

求解拉丁方补全问题的 交换松弛局部搜索算法

吕志鹏

zhipeng.lv@hust.edu.cn

2024-10-24

简介

Introduction

问题

Problem

算法

Algorithm

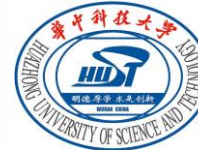
实验

Experiment

结论

Conclusion

Huazhong University of Science & Technology
Université de Picardie Jules Verne

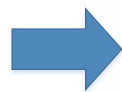


简介

• 问题概述

- 拉丁方补全问题 (Latin Square Completion, LSC) 是一个经典谜题
 - 解谜类小游戏是AI算法的常用测试基准
- LSC要求在空白单元格中填写数字，使得每行每列中均无重复数字
 - 将所有单元格着色，以确保同一行/列中没有两个单元格颜色相同

0	1		3	4	5	6
1	2	5	6	0	3	4
4	6					2
5	4					0
	5					
6	0					3
2	3	0	4		6	5

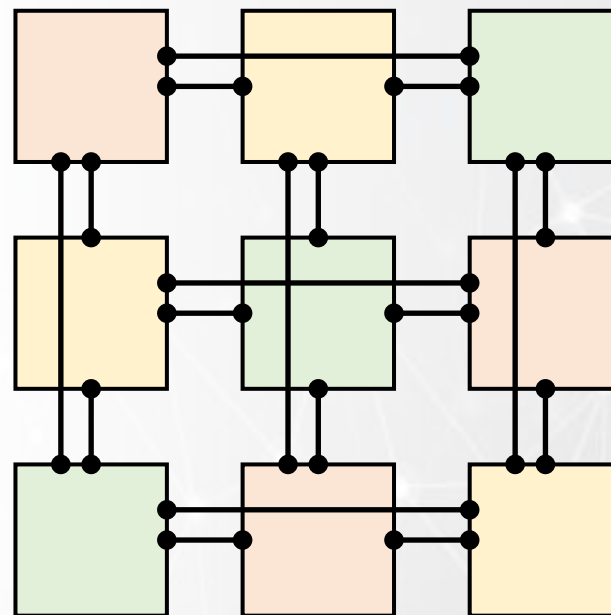
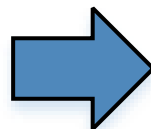


0	1	2	3	4	5	6
1	2	5	6	0	3	4
4	6	3	0	5	1	2
5	4	6	1	3	2	0
3	5	4	2	6	0	1
6	0	1	5	2	4	3
2	3	0	4	1	6	5

• 难点与挑战

- LSC 可被归约为带预着色的图着色问题 (GCP)
 - 单元格→ 顶点
 - 同行或同列的约束→ 边
 - 数字→ 颜色
- 显然LSC也是NP完全的

0	1	2
1	2	0
2	0	1



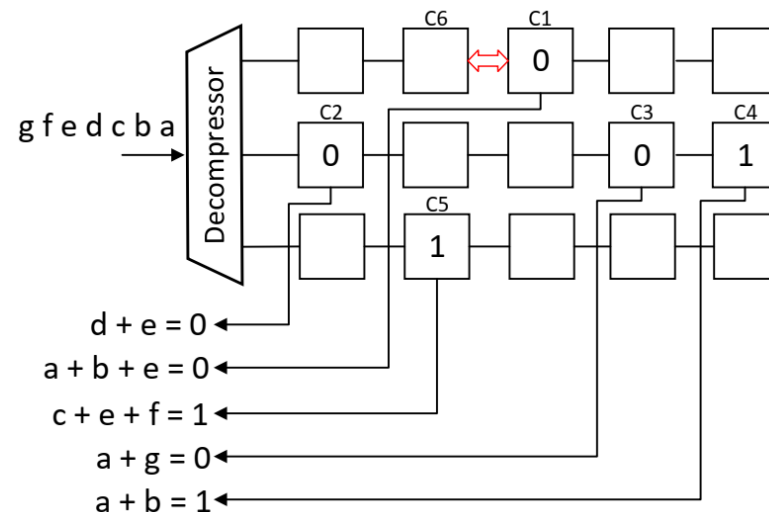
• 相关问题的应用价值

• 经典应用场景

- 电力线通信设计
- 光网络交换机配置
- 误差校正码设计

• 所提算法经简单改造可求解其他实际应用问题

- 数字芯片测试 (DFT) 中的扫描链重新排序



IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 50, NO. 6, JUNE 2004

Permutation Arrays for Powerline Communication and Mutually Orthogonal Latin Squares

Charles J. Colbourn, Torleiv Kløve, *Fellow, IEEE*, and Alan C. H. Ling

Abstract—We develop a connection between permutation arrays that are used in powerline communication and well-studied combinatorial objects, mutually orthogonal latin squares (MOLS). From this connection, many new results on permutation arrays can be obtained.

Index Terms—Doubly resolvable design, mutually orthogonal latin squares (MOLS), permutation array, permutation code, powerline communications.

Algorithmica (1999) 24: 128–138

Algorithmica
© 1999 Springer-Verlag New York Inc.

