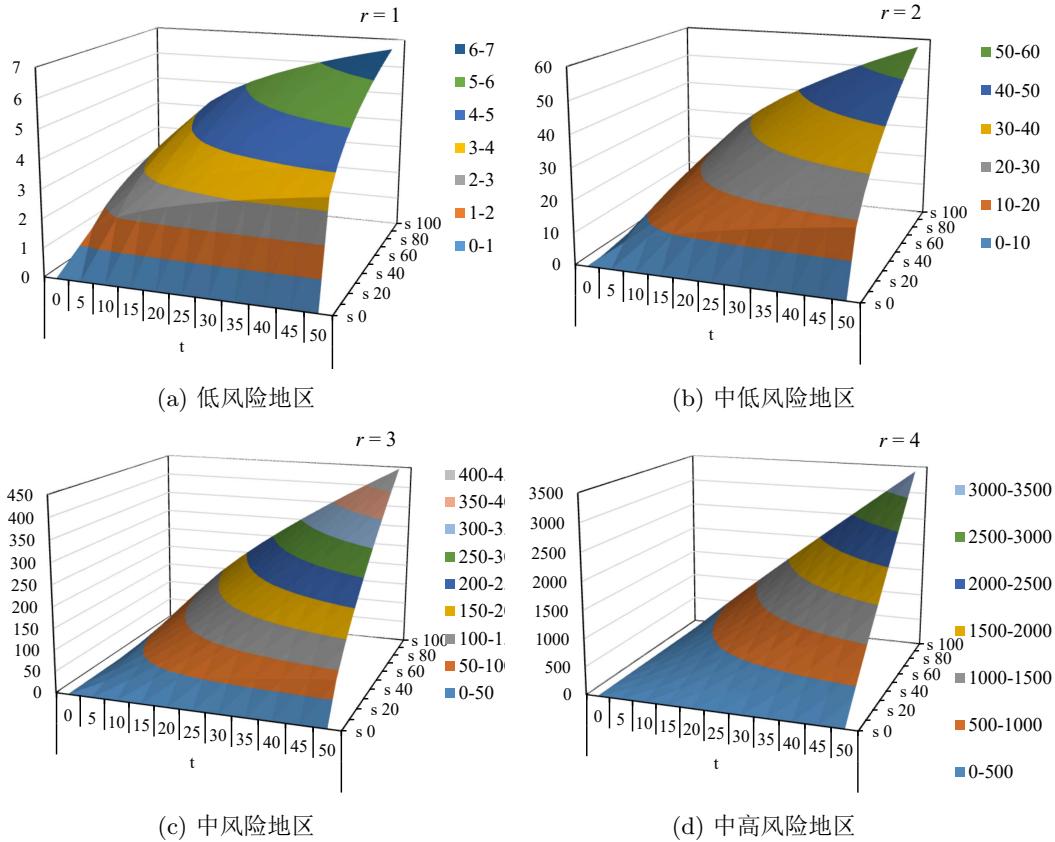


图1 人员返程传播风险数值变化图

再考虑复工后的员工工作情况, 设企业 $E_i$ 有 $K'_i$ 个独立工作区, 其中第 $k$ 个工作区的工作面积为 $a_{ik}$ (单位取 $100\text{m}^2$ ), 工作人数为 $s'_{ik}$ , 日均工作时长为 $t'_{ik}$ , 企业所在地的疫情风险等级为 $r_i$ , 据此估算该工作区的人员工作传播风险( $1 \leq k \leq K'_i$ ):

$$R_{ik}^{(W)} = b^{\beta \cdot r_i \log_2(1+s'_{ik} t'_{ik}/a_{ik})} - 1, \quad (2)$$

其中 $\beta$ 为一个系数, 对封闭工作区取值为1, 对半开放工作区(如敞篷车间)取值为0.5。

企业申报的人员信息还包括复工后的员工居住地区, 以及需要乘坐公共交通上下班的员工数量(不计自驾车人员)。设当地划分为 $K$ 个地区, 其中第 $k$ 个地区的疫情风险等级为 $r_k$ 。基于各企业 $E_i$ 的员工居住信息, 可统计出第 $d$ 日乘坐公交从第 $k$ 个地区到第 $k'$ 个地区的上班总人数 $s_{kk'}(d)$ 及平均交通时间 $t_{kk'}(d)$  ( $1 \leq d \leq 7; 1 \leq k \leq K; 1 \leq k' \leq K$ , 其中 $k$ 和 $k'$ 可以相等, 表示居住和上班地点在同一个区), 据此估算相应的人员公交传播风险:

$$R_{kk'}^{(P)}(d) = b^{\frac{r_k+r_{k'}}{2} \log_2(1+s_{kk'}(d)t_{kk'}(d)/\delta_{kk'})} - 1, \quad (3)$$

其中 $\delta_{kk'}$ 为上下班高峰期从第 $k$ 个地区和到第 $k'$ 个地区的公交班次。

值得一提的是, 上述疫情传播风险估算方法主要是基于浙江省疫情防控实践分析得出的。在不同环境下, 也可以使用其他的风险估算方法, 如其他数值计算模型<sup>[16]</sup>、马尔科夫链模型<sup>[17]</sup>、神经网络模型<sup>[18]</sup>、网络传播模型<sup>[19]</sup>等。这不会影响到问题模型和求解算法的整体框架。此外, 风险估算方法还可以在实际应用中动态完善<sup>[20]</sup>; 在生成了一个决策周期的复工复产规划方案并付诸实施后, 我们基于健康码风险跟踪等机制来比对风险估算的准确性, 根据反馈结果修正上述估算公式中的一些参数, 以提高下一决策周期的规划水平。

