

[文章编号] 1003-4684(2022)01-0029-05

# 求解最小支配集的线性混合整型规划算法

程咏锋, 吴歆韵, 熊才权

(湖北工业大学计算机学院, 湖北 武汉 430068)

[摘 要] 提出了一个高效的求解最小支配集问题的线性混合整数规划算法(MILP)。该算法主要针对最小支配集问题的特点建立整数规划模型,并通过 Gurobi 求解器进行优化求解。采用当前国际文献公开的共 74 个算例作为算法测试实验集,与 FKW 算法、传统的 Grandoni 算法以及改进的 Grandoni 算法进行比较。实验结果表明,该算法的计算效率明显优于其它的精确算法,且在所有算例上都能得到精确解。

[关键词] 最小支配集; 线性整数规划算法; Gurobi 求解器; 精确算法

[中图分类号] TP30 [文献标识码] A

最小支配集问题为 NP 难问题,是一类重要的组合优化问题。对于最小支配集问题的求解不可能出现多项时间精确算法。最小支配集问题及其各种拓展问题在多文档摘要<sup>[1]</sup>, 社会网络的建模和研究<sup>[2]</sup>, 通信网络<sup>[3]</sup>等诸多领域具有广泛的应用。

目前,求解最小支配集问题精确算法均为非多项式时间算法,Fomin F V<sup>[4]</sup>等提出一种分支简化(FKW)算法,尽管最早突破  $O(2^n)$  界限,但 FKW 仍难令人满意,如果图中没有低于 2 度的节点,算法将回归平凡的枚举。Grandoni F<sup>[5]</sup>等人提出了求最小支配集的 Grandoni 算法,但该算法只能求最小支配集的大小,不能求具体的最小支配集;Lu gang<sup>[6]</sup>等对该算法提出了修改,应用于求解具体的最小支配集;Header<sup>[7]</sup>等提出一种混合遗传算法,与局部搜索算法相结合,设计出特定的适应度函数求解最小支配集,该算法在文献[8]中得到改进。David<sup>[9]</sup>等针对大型图的最小支配集问题,提出一种新的基于顺序的随机局部搜索算法。Zhu X<sup>[10]</sup>提出求解最小支配集的多项式时间近似方案,但这些算法不能在一般图上得到应用。Yuan Fuyu<sup>[11]</sup>通过分析大规模实例,结合支配集问题的特点提出一个快速高效地求解最小支配集问题的算法(Fast MDS),其他最小连通支配集算法参见文献[12-14]。

本文根据最小支配集问题的性质,构造出一种求解该问题的数学模型,采用 java 语言实现邻接支配矩阵的存储和根据数学模型利用 Gurobi 优化器

进行线性求解,并与文献<sup>[6]</sup>提及到的 FKW 算法,Grandoni 算法和改进的 Grandoni 算法在当前国际文献公开的共 74 个算例上进行比较,该算法的计算效率明显优于其它的精确算法,并且在所有算例上都能得到较好的精确解。

## 1 最小支配集问题的描述

### 1.1 问题定义

对于无向不连通图  $G=(V,E)$ ,图  $G$  的顶点集合为  $V=V(G)$ ,图  $G$  的边集合为  $E=E(G)$ 。若顶点集合  $V$  当中存在一个子集  $D \subseteq V$ ,对于集合  $V$  中的任意一个顶点  $V_p$ ,要么  $V_p \in D$ ,要么  $V_p$  与集合  $D$  当中某个顶点相邻,则将集合  $D$  称为图  $G=(V,E)$  的一个支配集。当集合  $D$  当中任意一个真子集都为非支配集时,则称集合  $D$  为图  $G$  的一个极小支配集。

设集合  $D$  是图  $G=(V,E)$  的一个支配集,当图  $G$  当中不存在任何一个支配集  $D_1$ ,使得  $|D_1| < |D|$ ,此时的集合  $D$  称为图  $G=(V,E)$  的一个最小支配集,图  $G$  当中所有最小支配集的大小称为图  $G=(V,E)$  的支配数,表示为  $\gamma(G)$ 。

以下给出最小支配集问题的相关定义:

定义 1 支配集(Dominating Set):集合  $D \subseteq V(G)$ ,若任意的  $V_p \in V(G) - D$ ,存在一个顶点  $V_q \in D$ ,使得顶点  $V_p$  与顶点  $V_q$  之间有边相连,满足这样条件的集合  $D$  就构成了图  $G$  的一个支配集,简记

