软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2018,29(12):3595−3603 [doi: 10.13328/j.cnki.jos.005378]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一类可分离 SAT 问题的 $O(1.890^n)$精确算法*

黄金贵, 王胜春

(湖南师范大学 信息科学与工程学院,湖南 长沙 410081)

通讯作者: 黄金贵, E-mail: hjg@hunnu.edu.cn

摘 要: 布尔可满足性问题(SAT)是指对于给定的布尔公式,是否存在一个可满足的真值指派.这是第 1 个被证明的 NP 完全问题,一般认为不存在多项式时间算法,除非 P=NP.学者们大都研究了子句长度不超过 k 的 SAT 问题(k-SAT),从全局搜索到局部搜索,给出了大量的相对有效算法,包括随机算法和确定算法.目前,最好算法的时间复杂度不超过 $O((2-2/k)^n)$,当 k=3 时,最好算法时间复杂度为 $O(1.308^n)$.而对于更一般的与子句长度 k 无关的 SAT 问题,很少有文献涉及.引入了一类可分离 SAT 问题,即 3-正则可分离可满足性问题(3-RSSAT),证明了 3-RSSAT 是 NP 完全问题,给出了一般 SAT 问题 3-正则可分离性的 $O(1.890^n)$判定算法.然后,利用矩阵相乘算法的研究成果,给出了 3-RSSAT 问题的 $O(1.890^n)$精确算法,该算法与子句长度无关.

关键词: 可满足性问题;NP 完全问题;正则可分离性;精确算法;算法复杂性

中图法分类号: TP301

中文引用格式: 黄金贵,王胜春.一类可分离 SAT 问题的 $O(1.890^n)$精确算法.软件学报,2018,29(12):3595−3603. http://www.jos.org.cn/1000-9825/5378.htm

英文引用格式: Huang JG, Wang SC. $O(1.890^n)$ exact algorithm for a class of separable SAT problems. Ruan Jian Xue Bao/ Journal of Software, 2018,29(12):3595−3603 (in Chinese). http://www.jos.org.cn/1000-9825/5378.htm

$O(1.890^n)$ Exact Algorithm for a Class of Separable SAT Problems

HUANG Jin-Gui, WANG Sheng-Chun

(College of Information Science and Engineering, Hunan Normal University, Changsha 410081, China)

Abstract: The Boolean satisfiability problem (SAT) refers to whether there is a truth assignment that satisfies a given Boolean formula, which is the first confirmed NP complete problem that generally does not exist a polynomial time algorithm unless P=NP. However many practical applications of such problems often take place and are in need of an effective algorithm to reduce their time complexity. At present, many scholars have studied the problem of SAT with clause length not exceeding k (k-SAT). From global search to local search, a large number of effective algorithms, including random algorithm and determination algorithm are developed, and the best result, including probabilistic algorithm and deterministic algorithm for solving k-SAT problems, is that the time complexity is less than $O((2-2/k)^n)$, and when k=3 the time complexity of the best algorithm is $O(1.308^n)$. However, there is little literature about SAT problems that are more general than clause length k. This paper discusses a class of separable satisfiability problems (SSAT), in particular, the problem of 3-regular separable satisfiability (3-RSSAT) where the formula can be separated into several subformulas according to certain rules. The paper proves that 3-RSSAT problem is NP complete problem because any SAT problem can be polynomially reduced to it. To determine 3-regular separability of the general SAT problem, an algorithm is given with time complexity is no more than $O(1.890^n)$. Then by using the result in the matrix multiplication algorithm optimal research field, an $O(1.890^n)$ exact algorithm is constructed for solving the 3-RSSAT problem, which is the WELL algorithm independent of clause length.

Key words: satisfiability problem; NP complete problem; regular separability; exact algorithm; algorithm complexity

* 基金项目: 国家自然科学基金(61271264, 11471110)
Foundation item: National Natural Science Foundation of China (61271264, 11471110)
收稿时间: 2017-03-17; 修改时间: 2017-07-07; 采用时间: 2017-08-26; jos 在线出版时间: 2018-06-07
CNKI 网络优先出版: 2018-06-07 14:53:33, http://kns.cnki.net/kcms/detail/11.2560.TP.20180607.1453.002.html

布尔可满足性(SATisfiability,简称 SAT)问题是指:对于给定的布尔公式 F,判定是否存在一个真值指派使得 F 满足.这是 Cook[1]在 1971 年证明了的第 1 个 NP 完全问题,后来也被 Levin[2]在 1973 年独立地证明.一般认为,不存在多项式时间算法除非 P=NP.这表明,要构造有效算法是非常困难的.简单直观的算法是:为每个变量分别指定 0 和 1 两个真值指派,验证每种可能指派下输入公式 F 是否满足.显然,这是指数时间算法,在最坏情况下,需要 $poly(n)\cdot 2^n$ 步,这里,n 是输入公式 F 包含的变量个数,$poly(n)$是 n 的多项式.在算法研究中,通常忽略多项式因子 $poly(n)$而只关注指数 2^n.研究的目标是降低指数的基,即:寻求 $c<2$,使得算法在最坏情况下不超过 $O(c^n)$.