关于生产信息, 综合考虑当地的 $m$ 个生产产业 $\{I_1, I_2, \dots, I_m\}$ 。每个企业可能涉及一个或多个产业, 企业 $E_i$ 申请复工的产业集合记为 $\Phi_i$ ; 对应地, 与产业 $I_j$ 相关的企业集合记为 $\Psi_j$ 。每个企业 $E_i$ 对其申请复工的每个产业 $I_j \in \Phi_i$ 需填报如下信息:

- 企业 $E_i$ 从启动复工之日起到产业 $I_j$ 正式恢复生产所需的天数 $\tau_{ij}$ ;
- 企业 $E_i$ 的产业 $I_j$ 恢复生产后的日均产能 $c_{ij}$ ;
- 企业 $E_i$ 的产业 $I_j$ 所依赖的上游产业集合 $\Lambda_{ij}$ , 以及该产业单位产能所需的每个上游产业 $I_{j'}$ 的产能 $c'_{ijj'}$ 。

设当前决策周期开始之前, 当地每个产业 $I_j$ 的已有产能为 $o_j$ , 累积库存量为 $h_j$ (对运输业等服务型产业令库存始终为0)。如果企业 $E_i$ 在第 $x_i$ 日开始启动复工, 则其每个相关产业 $I_j$ 自 $x_i + \tau_{ij}$ 日起恢复产能 $c_{ij}$ 。那么在决策周期的第 $d$ 天( $1 \leq d \leq 7$ ), 产业 $I_j$ 预期达到的产能 $o_j(d)$ 为:

$$o_j(d) = o_j + \sum_{E_i \in \Psi_j} \text{sign}(x_i(d+1-x_i-\tau_{ij})) c_{ij}, \quad (4)$$

其中符号函数 $\text{sign}(\cdot)$ 对正数返回1, 对0或负数返回0。当 $x_i=0$ 时, 企业 $E_i$ 未批准复工, 其产能贡献为0; 当 $x_i>0$ 时, 企业 $E_i$ 自第 $x_i+\tau_{ij}$ 日起恢复产能, 那么在 $d>x_i+\tau_{ij}$ 时其产能贡献为 $c_{ij}$ 。

在决策前, 社会对产业 $I_j$ 的每日需求量预估为 $p_j(d)$ ; 由于复工产业新增了对上游产业的依赖, 故重新估算后的每日需求量为:

$$p'_j(d) = p_j(d) + \sum_{i=1}^n \sum_{I_{j'} \in \Phi_i} \text{sign}(x_i(d+1-x_i-\tau_{ij'})) c_{ij'} c'_{ij'j}, \quad (5)$$

基于产能和消耗,  $I_j$ 的库存量 $h_j(d)$ 更新如下:

$$h_j(0) = h_j, \quad (6)$$

$$h_j(d) = \max(0, h_j(d-1) + o_j(d-1) - p'_j(d-1)), \quad 1 \leq d \leq 7. \quad (7)$$

由此可估算每个产业每日的需求总满足率为:

$$\theta_j(d) = \frac{\min(h_j(d) + o_j(d), p'_j(d))}{p'_j(d)}。 \quad (8)$$

为定量评估企业产能对当地经济社会发展的作用,这里为每个产业 $I_j$ 设置一个权重 $w_j$ (目前我们的方法是按重要性由高到低划分为10档,每档产业的初始权重依次为20, 16, 12, 10, 8, 6, 4, 3, 2, 1,而后在问题中进行归一化,使所有产业的权重之和为1)。注意一些纯上游产业(如原材料产业)的权重可为0,其作用通过依赖其下游产业来体现。问题的目标函数是使得决策周期内各产业的需求总满足率最大化:

$$\max f(x) = \sum_{d=1}^7 \sum_{j=1}^m w_j \theta_j(d)。 \quad (9)$$

该问题需要满足如下约束条件:

- 每个产业所依赖的上游产业产能足够(即企业不能复工后又因为上游产业产能不够而停工):

$$h_j(d) + o_j(d) - p_j(d) \geq \sum_{i=1}^n \sum_{I_{j'} \in \Phi_i} \text{sign}(x_i(d+1-x_i-\tau_{ij'})) c_{ij'} c'_{ij'}, \quad 1 \leq d \leq 7; 1 \leq j \leq m; \quad (10)$$

- 每日外地返程员工不超过给定的上限:

$$\sum_{(i,k) \in \Omega(d)} s_{ik} \leq \hat{s}(d), \quad 1 \leq d \leq 7, \quad (11)$$

其中 $\Omega(d) = \{(i,k) | x_i > 0 \wedge x_i + \lfloor t_{ik}/24 \rfloor = d\}$ ,即所有批准复工复产企业中,员工正好在第 $d$ 天返回当地的 $(i,k)$ 元组集合;

- 在决策周期内,批准复工复产企业的返程传播风险之和不超过给定的上限:

$$\sum_{i=1}^n \sum_{k=1}^{K_i} R_{ik}^{(T)} \leq \hat{R}^{(T)}; \quad (12)$$

- 批准复工复产企业每个工作区的工作传播风险不超过给定的上限:

$$R_{ik}^{(W)} \leq \hat{R}^{(W)}, \quad 1 \leq i \leq n; 1 \leq k \leq K'_i; \quad (13)$$