Approximating Latin Square Extensions¹

S. R. Kumar,² A. Russell,³ and R. Sundaram⁴

Abstract In this paper we investigate the problem of computing the maximum number of entries which can be added to a partially filled latin square. The decision version of this question is known to be NP-complete. We present two approximation algorithms for the optimization version of this question. We first prove that the greedy algorithm achieves a factor of $\frac{1}{2}$. We then use insights derived from the linear relaxation of an integer program to obtain an algorithm based on matchings that achieves a better performance guarantee of $\frac{1}{2}$. These are the first known polynomial-time approximation algorithms for the latin square completion problem that achieve nontrivial worst-case performance guarantees. Our study is motivated by applications to lightpath assignment and switch configuration in wavelength routed multihop optical networks.

IEEE TRANSACTIONS ON COMPUTERS, VOL. 64, NO. 7, JULY 2015

Non-Binary Orthogonal Latin Square Codes for a Multilevel Phase Charge Memory (PCM)

Kazuteru Namba, *Member, IEEE* and
Fabrizio Lombardi, *Fellow, IEEE*

Abstract—This manuscript proposes non-binary orthogonal Latin square (OLS) codes that are amenable to a multilevel phase change memory (PCM). This is based on the property that the proposed $(n$ symbols, k symbols) t -symbol error correcting code uses the same H matrix as an $(n$ bits, k bits) binary t -bit error correcting OLS code. The new codes are shown to have a shorter check bit length and better probability in encoding/decoding than conventional binary OLS codes. Extensive results are provided for assessment and comparison. The proposed codes are also shown to be always better than the matrix codes, i.e. independently of the metric and the parameters employed in the comparison.

• 主流求解方案

• 求解框架

- LSC 可被归约为带预着色的图着色问题 (GCP)

• 化简规则

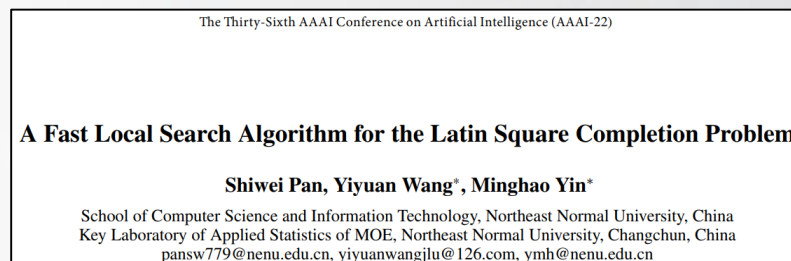
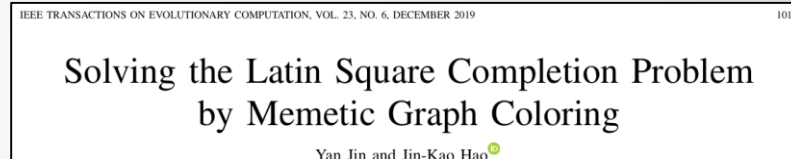
- 若一个单元格只能填一个数字，则其所有同行同列的单元格都不能填该数字
- 若某一行/列只有一个单元格可以填某个数字，则该数字必须被该单元格填写

• 邻域结构

- 改变一个单元格内填充的数字

• 疏散性策略

- 扰动
- 交叉



1948

至今

大约75年

简介

Introduction

问题

Problem

算法

Algorithm

实验

Experiment

结论

Conclusion

Huazhong University of Science & Technology
Université de Picardie Jules Verne



拉丁方补全问题

• 拉丁方补全问题的形式化定义

• 原始判定问题

• 已知

- 一个大小为 $N \times N$ 的正方形表格，其单元格集合记为 C
 - 可填写的数字范围为 $D = \{1, 2, \dots, N\}$ (也称作颜色域)
 - 一个三元组集合 P ，其元素为 $\langle \text{行}, \text{列}, \text{数字} \rangle$ ，规定若干固定填写某数字的单元格

• 决策

- 单元格到数字的映射 $C \rightarrow D$ ，即确定决策变量 $x_{ij} \in \{1, 2, \dots, N\}, \forall i, j \in \{1, 2, \dots, N\}$ 的取值

• 优化目标

- 0

• 约束

- 每个数字在每行/列中必须恰好出现一次 (all-different约束)
 - $x_{ij} \neq x_{ik}, \forall i, j, k \in \{1, 2, \dots, N\}$
 - $x_{ij} \neq x_{kj}, \forall i, j, k \in \{1, 2, \dots, N\}$
- 某些单元格需固定填写某个数字
 - $x_{ij} = k, \forall \langle i, j, k \rangle \in P$

• 拉丁方补全问题的形式化定义

• 松弛列冲突转换为优化问题

• 已知

- 一个大小为 $N \times N$ 的正方形表格，其单元格集合记为 C
 - 可填写的数字范围为 $D = \{1, 2, \dots, N\}$
 - 一个三元组集合 P ，其元素为 $\langle \text{行}, \text{列}, \text{数字} \rangle$ ，规定若干固定填写某数字的单元格

• 决策

- 单元格到数字的映射 $C \rightarrow D$ ，即确定决策变量 $x_{ij} \in \{1, 2, \dots, N\}, \forall i, j \in \{1, 2, \dots, N\}$ 的取值