[收稿日期] 2021-07-23

[基金项目] 国家自然科学基金(61902116);湖北工业大学博士启动基金(BSQD2019022)

[第一作者] 程咏锋(1995-),男,湖北黄冈人,湖北工业大学硕士研究生,研究方向为计算机科学与技术

[通信作者] 吴歆韵(1987-),男,江苏丹阳人,工学博士,湖北工业大学副教授,研究方向为组合优化算法

为  $DS$ 。

定义 2 极小支配集 (Minimal DS): 设集合  $D$  为图  $G=(V,E)$  的一个支配集, 当集合  $D$  中的任意一个真子集都无法构成图  $G=(V,E)$  的一个支配集, 则称集合  $D$  为图  $G=(V,E)$  的一个极小支配集。

定义 3 最小支配集 (Minimum DS): 在图  $G=(V,E)$  的所有支配集当中, 顶点个数最少的那个支配集称为图  $G=(V,E)$  的最小支配集。简称 MDS。

由相关支配集的定义可知极小支配集和最小支配集的相关性质:

1) 对于一个任意的非空图  $G=(V,E)$ , 可以找到若干个极小支配集, 这些极小支配集的阶 (顶点数) 不一定相同。

2) 对于一个任意的非空图  $G=(V,E)$ , 可以找到若干个最小支配集, 这些最小支配集的阶 (顶点数) 一定相同。

3) 如果一个集合  $X$ , 满足最小支配集的条件, 那么集合  $X$  一定满足极小支配集的条件; 反之, 若集合  $X$  满足极小支配集的条件, 那么集合  $X$  未必满足最小支配集的条件。

1.2 整数规划模型

最小支配集问题指的是对于给定的图  $G=(V,E)$ , 在集合  $V$  中找到一个支配集  $D$ , 使得支配集  $D$  中所含节点的个数达到最小。通过引入  $n$  维 0-1 向量  $\mathbf{X}=(x_1,x_2,x_3,\cdots,x_n)$ ,  $x_p=1$  表示顶点  $p \in D$ ,  $x_p=0$  表示顶点  $p \notin D$ , 则最小支配集的整数规划模型可以描述为式 (1)-(4):

$$\min f(\mathbf{X}) = \sum_{p=1}^n x_p \tag{1}$$

s.t.

$$\sum_{p=1}^n x_p * L_{p,q} \geq 1 (1 \leq p \leq n, 1 \leq q \leq n) \tag{2}$$

$$L_{p,q} = \begin{cases} 1 & \text{顶点 } V_p \text{ 与顶点 } V_q \text{ 相关联} \\ 0 & \text{顶点 } V_p \text{ 与顶点 } V_q \text{ 不相关联} \end{cases} \tag{3}$$

$$x_p = \begin{cases} 1 & \text{顶点 } V_p \text{ 作为支配集} \\ 0 & \text{顶点 } V_q \text{ 不作为支配集} \end{cases} \tag{4}$$

四个公式分别表示为: 1)  $f(\mathbf{X})$  作为目标函数表示以最少的顶点集合作为支配点, 2) 为约束条件表示任意一个顶点  $V_p$  要么是支配点, 要么被支配集和当中的点所支配,  $n$  表示图顶点的个数。3)  $L_{p,q}$  为常量, 表示第  $p$  节点是否可以与第  $q$  节点互相支配, 是则为 1, 否则为 0。4) 变量  $x_p$  表示是否选取第  $p$  个节点作为最小支配集, 是则为 1, 否则为 0。

2 图的最小支配集的多项式方程组模型与 Gurobi 求解器求解

Gurobi 是由美国 Gurobi 公司开发的新一代大

规模优化器, 综合能力强, 性价比较优。它支持 Windows, Linux, MacOSX 等多种平台, 采用了最新的优化技术, 充分利用多核处理器优势, 支持并行计算; 问题尺度只受限制于计算机内存容量, 不对变量数量和约束数量有限制; 提供了方便轻巧的 Java 接口; 能够快速高效地解决大规模线性问题、二次型目标问题、混合整数线性问题和二次型问题等数学问题。