- 每日公交传播风险不超过给定的上限:

$$R_{kk'}^{(P)}(d) \leq \hat{R}^{(P)}, \quad 1 \leq d \leq 7; 1 \leq k \leq K; 1 \leq k' \leq K; \quad (14)$$

其中约束(13)可以在企业申报时直接计算,如不满足则直接不批准复工复产。这些企业可以在问题求解之前从候选企业集合中删除。不过,如果后续企业所在地的疫情风险等级降低,该约束可能又变满足。其余约束则要在每一个问题的候选解上进行检查。

该问题被建模为一个非线性整数规划问题模型。解的每一维的取值范围在0~8之间,那么所有候选解的数量为 $8^n$ 。例如杭州市自2月8日起开始企业复工申请,当日申请的企业数量就达到29 814。此时问题的解空间会相当庞大,传统的数学规划法求解起来十分困难,需要设计高效的运筹优化算法来进行求解。

## 2 规划优化算法

本文提出了一个禁忌搜索算法<sup>[14,15]</sup>来求解上述企业复工复产规划问题。算法从一个初始解出发, 每次在解空间中搜索当前解周围的“邻居”解, 并选取一个最有可能改善搜索过程的“邻居”解来替换当前解, 从而不断迭代寻优。算法还通过一个禁忌表来记录最近执行过的搜索动作, 并在接下来的搜索过程中禁止重复这些动作, 从而降低多次重复搜索同一区域的概率。为提高搜索效率, 本文还设计了一个贪心策略来生成禁忌搜索算法的初始解, 在搜索过程中动态改变邻域大小, 以及在搜索陷入停滞时跳出局部最优。

### 2.1 贪心策略

贪心策略按产业权重优先的顺序来安排企业复工复产。对每个产能不能满足需求的产业 $I_j$ , 找出一个产能最大的企业, 如果该企业复工不违反风险约束, 且上游产业产能满足要求, 则安排该企业复工复产, 否则继续找下一个产能最大的企业。Algorithm 1给出了贪心策略的算法伪代码, 其调用了Algorithm 2中的子过程来尝试安排企业复工复产, 注意该子过程又递归调用了自身来尝试安排上游产业企业复工复产以满足产能依赖。

**Algorithm 1:** Greedy procedure for creating a solution to the problem.

---

```

1 Sort all industries  $\{I_1, I_2, \dots, I_m\}$  in non-increasing order of  $w_j$ ; // 产业按优先
   级排序
2 Let  $d = 1$ ,  $R^{(T)} = 0$ ,  $\mathbf{E} = \{E_1, E_2, \dots, E_n\}$ ;
3 while  $d \leq 7$  do
4   Let  $j = 1$ ,  $R^{(P)}(d) = 0$ ;
5   while  $j \leq m$  do
6     Let  $o_j(d) = o_j$ ,  $\mathbf{E}' = \{E_i \in \mathbf{E} | \tau_{ij} < d\}$ ;
7     while  $h_j(d) + o_j(d) < p'_j(d) \wedge \mathbf{E}' \neq \emptyset$  do
8       Let  $E_{i^*} = \max_{E_i \in \mathbf{E}'} c_{ij}$ ; // 选择一个能为  $I_j$  贡献最大产能的企业
9       Call Algorithm 2 to resume the production of  $E_{i^*}$  at the  $(d - \tau_{i^*j})$ th day;
10      if the call of Algorithm 2 returns false then Remove  $E_{i^*}$  from  $\mathbf{E}'$ ;
11       $j \leftarrow j + 1$ ;
12     $d \leftarrow d + 1$ ;
13 return  $x$ .

```

---

### 2.2 禁忌搜索算法

禁忌搜索算法以上述贪心策略生成的解为初始解 $\mathbf{x}$ , 而后不断对当前解进行邻域搜索, 每个邻居解通过随机改变某一维 $x_i$ 来获得, 故每个解最多有 $7n$ 个邻居解。算法每次搜索其中的 $K_N$ 个邻居解( $K_N$ 为一个控制参数)。考虑到 $x_i$ 取值的不同含义, 对其进行改变的具体方式为: 如 $x_i = 0$ , 则设 $x_i$ 为1~7之间的一个随机值; 如 $x_i > 0$ , 则 $x_i$ 有0.5的概率变为0, 另外0.5的概率变为1~7之间另一个整数值。在 $K_N$ 个邻居解中, 最好的一个(未被禁忌或好于 $\mathbf{x}$ )被设为新的当前解 $\mathbf{x}$ , 其改变的维度 $i$ 被加入到禁忌表, 而后继续下一次迭代搜索。禁