• 优化目标

- 最小化每列的颜色重复次数

$$\sum_{j=1}^N \sum_{i=1}^N \sum_{k=1}^N x_{ij} \odot x_{kj}$$

• 约束

- 每个数字在每行中必须恰好出现一次 (all-different约束)

$$x_{ij} \neq x_{ik}, \forall i, j, k \in \{1, 2, \dots, N\}$$

文献中的算法均把本约束也一并松弛

- 某些单元格需固定填写某个数字

$$x_{ij} = k, \forall \langle i, j, k \rangle \in P$$

简介

Introduction

问题

Problem

算法

Algorithm

实验

Experiment

结论

Conclusion

基于交换松弛的局部搜索

• 基于交换松弛的局部搜索 (SRLS) 的整体框架与关键技术

基于推理规则的颜色域缩减

无行冲突的初始解生成

基于行内交换的邻域结构

颜色域松弛

次级目标函数

自适应重启机制

初始解生成阶段

迭代改进阶段

Algorithm 1 The main framework of the SRLS algorithm

Input: A Latin square graph $G = (V, E)$

Output: A coloring solution C

```
1: A coloring solution  $C \leftarrow \text{Initialize}(G)$  // Algorithm 2
2: The best found solution  $C^* \leftarrow C$ 
3: while elapsed time < time limit do
4:   Select the best move  $m(u, v)$  based on the neighborhood evaluation and tabu strategy
5:    $C \leftarrow C \oplus m(u, v)$  // swap the colors of  $u$  and  $v$ 
6:   if  $|CE(C)| = 0$  then
7:     return  $C$ 
8:   else if  $|CE(C)| \leq |CE(C^*)|$  then
9:      $C^* \leftarrow C$  // replace the best found solution
10:  else
11:     $C \leftarrow \text{AdaptiveRestart}(C, C^*)$  // Algorithm 3
12: return  $C^*$ 
```

$CE(C)$ 为着色方案 C 的冲突边集合

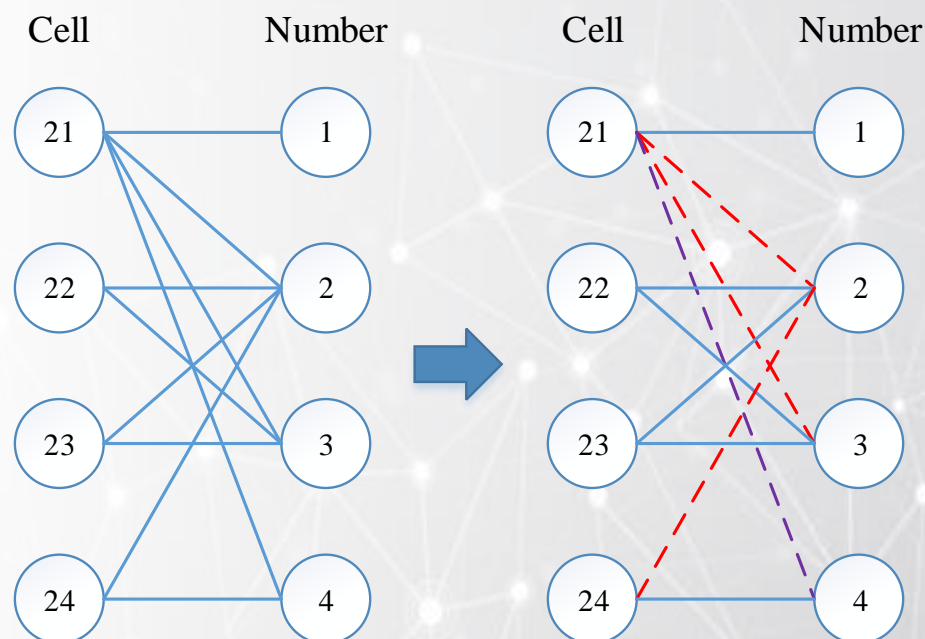
• 约简规则

- 从二分图的视角观察all-different约束与每个单元格的颜色域
 - 寻找从 N 个单元格到 N 个数字的完美匹配 (双射)
 - 单元格与其颜色域中每个颜色连一条边
 - 有缺失的颜色域对应非完全二分图
 - 部分边必须被选中: 固定单元格 (原始约束)、不应填所有其他数字 (约束传播)
 - 部分边不应被选中: 同行或同列有其他单元格必须填该数字 (约束传播)

{2,3}	4	1	{2,3}
{1,2,3,4}	{2,3}	{2,3}	{2,4}
{2,3}	{2,3}	4	1
{2,3,4}	1	{2,3}	{2,3}