设图  $G=(V,E)$  是无向图, 则图  $G$  的邻接矩阵定义为  $\mathbf{A}=[v_{pq}]_{n \times n}$ , 其中

$$v_{pq} = \begin{cases} 0 & p \neq q \\ 0 & p \neq q, e_{pq} \notin E \\ 1 & p = q, e_{pq} \in E \end{cases} \tag{1}$$

通常将  $\mathbf{A}$  的第  $i$  行记为  $A_i, 1 \leq i \leq n$ 。此外在此基础上定义邻接支配矩阵

$$\mathbf{A}^*=[v_{pq}]_{n \times n}$$

其中

$$v_{pq} = \begin{cases} 1 & p \neq q \\ 0 & p \neq q, e_{pq} \notin E \\ 1 & p = q, e_{pq} \in E \end{cases} \tag{2}$$

通常将  $\mathbf{A}^*$  的第  $i$  行记为  $A_i^*, 1 \leq i \leq n$ 。

对于给定具有  $n$  个顶点的无向图  $G=(V,E)$ , 其邻接支配矩阵为  $\mathbf{A}^*=[v_{pq}]_{n \times n}$ 。由最小支配集的定义可知, 求解一个多元多项式方程组的 0-1 解与求解图  $G$  的最小支配集这两个问题是等价的, 即  $n$  维向量  $\mathbf{X}$  中所含 1 的个数达到最小。

若  $D$  为  $G$  的一个支配集, 则由支配集的定义可知  $V-D$  中任意一个节点都与  $D$  里至少一个节点是相邻的。由于在  $V$  中有  $r_p=A_i^* \times (1,1,\cdots,1)expT$  个顶点与  $V_p$  相邻, 则可知包括  $V_p$  在内的这  $r_p+1$  个顶点至少有一个顶点在支配集  $D$  中, 由  $n$  维向量  $\mathbf{X}$  的定义就可得到  $n$  元多项式约束方程组:

$$(C_k) \begin{cases} A_1^* \times \mathbf{X} \geq 1 \\ A_2^* \times \mathbf{X} \geq 1 \\ \vdots \\ A_n^* \times \mathbf{X} \geq 1 \end{cases} \tag{3}$$

设  $n$  维向量  $\mathbf{X}_k=(x_1,x_2,x_3,\cdots,x_n)expT$ , 若该  $n$  维向量满足  $(C_k)$  的所有不等式, 故向量  $\mathbf{X}_k$  是方程组  $(C_k)$  的一个解。则其中取值为 1 的变量所对应的顶点构成的集合为支配点集合, 取值为 0 的变量所对应的顶点构成的集合为被支配点集合。

通过 Java 语言实现无向图的支配矩阵存储, 设置最优化模型的系数列表, 调用 Gurobi 优化器进行优化求解。首先使用 grbModel.addVar 函数构建目标函数系数列表; 其次使用 grbModel.addConst 构建约束条件的系数列表; 最后使用 grbModel.setObjective 函数优化求解。利用公共线性规划求解器对其进行求解, 使得目标函数值不断减小, 最终可

求得一个最小的  $k$  值。

### 3 实验结果与分析

本节主要是对求解最小支配集的 MILP 算法的性能进行实验分析。并与文献<sup>[6]</sup>提及到的 FKW 算法和 Grandoni 算法以及改进的 Grandoni 算法在当前国际文献公开的共 74 组算例上进行比较。第一组算例为文献中作者<sup>[15]</sup>随机生成的大规模算例。第二组算例,包含广泛的真实图,研究于许多领域,特别是在图染色问题上<sup>[16]</sup>。

实验环境:本次实验所涉及的算法均以 Java 语言进行编程实现,测试运行的 PC 机运行环境为 Core i5 3.4 GHz CPU 和 8.0 GB RAM,操作系统为 Windows 10,JVM 为 Oracle JRE 1.8。为了验证算法的性能,采用两组算例进行实验验证。

#### 3.1 随机算例的测试

第一组算例为随机生成的 24 组小规模算例,算例生成方法为:在一个  $N \times N$  的固定区域中随机选取一定数量的节点作为算例的图节点;当区域内的两节点  $p, q$  之间的距离小于某个设定的常数  $k$  值时,将  $\{p, q\}$  作为算例图中的一条边,即两节点  $p, q$  之间相互连通。该组随机算例节点数从 20—120,每组算例运行 10 次,求出最优解的时间并取平均值,结果见表 1。表 1 前两列分别表示算例的名称和最小支配集个数,其余列表示每个算法运行 10 次找到最小支配集个数时所需要的平均时间。