---

**Algorithm 2:** Sub-procedure for determining the production resumption of  $E_i$  at the  $d$ th day.

---

**Input:**  $i, d, R^{(T)}, \mathbf{E}$  (the set of enterprises that have not been approved)

**Output:** whether the resumption is successful

/\* 检查批准 $E_i$ 复工后是否会违反风险约束 \*/

```

1 Temporally approve  $E_i$  at the  $d$ th day;
2 for  $d' = d$  to  $\gamma$  do
3   Update the number of employees at the  $d'$ th day;
4   if  $s(d') > \hat{s}(d')$  then return false;
5   foreach pair  $k, k'$  in the current region do
6     Recalculate  $R_{kk'}^{(P)}(d')$  according to Eq.(3);
7     if  $R_{kk'}^{(P)}(d') > \hat{R}^{(P)}$  then return false;
8   if  $R^{(T)} + \sum_{k=1}^{K_i} R_{ik}^{(T)} > \hat{R}^{(T)}$  then return false;
9   /* 检查 $E_i$ 依赖的上游产业产能是否都得到满足 */ / /
10  for  $I_j \in \Phi_i$  do
11    for  $I_{j'} \in \Lambda_{ij}$  do
12      Let  $\mathbf{E}' = \{E_{i'} \in \mathbf{E} | \tau_{i'j'} < d\}$ ;
13      while  $h_{j'}(d) + o_{j'}(d) < p'_{j'}(d) \wedge \mathbf{E}' \neq \emptyset$  do
14        Let  $E_{i^*} = \max_{E_{i'} \in \mathbf{E}'} c_{i'j'}$ ;
15        Call Algorithm 2 to resume the production of  $E_{i^*}$  at the  $(d - \tau_{i'j'})$ th day;
16        if the recursive call of Algorithm 2 returns false then Remove  $E_{i^*}$  from
17           $\mathbf{E}'$ ;
18         $o_j(d) \leftarrow o_j(d) + c_{i^*j};$ 
19        /* 成功安排复工复产 */ / /
20       $x_{i^*} \leftarrow d;$ 
21      for  $d' = d$  to  $\gamma$  do update the number of employees at the  $d'$ th day;
22       $R^{(T)} \leftarrow R^{(T)} + \sum_{k=1}^{K_i} R_{ik}^{(T)};$ 
23      Remove  $E_{i^*}$  from  $\mathbf{E}$ ;
24      return true.

```

---

忌表有一个预设长度  $tlen$ , 如果其中的元素数量已满, 那么新加入元素时, 从表中移除最早加入的元素。

Algorithm 3给出了禁忌搜索算法的伪代码。

---

**Algorithm 3:** Tabu search algorithm for the problem.

---

```

1 Call Algorithm 1 to create an initial solution  $\mathbf{x}$ ;
2 Create an empty tabu list  $T$ ;
3 Let  $\mathbf{x}^* = \mathbf{x}$ ,  $k_p = 0$ ,  $t = 0$ ;
4 while  $t \leq t^{\max}$  do
5   Create a neighboring solution  $\mathbf{x}'$  by randomly changing a dimension  $i$  of  $\mathbf{x}$ ,
   where  $i \notin T$ ;
6   for  $k = 1$  to  $K_N - 1$  do
7     Create a neighboring solution  $\mathbf{x}''$  by randomly changing a dimension  $i$  of  $\mathbf{x}$ ;
8     if  $i \notin T \vee f(\mathbf{x}'') > f(\mathbf{x})$  then
9       if  $f(\mathbf{x}'') > f(\mathbf{x}')$  then  $\mathbf{x}' \leftarrow \mathbf{x}''$ ;
10     $\mathbf{x} \leftarrow \mathbf{x}'$ ;
11    Add the changing dimension  $i$  of  $\mathbf{x}'$  to  $T$ ;
12    if  $|T| > tlen$  then remove the first element of  $T$ ;
13    if  $f(\mathbf{x}) > f(\mathbf{x}^*)$  then
14       $\mathbf{x}^* \leftarrow \mathbf{x}$ ;
15       $k_p \leftarrow 0$ ;
16    else if  $++k_p = K_p$  then
17      set  $\mathbf{x}$  to a new random solution;
18       $k_p \leftarrow 0$ ;
19     $t \leftarrow t + 1$ ;
20    Update  $K_N$  according to Eq.(15);
21 return  $\mathbf{x}^*$ .

```

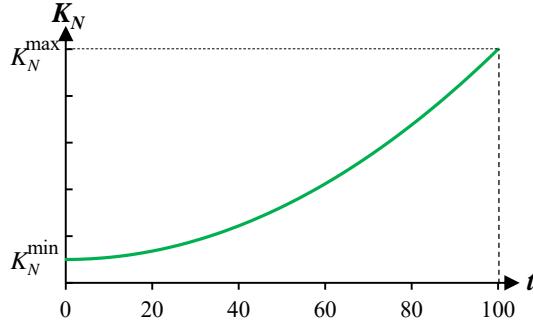
---

上述为标准的禁忌搜索算法流程。但在实际应用中, 其性能并不总是令人满意, 故本文提出了两种改进策略: 一是动态改变邻域大小  $K_N$ , 算法开始时  $K_N$  值较小, 当前解移动较快, 更利于全局搜索; 后期  $K_N$  值逐渐变大, 以加强在当前解周围的局部搜索<sup>[21]</sup>。设算法最大迭代次数为  $t_{\max}$ , 本文设置  $K_N$  随迭代次数非线性递增:

$$K_N(t) = K_N^{\min} + \left( \frac{t}{t^{\max}} \right)^2 (K_N^{\max} - K_N^{\min}), \quad (15)$$

其中  $K_N^{\min}$  和  $K_N^{\max}$  分别为邻域大小的最小值和最大值。图2给出了当  $t^{\max} = 100$  时  $K_N$  的变化曲线图。

二是设置一个阶段性迭代次数上限  $K_p$ , 如果算法从某个解出发迭代搜索  $K_p$  次后, 当前最优解  $\mathbf{x}^*$  仍没有改善, 则说明当前搜索区域价值可能不高; 此时把当前解  $\mathbf{x}$  重设为一个新的随机解, 从新解出发继续搜索, 以探索解空间中新的区域。在实际应用中, 企业复工