{2,3}	4	1	{2,3}
{1,4}	{2,3}	{2,3}	{4}
{2,3}	{2,3}	4	1
{2,3,4}	1	{2,3}	{2,3}



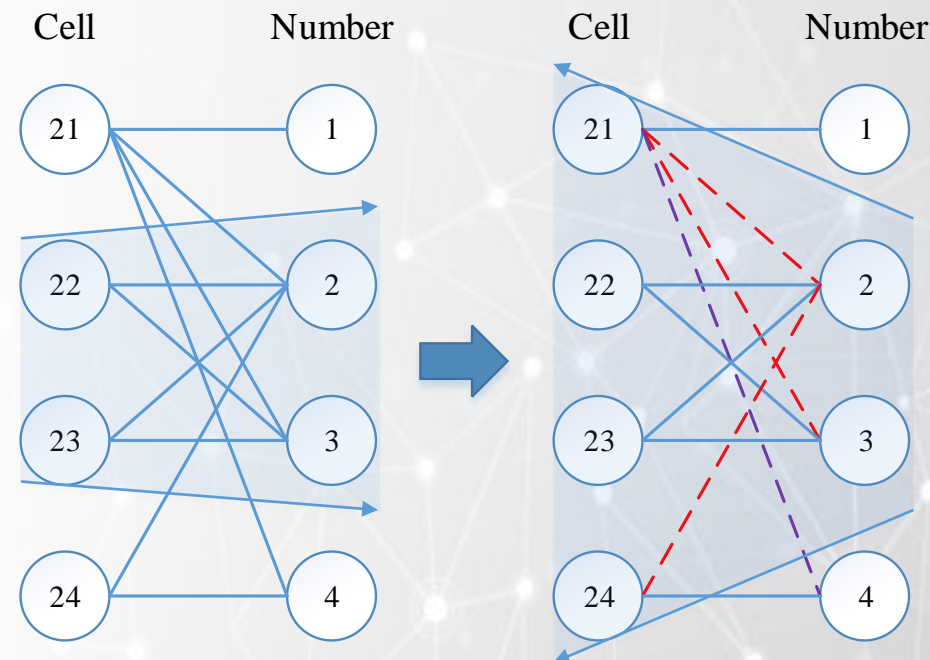
• 约简规则

- 考虑存在完美匹配的充要条件：霍尔定理 $\forall T \subseteq X, |T| \leq |N(T)|$
 - 二分图存在完美匹配 \Leftrightarrow 一侧节点的任意子集基数不大于其邻居节点并集的基数
- 约简规则 1 (单元格 \rightarrow 数字): 设 D 为某个行/列中一组单元格 C 的颜色域的并集。若 $|C|=|D|$ ，则同一行/列中除了 C 中的单元格之外，其他单元格均不能填 D 中的任何数字

{2,3}	4	1	{2,3}
{1,2,3,4}	{2,3}	{2,3}	{2,4}
{2,3}	{2,3}	4	1
{2,3,4}	1	{2,3}	{2,3}



{2,3}	4	1	{2,3}
{1,4}	{2,3}	{2,3}	{4}
{2,3}	{2,3}	4	1
{2,3,4}	1	{2,3}	{2,3}



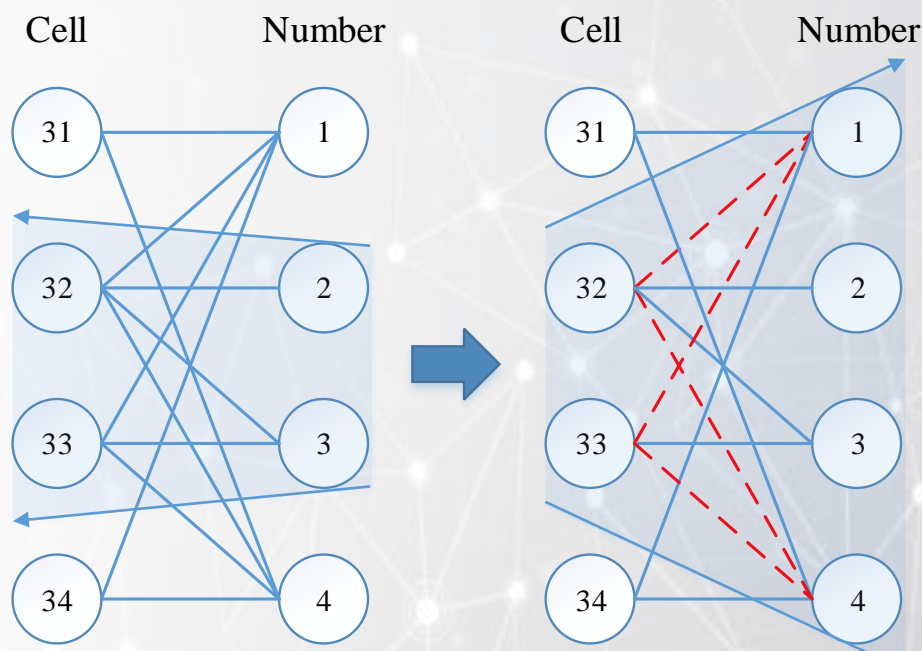
• 约简规则

- 考虑存在完美匹配的充要条件：霍尔定理 $\forall T \subseteq X, |T| \leq |N(T)|$
 - 二分图存在完美匹配 \Leftrightarrow 一侧节点的任意子集基数不大于其邻居节点并集的基数
- 约简规则 2 (数字 \rightarrow 单元格): 设 D 为一组数字, C 为可用 D 中某些数字填充的单元格集合。若 $|C|=|D|$, 则 C 中的单元格只能填 D 中的数字

{1,4}	{1,3,4}	{1,3,4}	2
3	{1,4}	2	{1,4}
{1,4}	{1,2,3,4}	{1,3,4}	{1,4}
2	{1,4}	{1,4}	3