表 1 随机算例对比						s
Instance	$\gamma(G)$	$T_{MILP}$	$T_{GR}$	$T_{IGR}$	$T_{FKW}$	
V20_d4	16	0.001	0.277	0.085	0.020	
V20_d18	11	0.001	0.163	0.060	0.038	
V20_d43	5	0.03	0.071	0.064	0.080	
V20_d69	3	0.001	0.048	0.043	0.021	
V40_d17	25	0.001	48.273	13.133	0.039	
V40_d128	7	0.010	54.685	36.966	4.566	
V40_d179	6	0.020	10.468	10.361	2.005	
V40_d308	3	0.010	0.955	0.975	0.051	
V60_d744	4	0.050	23.064	23.012	0.239	
V60_d931	4	0.020	3.106	3.084	0.057	
V60_d1004	3	0.030	1.484	1.532	0.045	
V60_d637	4	0.030	101.531	99.931	0.535	
V80_d1240	4	0.080	120.312	118.413	0.696	
V80_d1604	3	0.030	53.608	57.908	0.208	
V80_d1848	4	0.090	10.585	10.684	0.063	
V80_d2049	3	0.130	3.043	3.033	0.065	
V100_d2116	4	0.300	120.361	118.242	2.642	
V100_d3176	3	0.220	19.499	19.065	0.110	
V100_d3604	2	0.030	3.076	3.065	0.04	

续表 1 随机算例对比					
Instance	$\gamma(G)$	$T_{MILP}$	$T_{GR}$	$T_{IGR}$	$T_{FKW}$
V100_d2417	4	0.320	140.392	138.473	1.314
V120_d4612	3	0.380	69.086	75.354	2.908
V120_d4288	3	0.260	247.736	248.589	0.167
V120_d4373	3	0.270	178.73	187.019	0.138
V120_d6306	3	0.360	46.562	48.800	0.192

图 1 为表 1 的信息汇总。通过图 1,可以明显看到,求出 24 组小规模算例的最小支配集,FKW 算法用了 16.239 s,Grandoni 算法用了 1257.115 s,改进的 Grandoni 算法用了 1217.891 s,而提出的 MILP 算法给出了 2.674 s 的最好结果,根据在随机生成的 24 组小规模算例上的结果,MILP 算法具有更好的性能。

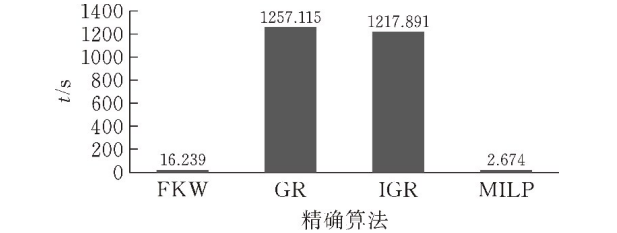


图 1 精确算法的总耗时

#### 3.2 图着色算例测试

由于第一组图的最小支配集较小,实验结果具有局限性。因此对第二组图进行测试,这个测试涉及到宽面图,该 50 组算例顶点规模从 11 到 864 个顶点不等,这些算例已经在许多研究中使用,特别是在图着色的问题上。这次测试包含 50 组算例,每组算例运行 10 次,测试结果见表 2。

在图着色算例上的测试结果见表 2,前四列分别表示算例的名称,顶点个数,边数,最小支配集个数,后四列分别表示 FKW 算法,GR 算法,IGR 算法,以及 MILP 算法求出最优解所需花费的时间。“—”表示未能给出最优解,“>3600”表示程序运行时间超过一小时,表 3 是表 2 的汇总,显示了各算法在 1 小时内成功求出最优解算例的个数。

从表 2 的结果可以看出,对于复杂算例的计算,IGR 算法的性能要优于 GR 算法,FKW 算法在计算时会造成程序崩溃。在除 Le450-5c 外的所有算例上,MILP 算法对于其余 49 个算例均能在一小时内找到最优解。通过表 3 的汇总可以看出,对比其他三个算法,MILP 算法在一小时的计算时间内成功率达到 98%,在所有算例上 MILP 算法的求解速度均快于其他算法,并且对于比较复杂的算例,MILP 算法求解的速度更快,这些数据表明,MILP 算法是求解最小支配集问题的一种高效的精确算法。