图2  $K_N$  随迭代次数变化的曲线图

复产申请的批准率往往会很低,那么采用完全随机方式生成的新解基本上都是不可行解。本文的方式是参照当前最优解 $\mathbf{x}^*$ 来生成随机解,设 $\mathbf{x}^*$ 的非0维度数量为 $D^*$ ,这表示n家企业中只有 $D^*$ 家被批准复工复产;算法生成新的随机解时,也仅对D维设置为非0的随机值,其余维度都设为0, $D$ 取 $D^*/2$ 到 $D^*$ 之间的一个随机数,即新解中批准复工复产的企业数量在 $D^*/2$ 到 $D^*$ 之间。

### 2.3 不可行解的处理

可行解的邻居解未必也是可行解。在本算法中,如果对当前解 $\mathbf{x}$ 搜索到的邻居解都是不可行的,那么也允许当前解转到不可行解上,因为很多时候经过一些不可行解后会发现更优的可行解。在对不可行解进行评估时,本文采用惩罚函数法,通过下列公式分别计算解 $\mathbf{x}$ 违反约束的程度:

$$\begin{aligned} V_c(\mathbf{x}) = & \sum_{d=1}^7 \sum_{j=1}^m \left[ \left( \sum_{i=1}^n \sum_{I_{j'} \in \Phi_i} \text{sign}(x_i(d+1-x_i-\tau_{ij'})) c_{ij'} c'_{ij'} \right) \right. \\ & \left. + p_j(d) - h_j(d) - o_j(d) \right], \end{aligned} \quad (16)$$

$$V_s(\mathbf{x}) = \sum_{d=1}^7 \left[ \left( \sum_{(i,k) \in \Omega(d)} s_{ik} \right) - \hat{s}(d) \right], \quad (17)$$

$$V_T(\mathbf{x}) = \left( \sum_{i=1}^n \sum_{k=1}^{K_i} R_{ik}^{(T)} \right) - \hat{R}^{(T)}, \quad (18)$$

$$V_P(\mathbf{x}) = \sum_{d=1}^7 \sum_{k=1}^K \sum_{k'=1}^K [R_{kk'}^{(P)}(d) - \hat{R}^{(P)}]. \quad (19)$$

在评估不可行解时,将违反约束的惩罚函数值纳入到目标函数中:

$$\begin{aligned} f(\mathbf{x}') = & f(\mathbf{x}) - M_1 \max(V_c(\mathbf{x}), 0) - M_2 \max(V_s(\mathbf{x}), 0) - M_3 \max(V_T(\mathbf{x}), 0) \\ & - M_3 \max(V_P(\mathbf{x}), 0), \end{aligned} \quad (20)$$

其中 $M_1$ , $M_2$ 和 $M_3$ 是三个正系数,本文分别设为1 000(产能转换为经济效益值,并以百万元为单位),5 000和200。在算法中,邻域解之间基于带惩罚的目标函数值进行比较(伪代码第8和第9行);但只有可行解才能用于更新当前最优解(伪代码第13行)。

### 3 数值计算结果

针对今年二月份后两周杭州、绍兴、金华三地的企业复工复产规划需求, 我们分别构建了问题实例, 并应用算法进行求解。6个问题实例按规模 $n$ 从小到大排列, 其基本特征如表1所示, 其中第4列表示各企业的员工来源地区的平均数量, 第5列表示各来源地区的平均员工数量, 第6列表示各企业关联产业的平均数量, 第7列表示各企业产业关联上游产业的平均数量。

表1 计算问题实例

ID	$n$	$m$	$\frac{\sum_{i=1}^n K_i}{n}$	$\frac{\sum_{i=1}^n \sum_{k=1}^{K_i} s_{ik}}{\sum_{i=1}^n K_i}$	$\frac{\sum_{i=1}^n  \Phi_i }{n}$	$\frac{\sum_{i=1}^n \sum_{I_j \in \Phi_i}  \Lambda_{ij} }{\sum_{i=1}^n  \Phi_i }$
#1	7 953	969	25.31	6.70	5.83	2.14
#2	10 306	1 608	26.27	8.25	4.46	2.46
#3	15 541	5 010	30.77	11.53	7.26	3.52
#4	18 100	6 029	31.08	10.96	6.74	4.15
#5	22 905	4 760	30.72	10.60	5.27	2.75
#6	27 032	5 833	29.12	11.19	4.81	3.24

为验证算法性能, 在每个问题实例上, 分别运行CPLEX整数规划求解器(版本12.6, 开启自动剪枝), 标准禁忌搜索算法(邻域大小固定, 不重置当前解, 记为TS), 以及本文提出的改进禁忌搜索算法(记为ITS)。算法有关参数设置为: 禁忌表预设长度 $tlen=20$ , 标准禁忌搜索算法邻域大小 $K_N=20$ , 改进禁忌搜索算法邻域大小范围 $K_N^{\min}=5$ ,  $K_N^{\max}=n/200$ , 重置当前解的最大无改进迭代次数 $K_p=30$ 。禁忌搜索算法的重复实验次数为30。计算环境为Intel i7-6500 2.5GHz CPU, 8GB DDR4内存, Windows 7操作系统。算法代码使用Visual C# 2015实现。根据实际应用求解需求, 每个算法的运行时间不超过90分钟。