{1,4}	{1,3,4}	{1,3,4}	2
3	{1,4}	2	{1,4}
{1,4}	{2,3}	{3}	{1,4}
2	{1,4}	{1,4}	3

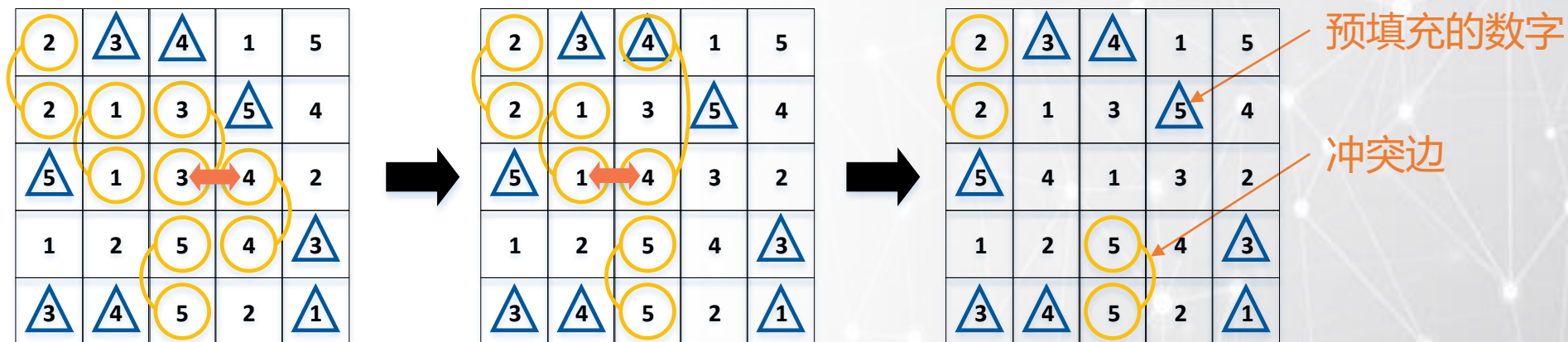


• 初始解生成

- 对每一行，随机为每个单元格分配不同的数字（发牌）
 - 确保行内无重复数字，但可能出现列内数字冲突

• 局部搜索

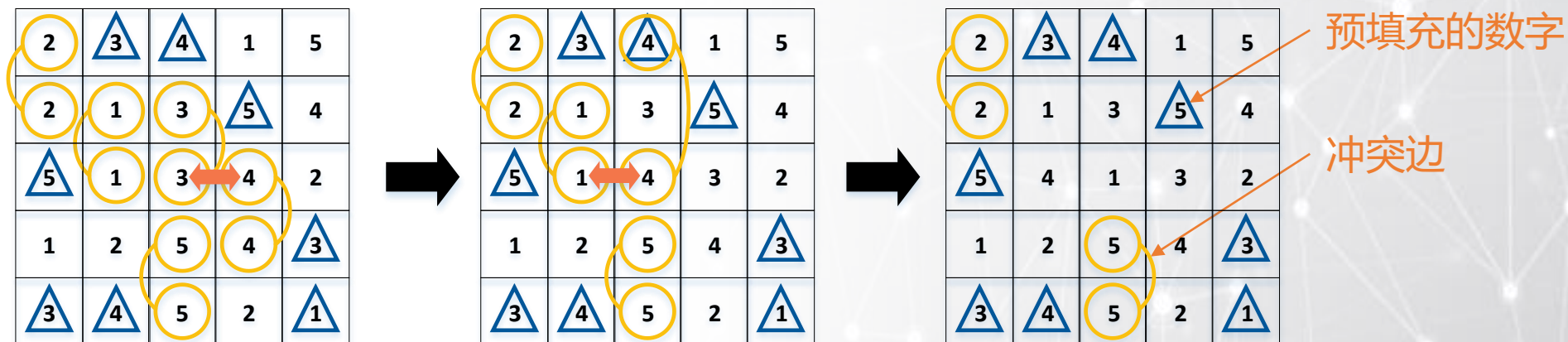
- 集中性：迭代执行消解最多冲突的邻域动作以改进当前解
 - 邻域动作是针对当前解的幅度有限的修改（例如 $x = 0 \rightarrow x = 1$ 、 $x \leftrightarrow y$ 等）
- 疏散性：使用禁忌搜索策略和重启策略跳出局部最优



• 局部搜索

• 朴素的邻域结构

- 更改单元格的数字：直接采用图着色问题的经典换色邻域（文献中的方案）
 - 未充分利用每个数字恰好出现 N 次的隐含条件
 - 约束松弛过多搜索容易迷失方向——到处都可以改进但都是拆东墙补西墙不本质
- 交换任意两个单元格的数字
 - 搜索空间过大，邻域评估复杂度高
 - 缺乏针对性，冲突数不变的动作极多