表 2 图着色算例对比

Instance	$V$	$E$	$\gamma(G)$	$T_{MILP}$	$T_{GR}$	$T_{IGR}$	$T_{FKW}$
Ann	138	986	12	0.001	>3600	0.164	>3600
David	87	812	2	0.001	>3600	0.07	>3600
Queen5-5	25	320	3	0.01	0.161	0.111	0.072
Queen6-6	36	580	3	0.01	0.477	0.416	0.068
Queen7-7	49	952	4	0.06	6.213	8.163	0.236
Queen8-8	64	1456	5	0.09	206.807	230.367	3.921
Queen8-12	96	2736	6	0.40	>3600	>3600	—
Queen9-9	81	2112	5	0.12	>3600	>3600	50.369
Queen10-10	100	2940	5	0.31	>3600	>3600	439.264
Queen11-11	121	3960	5	0.72	>3600	>3600	1127.183
Queen12-12	144	5192	6	2.53	>3600	>3600	—
Queen13-13	169	6656	7	2.16	>3600	>3600	—
Queen14-14	196	8372	8	13.0	>3600	>3600	>3600
Queen15-15	25	10360	9	246.13	>3600	>3600	—
Queen16-16	256	12640	9	185.46	>3600	>3600	—
Myciel3	11	20	3	0.001	0.067	0.096	0.104
Myciel4	23	71	4	0.01	0.191	0.219	0.115
Myciel5	47	236	5	0.01	375.154	388.484	1.041
Myciel6	95	755	6	0.09	>3600	>3600	>3600
Myciel7	191	2360	7	0.41	>3600	>3600	—
Miles250	128	774	22	0.01	>3600	>3600	>3600
Miles500	128	2340	9	0.001	>3600	>3600	>3600
Miles750	128	4226	6	0.001	>3600	219.16	>3600
Miles1000	128	6432	4	0.01	1272.308	1.539	8.648
Miles1500	128	10396	2	0.001	2.478	0.218	0.135
Mulsol.i.1	197	3925	2	0.001	>3600	0.158	0.124
Mulsol.i.2	188	3885	2	0.001	>3600	173.636	0.107
Mulsol.i.3	184	3916	2	0.001	>3600	163.917	0.056
Mulsol.i.4	185	3946	2	0.001	>3600	169.925	0.059
Mulsol.i.5	186	3973	2	0.001	>3600	174.448	0.054
Huck	74	602	9	0.001	>3600	0.124	1.364
Games120	120	1276	13	0.33	>3600	>3600	>3600
Zeroin.i.1	211	4100	2	0.001	>3600	0.131	0.088
Zeroin.i.2	211	3541	2	0.001	>3600	7.32	0.093
Zeroin.i.3	206	3540	2	0.001	>3600	8.742	0.082
Schooll	385	19095	15	0.16	>3600	>3600	—
School-nsh	352	14612	15	0.16	>3600	>3600	—
Fpsol2.i.1	496	11654	2	0.001	>3600	0.293	0.063
Fpsol2.i.2	451	8691	2	0.01	>3600	0.429	>3600
Fpsol2.i.3	425	8688	2	0.01	>3600	0.419	0.071
Le450-15a	450	8168	23	1.62	>3600	>3600	—
Le450-15b	450	8169	25	6.65	>3600	>3600	>3600
Le450-15c	450	16680	11	211.36	>3600	>3600	>3600
Le450-15d	450	16750	11	132.83	>3600	>3600	—
Le450-5a	450	5714	28	2500	>3600	>3600	—
Le450-5b	450	5734	28	2540	>3600	>3600	—
Le450-5c	450	9803	20	>3600	>3600	>3600	—
Inithx.i.1	864	18707	2	0.02	>3600	0.359	0.063
Inithx.i.2	645	13979	2	0.01	>3600	2.593	0.078
Inithx.i.3	621	13969	2	0.02	>3600	2.765	0.062

表 3 图着色算例结果汇总

	MILP	GR	IGR	FKW
Succeed	49	9	27	26
Fail	1	41	23	24
Success rate/%	98	18	54	52

4 结 论

本文提出了一种求解最小支配集问题的线性混合整数规划算法,在 MILP 算法中,根据最小支配集问题的特点,提出一种新的整数规划模型,将该问题



转化为线性规划问题,运用 Gurobi 求解器进行线性求解。采用当前国际文献公开的共 74 组算例作为算法测试实验集,在与文献中介绍的其他精确算法进行比较后发现,MILP 算法在中小规模的算例上能够以较快的时间求出最优解,对于较为复杂的算例,MILP 算法都能求出最优解,并且在计算出最优解的时间上具有明显优势。

[ 参 考 文 献 ]

[1] SHEN C, LI T. Multi-document summarization via the minimum dominating set[C]// International Conference on Computational Linguistics (ICCL). IEEE, 2010, 20-22.