表2给出了每个问题实例上CPLEX、贪心策略及两个禁忌搜索算法求得解的目标函数值(括号内的数字为30次实验值的标准差)。图3中还比较了各算法的解的目标函数值及其分布(其中Q1表示前25%, Q3表示前75%)。在每个问题实例上, ITS均取得了显著优于其他算法的结果。使用贪心策略直接得到的解质量较低, 其目标函数值大约只有ITS求解结果的10%~20%, 这说明仅按产业重要性一步步安排复工复产的效果并不好, 如优先安排了权重最高的产业, 很可能占用了过多的复工人员名额, 导致多个其他产业难以满足需求; 而问题的全局最优解往往要求在各个产业之间达到较好的平衡。

表2 计算实验结果

ID	CPLEX	Greedy	TS	ITS
#1	2 416	662	3 197 (58.42)	3 375 (32.82)
#2	1 832	696	4 861 (71.21)	5 783 (47.39)
#3	7 052	2 546	17 730 (397.25)	24 226 (272.50)
#4	3 779	1 938	16 925 (521.90)	21 781 (435.66)
#5	4 510	1 301	13 768 (679.82)	23 896 (362.85)
#6	2 327	978	11 790 (825.00)	26 920 (306.06)

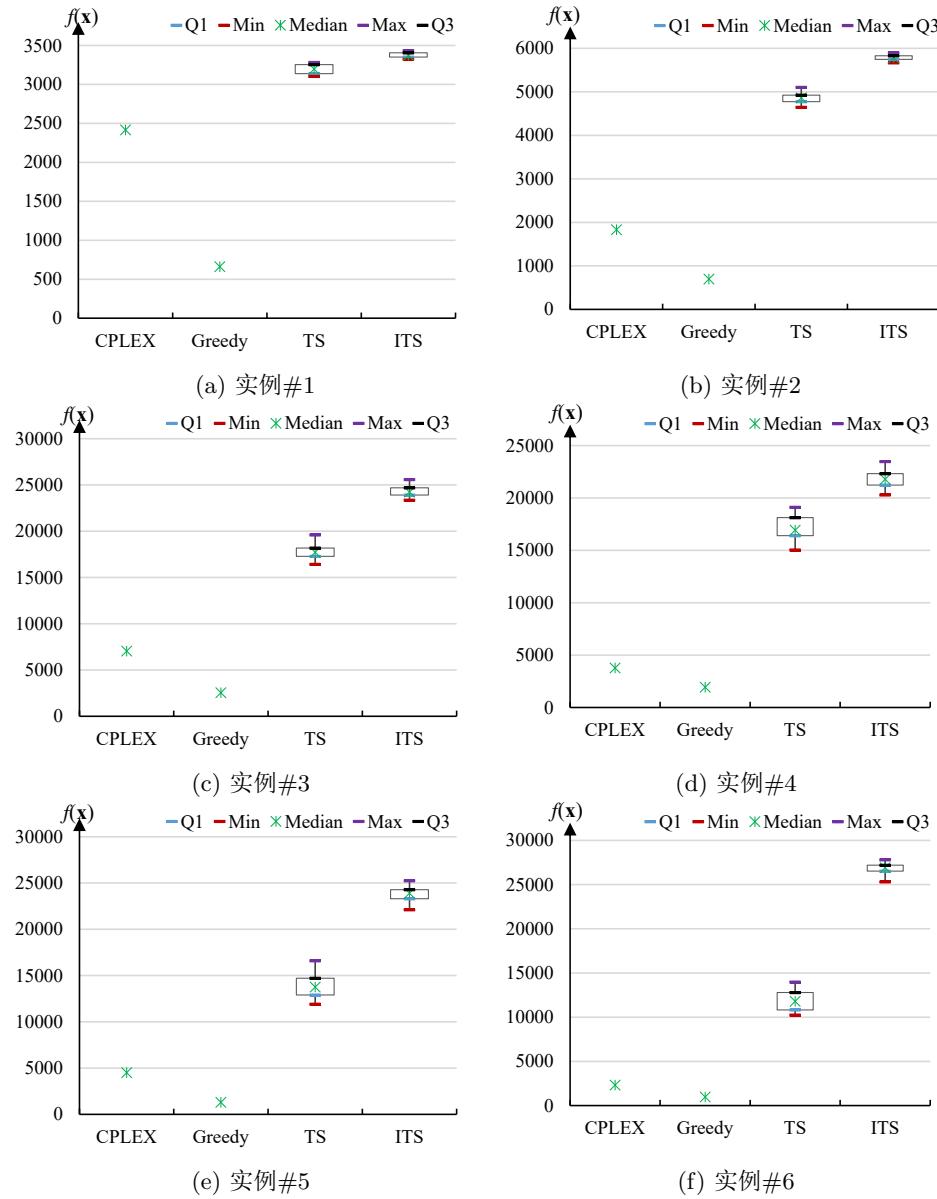


图3 算法实验结果比较图

其他三个算法均能在贪心解的基础上有所提升。其中CPLEX的求解结果仅在规模最小的实例#1上大大优于贪心解，在其他实例上相对于贪心解的优势比较有限，这是因为CPLEX的搜索效率不高，随着解空间的快速增长，它在规定的运行时间内只能搜索解空间中很小的一部分。相比较而言，禁忌搜索算法表现出了很高的搜索效率，而采用了改进策略的ITS算法性能又显著优于标准的TS算法。在实例#1上，ITS算法求得的目标函数值比TS算法提高了约5%；在实例#2、#3和#4上，ITS算法的性能提升在20%~37%之间；而在更大规模的实例#5和#6上，ITS算法的性能提升分别达到了74%和128%，这充分说明了本文所设计的两种改进策略在求解问题(特别是大规模问题)时所发挥的作用。此外，在