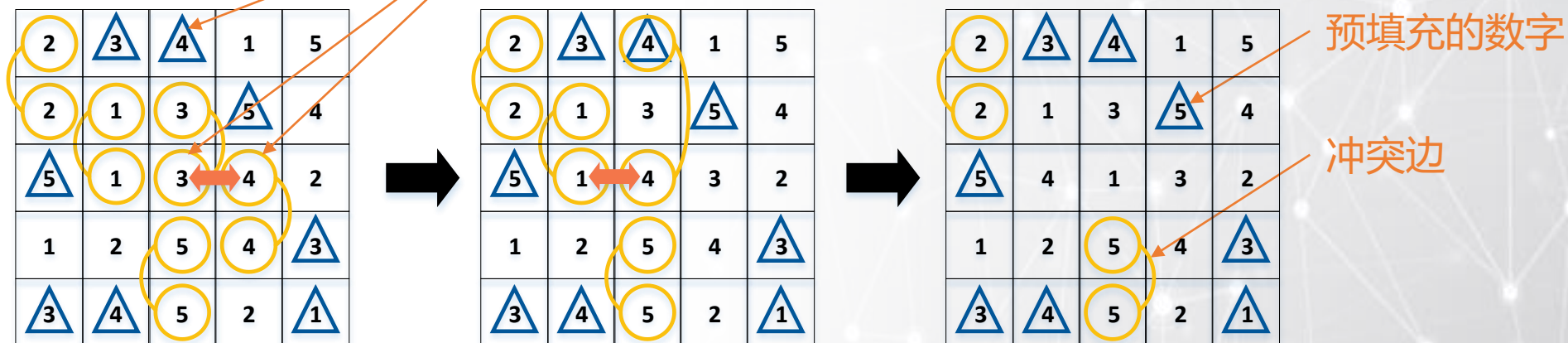


• 局部搜索

• 基于交换松弛的邻域

- 交换一个**冲突单元格**与**同一行**中的另一个**单元格**的数字
 - 即至少解决一个冲突（虽然可能引入新的冲突）
- 允许违反约简后的颜色域（提高解空间的连通性）
 - 但原本固定的数字不参与交换，避免无谓的约束违反

交换数字3和数字4将会导致与同一列预填充的数字4冲突，即（约简后）被交换的4不在该单元格的颜色域中，但该交换仍然被允许执行



• 局部搜索

• 用于邻域评估的分层目标函数

- 首要目标函数 (f_1) : 最小化冲突边数量 (减少显式冲突)
 - 在局部搜索后期, 交换动作通常会减少 $\{-2, -1, 0, 1, 2\}$ 个冲突
 - 大量交换动作第一目标函数值相同, 评估结果区分度低, 容易梯度消失

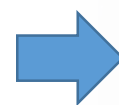
$$\Delta f_1(u, v) = |CE(C')| - |CE(C)|$$

- 次级目标函数 (f_2) : 最小化所填数字超出其颜色域的单元格数量 (减少隐式冲突)
 - 一定程度上缓解梯度消失问题

$$\Delta f_2(u, v) = |DV(C')| - |DV(C)|$$

2	3	4	1	5
4	1	3	5	2
5	1	3	2	4
1	5	2	4	3
3	4	5	2	1

Column Color	1	2	3	4	5
1	1	2	0	1	1
2	1	0	1	2	1
3	1	1	2	0	1
4	1	1	1	1	1
5	1	1	1	1	1



2	3	4	1	5
4	1	3	5	2
5	1	2	3	4
1	5	2	4	3
3	4	5	2	1

Column Color	1	2	3	4	5
1	1	2	0	1	1
2	1	0	1+1	2-1	1
3	1	1	2-1	0+1	1
4	1	1	1	1	1
5	1	1	1	1	1

• 局部搜索

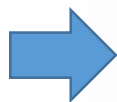
• 增量评估

- 主要目标：维护每列中每种颜色的数量
 - 若换入该列的数字数量 ≥ 1 ，则新增1个冲突 ($1 \rightarrow 2+$)
 - 若换出该列的数字数量 ≥ 2 ，则减少1个冲突 ($2 \rightarrow 1$)
 - 执行交换动作后：换入该列的数字数量加1，换出该列的数字数量减1
- 次要目标：维护每个单元格上每个数字的可用性
 - 若填入的数字不在颜色域内，则新增1个冲突
 - 若原数字不在颜色域内，则减少1个冲突

“两文字监督”

2	3	4	1	5
4	1	3	5	2
5	1	3	2	4
1	5	2	4	3
3	4	5	2	1

Column Color	1	2	3	4	5
1	1	2	0	1	1
2	1	0	1	2	1
3	1	1	2	0	1
4	1	1	1	1	1
5	1	1	1	1	1



2	3	4	1	5
4	1	3	5	2
5	1	2	3	4
1	5	2	4	3
3	4	5	2	1

Column Color	1	2	3	4	5
1	1	2	0	1	1
2	1	0	1+1	2-1	1
3	1	1	2-1	0+1	1
4	1	1	1	1	1
5	1	1	1	1	1