[2] DINH T N, SHEN Y, NGUYEN D T. On the approximability of positive influence dominating set in social networks[J]. Journal of Combinatorial Optimization, 2014, 27(3): 487-503.

[3] CHENG X, MIN D, DU D H, et al. Virtual backbone construction in multihop ad hoc wireless[J] networks. Wireless Communications & Mobile Computing, 2010, 6(2): 183-190.

[4] FOMIN F V, KRATSCH D, WOEGINGER G J. Exact (exponential) algorithms for the dominating set problem [J]. Lecture Notes in Computer Science, 2004, 3353: 245-256.

[5] GRANDONI F. A note on the complexity of minimum dominating set [J]. Journal of Discrete Algorithms, 2006, 4(2): 209-214.

[6] 路纲,周明天,唐勇,等.任意图支配集精确算法回顾[J].计算机学报, 2010, 33(6): 1073-1087.

[7] HEDAR A-R, ISMAIL R. Hybrid genetic algorithm for minimum dominating set problem[C]// International conference on computational science and its applica-

tions; Springer, 2010, 457-467.

[8] POTLURI A, SINGH A. Two hybrid meta-heuristic approaches for minimum dominating set problem[C]// International Conference on Swarm, Evolutionary, and Memetic Computing; Springer, 2011, 97-104.

[9] CHALUPA D. An order-based algorithm for minimum dominating set with application in graph mining[J]. Information Sciences, 2018, 426: 101-116.

[10] ZHU X, WANG W, SHAN S, et al. A PTAS for the minimum weighted dominating set problem with smooth weights on unit disk graphs [J]. Journal of Combinatorial Optimization, 2012, 23(4): 443-450.

[11] 袁福宇. 若干支配集优化问题求解的方法研究[D]. 长春: 东北师范大学, 2019.

[12] SIMONETTI L, CUNHA A, LUCENA A. The minimum connected dominating set problem: formulation, valid inequalities and a branch-and-cut algorithm[C]// International Conference on Network Optimization, 2011, 34-37

[13] GENDRON B, LUCENA A, DA CUNHA A S, et al. Benders decomposition, branch-and-cut, and hybrid algorithms for the minimum connected dominating set problem[J]. INFORMS Journal on Computing, 2014, 26(4): 645-657.

[14] 王灵敏,周淘晴,吴歆韵,等. 求解最小连通支配集问题的变深度邻域搜索算法[J]. 中国科学:信息科学, 2016(4): 445-460.

[15] JOVANOVIĆ R, TUBA M. Ant colony optimization algorithm with pheromone correction strategy for the minimum connected dominating set problem[J]. Computer Science & Information Systems, 2013, 10(1): 133-149.

[16] RUTGERS. Coloring problems dimacs graph format [EB/OL].[2019-01-15]. Coloring Problems; DIMACS Graph Format (free.fr).

# A Hybrid Integer Linear Programming Algorithm for Solving the Minimum Dominating Set

CHENG Yongfeng, WU Xinyun, XIONG Caiquan

(School of Computer Science, Hubei Univ. of Tech., Wuhan 430068, China)

**Abstract:** An efficient linear mixed integer programming algorithm (MILP) for solving the minimum dominating set problem is proposed. The algorithm mainly establishes an integer programming model based on the characteristics of the minimum dominating set problem, and optimizes it through the Gurobi solver. A total of 74 calculation examples published in the current international literature are used as the algorithm test experiment set to compare with the FKW algorithm, the traditional Grandoni algorithm and the improved Grandoni algorithm. Experimental results show that the computational efficiency of this algorithm is significantly better than other accurate algorithms, and accurate solutions can be obtained on all calculation examples.

**Keywords:** minimum dominating set; linear integer programming algorithm; gurobi solver; exact algorithm