后三个问题实例上, 标准TS算法30次运行得到的目标函数值分布区间较大, 而ITS算法得到的目标函数值差距较小, 这说明ITS算法也具有较好的稳定性。

计算得出的复工复产方案与实际应用之间总是会有一定的偏差, 如对社会需求量的预估可能不够准确、企业员工返厂可能会有延迟等。为检验模型和算法结果的有效性, 我们统计了绍兴、金华、杭州三个地区在2月18日至3月2日复工复产期间主要产业产能的实际平均社会需求满足率(包含了初始权重为20, 16, 12, 10, 8的各项产业, 它们在当地经济社会中占主导地位; 初始权重小于8的产业重要性普遍较低, 且数量较多而不便统计), 并将其与本文方法计算出的需求满足率进行比较, 结果分别显示在图4, 图5和图6中。从结果中可以看出, 在统计的时间段内, 各地区主要产业的产能均稳步上升, 这说明了本文提出的复工复产规划方法的有效性。总的说来, 计算结果和实际结果的偏差不大, 且大致趋势比较吻合, 这验证了本文模型中产能计算方法的合理性。相对来说, 安排复工复产的早期误差较大, 这是因为早期生产力量薄弱, 受一些意外干扰因素的影响较大; 随着各产业产能的逐步恢复, 产业链上下游的供给/需求不断趋向合理平衡, 整体抗干扰能力增强, 所以后期误差不断减小。在大部分时间段内, 实际达到的需求满足率低于计算值, 因为受到干扰后的实际产能无法达到根据模型计算出的理想结果; 但后期有很多企业生产恢复良好, 会在预期申报的基础上加大产能, 因此也出现了一些需求满足率高于计算值的情况。

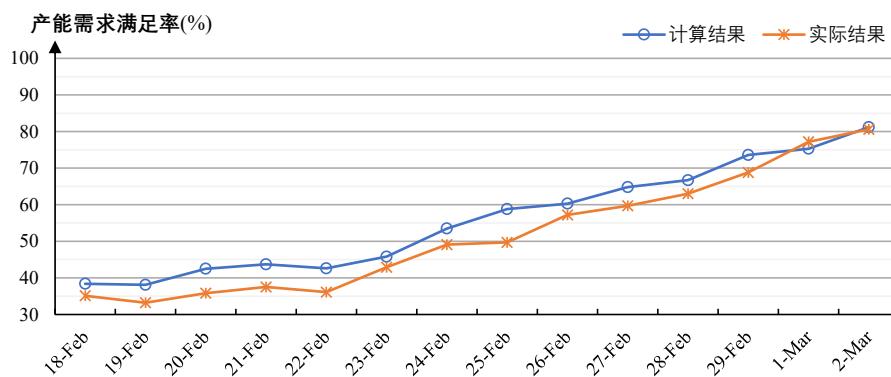


图4 绍兴地区主要产业产能的社会需求满足率(本文方法计算结果与实际结果的比较)

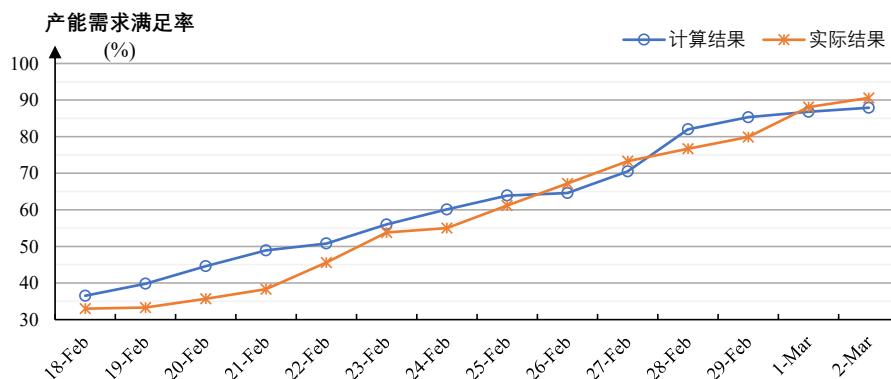


图5 金华地区主要产业产能的社会需求满足率(本文方法计算结果与实际结果的比较)

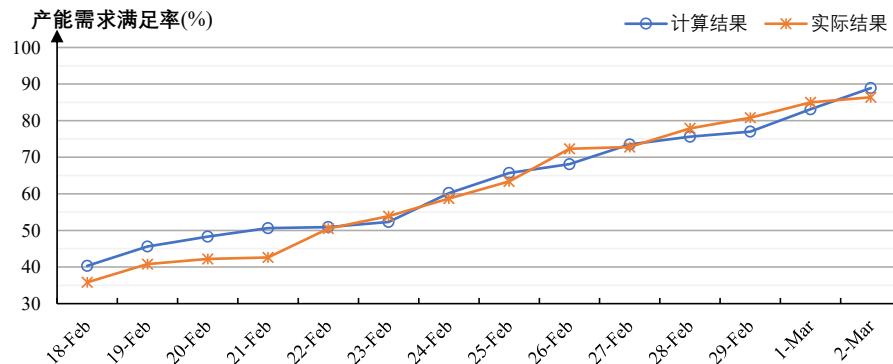


图6 杭州地区主要产业产能的社会需求满足率(本文方法计算结果与实际结果的比较)