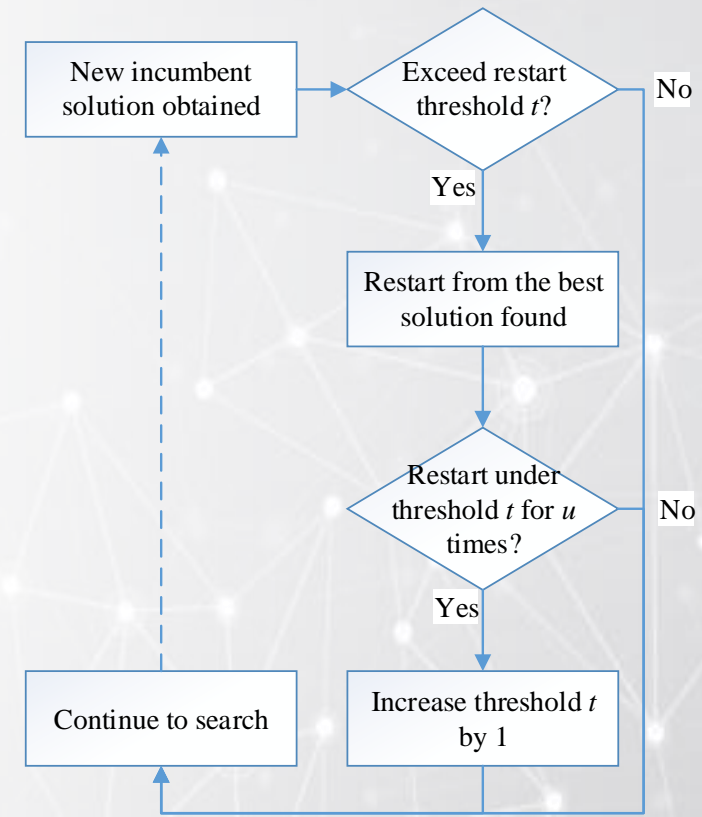
• 局部搜索

• 禁忌策略

- 在 $f_1 + \mu$ 次迭代中禁止将某数字填入特定单元格, 其中 $\mu \sim U(0,9)$
- 如果至少有一个 <单元格, 数字> 对处于禁忌状态, 则该交换移动处于禁忌状态
- 特赦准则: 如果邻居解优于历史最优解, 则临时解除禁忌状态

• 自适应重启

- 搜索可能会陷入僵局难以逃离局部最优
- 扰动或交叉方法破坏性过大
- 连续若干轮迭代无法改进历史最优解后重启?
 - 该策略对参数非常敏感
- 当前解与历史最优解的差距超过阈值时重启
 - 阈值随重启次数的增加逐渐增加
 - 由于可行解最终必须是无冲突的 ($f_1 = 0$), 故阈值参数的合理取值范围往往远小于最大迭代次数



简介

Introduction

问题

Problem

算法

Algorithm

实验

Experiment

结论

Conclusion

实验

• 与参考算法的对比

• 基准算例

- 1,800 个随机算例
- 19 个 COLOR03 竞赛算例

• 参考算法

- 2 个精确求解器
- 4 种 ILS 变体算法
- MMCOL
- FastLSC

• 实验结果

- 与精确求解器相比，
成功率提高超过 40%
- 与启发式方法相比，
速度提升超过 8 倍

Instance	OR-tools		Gurobi		SRLS	
	SR(%)	t_{avg} (s)	SR(%)	t_{avg} (s)	SR(%)	t_{avg} (s)
QWH-50	82.83	120.92	50.83	409.18	99.94	3.08
QWH-60	36.33	243.04	17.00	14.50	100.00	1.30
QWH-70	16.83	8.75	16.67	1.68	100.00	1.70
COLOR03	52.63	15.32	36.84	108.12	92.81	51.14

Table 2: Results of Gurobi, OR-tools and SRLS on all instances with the time limit of 1000 seconds.

Instance	1-ILS*		2-ILS		3-ILS		Tr-ILS*		MMCOL		FastLSC		SRLS	
Type	#solv	t_{avg} (s)	#solv	t_{avg} (s)	#solv	t_{avg} (s)	#solv	t_{avg} (s)	#solv	t_{avg} (s)	#solv	t_{avg} (s)	#solv	t_{avg} (s)
QWH-50-30	2998	3.19	3000	0.49	2881	4.66	3000	3.44	3000	0.17	3000	0.56	3000	0.05
QWH-50-40	2974	2.29	3000	0.39	2739	4.59	2986	2.63	3000	0.13	3000	0.52	3000	0.04
QWH-50-50	2675	2.57	2987	0.66	1780	5.78	2805	3.24	3000	0.17	3000	0.53	3000	0.03
QWH-50-60	364	5.95	1609	3.72	43	7.68	430	6.42	2995	1.54	3000	1.19	3000	0.05
QWH-50-70	0	N/A	0	N/A	0	N/A	0	N/A	53	5.78	539	6.36	2189	3.63
QWH-50-80	3000	0.83	3000	< 0.01	3000	0.83	3000	0.82	3000	< 0.01	3000	< 0.01	3000	< 0.01
QWH-60-30	2893	7.33	3000	1.54	953	8.37	2569	7.75	3000	0.40	3000	1.02	3000	0.13
QWH-60-40	2929	5.11	2996	1.05	1010	7.54	2834	6.03	3000	0.32	3000	0.92	3000	0.10
QWH-60-50	2410	5.10	2954	1.32	395	7.66	2160	6.40	3000	0.39	3000	0.89	3000	0.07
QWH-60-60	100	7.19	1440	4.35	0	N/A	84	7.78	2801	3.34	2983	2.48	3000	0.11
QWH-60-70	0	N/A	0	N/A	0	N/A	0	N/A	7	4.25	288	7.28	2429	3.88
QWH-60-80	2972	1.84	2991	0.07	2971	1.93	2981	1.85	3000	< 0.01	3000	< 0.01	3000	< 0.01
QWH-70-30	0	N/A	2998	3.78	0	N/A	0	N/A	3000	0.83	3000	1.75	3000	0.29
QWH-70-40	821	9.20	2988	2.40	36	9.40	303	9.19	3000	0.65	3000	1.49	3000	0.21
QWH-70-50	1391	8.42	2862	2.50	0	N/A	418	8.78	3000	0.85	3000	1.49	3000	0.15
QWH-70-60	17	8.56	975	5.18	0	N/A	0	N/A	1961	4.64	2846	3.88	3000	0.19
QWH-70-70	0	N/A	0	N/A	0	N/A	0	N/A	1	5.02	134	7.59	2180	4.37
QWH-70-80	2819	4.37	2935	0.53	2724	6.22	2850	4.76	3000	0.05	3000	< 0.01	3000	< 0.01
avg. speed-up		65.13		22.09		106.24		75.53		8.35		11.12		1.00

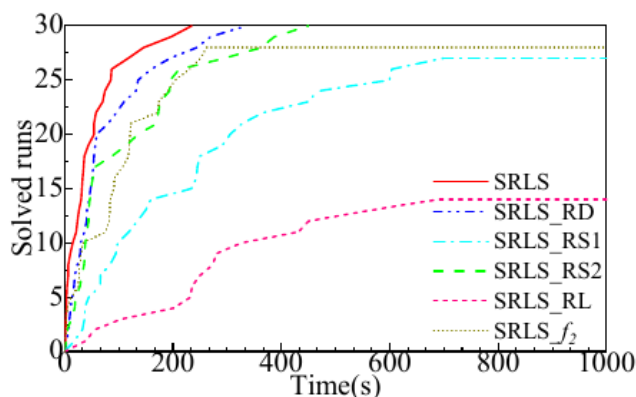
Table 3: Results of SRLS and other reference algorithms on 1800 random instances under the time limit of 10 seconds.

• 对 SRLS 组成部分的有效性分析

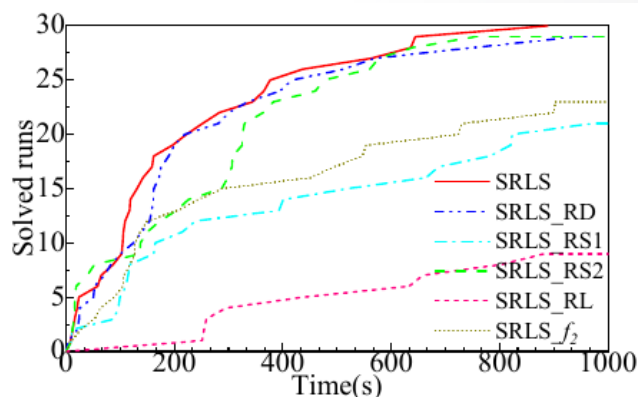
• 简化后的版本

- SRLS_RD: 仅使用文献中的约简规则
- SRLS_RS1: 禁用自适应重启机制
- SRLS_RS2: 每 105 次迭代重启一次局部搜索
- SRLS_RL: 禁用颜色域松弛技术
- SRLS_ f_2 : 禁用第二目标函数

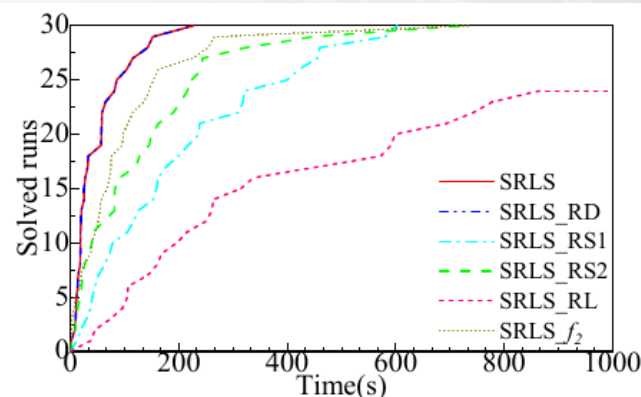
• 在困难算例上的实验结果表明，所提策略在 SRLS 中均发挥了重要作用



(a) QWH-50-70-100



(b) QWH-50-70-26



(c) QWH-60-70-50

简介

Introduction

问题

Problem

算法

Algorithm

实验

Experiment

结论

Conclusion

结论

• 结论

- 提出了两个新的约简规则缩减颜色域，相比之前的规则能更大程度地化简
- 引入了一种基于交换的新邻域，在同一行中交换两个单元格的数字，并始终保持每行无冲突
- 通过引入颜色域松弛技术，扩展了基于交换的邻域的搜索空间
- 采用了两级目标函数，以选择既具有即时影响又具有潜在影响的相邻解
- 提出了一种自适应重启机制，在当前解的质量相对较差时，返回历史最优解重新启动局部搜索

谢谢!