## 4 结论和展望

疫情防控关乎生命, 复工复产关乎生计。为应对疫情条件下企业复工复产规划这一挑战, 本文提出了一个整数规划问题模型, 其目的是要在不违反疫情传播风险等约束下, 在大量候选企业中选择一部分并安排其复工复产顺序, 使得所有相关产业的综合产能需求满足率最大化。模型中还包括了疫情条件下企业员工返程传播风险、工作传播风险、上下班公交传播风险的估算方法。为有效求解该优化问题, 本文提出了一个改进的禁忌搜索算法, 它综合使用贪心策略、可变邻域大小和重新初始化等机制来提高搜索效率。本文提出的模型和方法应用于杭州、绍兴、金华等地区的企业复工复产规划问题实例, 计算结果验证了算法的效率。

本文仅研究了企业复工复产的规划问题。在批准企业复工复产后, 还需要将“精密智控”扩展到对企业防疫、金融、水电等服务的精准调度, 对员工日常生活的全面保障, 以及对员工健康风险的精确管控, 以确保复工复产效果, 促进经济社会持续健康发展, 其中同样包含一系列复杂的优化规划调度问题, 需要应用各种高效的优化算法来进行求解, 有关内容将在后续论文中作进一步研究。这些研究对于巩固疫情防控成果、化解疫情不利影响、保持我国经济长期向好的基本面具有重要意义。

## 参 考 文 献

- [1] 我省召开疫情防控工作第二十九场新闻发布会提高产能恢复率引才招商“云上见” [N/OL]. 浙江日报, 2020-02-27[2020-03-17]. [http://www.zj.gov.cn/art/2020/2/27/art\\_1228996602\\_42013277.html](http://www.zj.gov.cn/art/2020/2/27/art_1228996602_42013277.html).
- [2] Hung Y F, Leachman R C. A production planning methodology for semiconductor manufacturing based on iterative simulation and linear programming calculations [J]. *IEEE Transactions on Semiconductor Manufacturing*, 1996, **9**(2): 257-269.
- [3] 黄贵东, 丁文英. 基于线性规划的生产计划优化研究 [J]. 物流技术, 2006, **2**: 83-84
- [4] Sawik T. Integer programming approach to production scheduling for make-to-order manufacturing [J]. *Mathematical and Computer Modelling*, 2005, **41**(1), 99-118.
- [5] de Kruijff J T, Hurkens C A J, de Kok T G. Integer programming models for mid-term production planning for high-tech low-volume supply chains [J]. *European Journal of Operational Research*, 2018, **269**(3): 984-997.

- [6] Chen K, Ji P. A mixed integer programming model for advanced planning and scheduling (APS) [J]. *European Journal of Operational Research*, 2007, **181**(1): 515-522.
- [7] Pochet Y, Wolsey L A. *Production Planning by Mixed Integer Programming* [M]. New York: Springer, 2006.
- [8] 陈可嘉, 叶舒婷. 面向供应链的高级计划与排程的混合整数规划模型 [J]. 中国机械工程, 2012, **23**(14): 1688-1692.
- [9] Fleten S E, Kristoffersen T K. Short-term hydropower production planning by stochastic programming [J]. *Computers & Operations Research*, 2008, **35**(8): 2656-2671.
- [10] Kazemi Zanjani M, Noureldath M, Ait-Kadi D. A multi-stage stochastic programming approach for production planning with uncertainty in the quality of raw materials and demand [J]. *International Journal of Production Research*, 2010, **48**(16): 4701-4723.
- [11] Jamalnia A, Soukhakian M A. A hybrid fuzzy goal programming approach with different goal priorities to aggregate production planning [J]. *Computers & Industrial Engineering*, 2009, **56**(4): 1474-1486.
- [12] da Silva A F, Marins F A S. A fuzzy goal programming model for solving aggregate production-planning problems under uncertainty: A case study in a brazilian sugar mill [J]. *Energy Economics*, 2014, **45**: 196-204.
- [13] Garey M R, Johnson D S. *Computers and Intractability: A Guide to the Theory of NP-Completeness* [M]. San Francisco: W.H. Freeman, 1979.
- [14] Glover F. Tabu search—part I [J]. *ORSA Journal on Computing*, 1989, **1**(3): 190-206.
- [15] Glover F. Tabu search—part II [J]. *ORSA Journal on Computing*, 1990, **2**(1): 4-32.
- [16] Zhu S, Srebric J, Spengler J D, et al. An advanced numerical model for the assessment of airborne transmission of influenza in bus microenvironments [J]. *Building and Environment*, 2012, **47**: 67-75.
- [17] 张玉, 刘新新, 蔡传锋, 等. 基于交通网络传递的传染病风险传播模型研究 [J]. 计算机与数字工程, 2017, **45**(12): 2359-2363.
- [18] Song Q, Zheng Y J, Shen W G, et al. Tridirectional transfer learning for predicting gastric cancer morbidity [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2020, doi:10.1109/TNNLS.2020.2979486.
- [19] 王霞, 唐三一, 陈勇, 等. 新型冠状病毒肺炎疫情下武汉及周边地区何时复工? 数据驱动的网络模型分析 [J]. 中国科学: 数学, 2020, **50**: 1-10.
- [20] Song Q, Zheng Y, Huang Y, et al. Emergency drug procurement planning based on big-data driven morbidity prediction [J]. *IEEE Transactions on Industrial Informatics*, 2019, **15**(12): 6379-6388.
- [21] Hansen P, Mladenović N, Todosijević R, et al. Variable neighborhood search: basics and variants [J]. *EURO Journal on Computational Optimization*, 2017, **5**(3): 423-454.