已有的研究大都基于 k-SAT,即限制公式 F 为合取范式(conjunctive normal form,简称为 CNF)且每个子句最多包含 k 个文字,记为 k-CNF 公式.已经证明:2-SAT 问题是 P 问题,可在多项式时间内判定[3];而 k-SAT($k\geqslant 3$)是 NP 完全问题.第 1 个上界小于 2^n 的结果是 $c=2^{1-\varepsilon}$(ε是与 k 有关的正数),由 Monien 和 Speckenmeyer 于 1980 年获得[4],且对于 3-SAT 为 $poly(n)\cdot 1.619^n$.这是第 1 次实现 $c=2$ 的突破,此后,在 SAT 上的研究都是着眼于降低指数上界中的基数 c[5−11].

k-SAT 问题的求解主要采用全局搜索或局部搜索方式的启发式算法,分为随机算法和确定算法两类.随机算法是随机选取一个真值指派,然后根据不满足子句随机调整某个变量指派,如此反复,以保证较大概率获得可满足指派.随机算法最好的结果是 Schőning[7]于 1999 年得到的 $c=2-2/k$,是一个非常简单的随机局部搜索算法.且当 $k=3,4$ 时,分别有 3-SAT 为 $c=1.334$ 和 4-SAT 为 $c=1.5$.这个结果直到 2014 年被 Hertli[8]改进,在 PPSZ 算法[10]基础上得到 3-SAT 和 4-SAT 的最好结果分别为 $c=1.308$ 和 $c=1.469$.k-SAT 问题的确定算法的最好结果是 $c=2(k-1)/k+\varepsilon$(ε是足够小的正数),2010 年由 Moser 等人[10]改进 Dantsin 等人[9]2002 年的算法而得.而 3-SAT 确定算法最好的结果是 $c=1.331$[11],由 Makino 等人于 2011 年得到.

这些最好结果都是针对 k-SAT 问题,当 k 较大时,$c=2-2/k$ 将接近于 2.目前还没有与子句长度 k 无关的结果.本文试图针对一类特殊的 SAT 问题,给出常数 $c(c<2)$的确定算法.该算法首先分析 CNF 公式的可分离性,然后对于 3-正则可分离公式,应用已有的矩阵相乘算法[12,13]得到了预期结果,即与子句长度无关的常数 $c\leqslant 1.890$.这表明:当 $k\geqslant 19$ 时,本文算法优于 k-SAT 问题的最好算法.

本文第 1 节给出记号说明和可分离 SAT 问题的相关定义,并讨论 3-正则可分离 SAT 问题的复杂性.第 2 节讨论 3-正则可分离 SAT 问题的布尔矩阵表示,并提出 3-正则可分离 SAT 问题的判定矩阵概念.第 3 节讨论一般 SAT 问题的 3-正则可分离性的判定算法.第 4 节利用目前矩阵相乘结果,给出 3-正则可分离 SAT 问题的与子句长度无关的 $O(1.890^n)$精确算法.

1 问题的描述与定义

1.1 记号说明

本文中的布尔公式均为 CNF 公式.一个 CNF 公式 F 是有限个子句(clause)的集合.子句 C 是有限个两两互不相同的文字(literal)的集合.文字 l 是变量(variable)v 或其否变量 $\bar{v}$.如果变量 v 有 1 个文字出现在子句中,称该子句包含这个变量.用 var(C),var(F)分别表示子句 C、公式 F 所包含的变量的集合,记 V:=var(F), n=|var(F)|.

变量集 V 上的真值指派(truth assignment)是函数α:$V\to\{0,1\}^n$,为 V 中每个变量指派一个布尔值.文字 $l=v$(或 $l=\bar{v}$)对于指派α是满足的,如果$\alpha(v)=1$(或$\alpha(v)=0$).子句 C 对于指派α如果至少包含 1 个满足的文字,则该子句是满足的,即$\alpha(C)=1$.公式 F 是满足的当且仅当它的所有子句都是满足的.一个公式是可满足的当且仅当对它的所有变量存在一个满足的真值指派.

变量子集 $V'\subset V$ 上的真值指派α'称为 V 的部分指派(partial truth assignment),它是 V 上真值指派α在子集 V' 上的投影,即$\alpha'=\alpha|_{V'}$.两个不相交的变量子集 $V_1\subset V,V_2\subset V$ 上的真值指派分别为α_1,α_2,联合指派记为$\alpha_1\alpha_2=\alpha_1\cup\alpha_2$,它是变量集 $V_1\cup V_2$ 上的真值指派.

变量集 V 的划分是指 V 的若干个两两互不相交的子集 $V_1,V_2,\ldots,V_m(m\geqslant 2)$,满足 $V_1\cup V_2\cup\ldots\cup V_m=V$,记为 $[V_1,V_2,\ldots,V_m]$,也称为变量集 V 的 m-划分.V 上的真值指派α可以分解成 m 个部分指派$\alpha_1,\alpha_2,\ldots,\alpha_m$,分别对应每个

子集上的真值指派,即 $\alpha_1=\alpha|_{V_1},\alpha_2=\alpha|_{V_2},...,\alpha_m=\alpha|_{V_m}$; 反之,每个划分子集上的真值指派的联合为 V 上的真值指派,即 $\alpha_1\alpha_2\ldots\alpha_m=\alpha$.

1.2 CNF公式的可分离性

CNF 公式 F 的可分离性(separability)反映的是子句之间的关系,也是子句与变量之间的关系.这种分离性包括完全分离和不完全分离:完全分离的子公式之间是不相关的,即不含有相同变量.可完全分离的公式,一定可以分解为几个独立的变量规模更小的 SAT 问题,其时间复杂度将大大降低.我们一般假定所给定的 CNF 公式不可完全分离.对于变量集 V 的 m-划分$[V_1,V_2,\ldots,V_m]$,其中,$m\geqslant 2$.公式中的每个子句只与其中的部分子集相关,这时的分离性又称为 m-可分离,如果每个子集大小相等则称为正则可分离(regular separability).本文主要讨论 2-正则可分离性和 3-正则可分离性,即,在变量集的平均 2-划分或平均 3-划分的基础上分离子句.下面给出具体定义.

1.2.1 2-正则可分离定义

定义 1. CNF 公式 F 称为是 2-正则可分离的,如果存在变量集 V=var(F)的一个平均 2-划分$[V_1,V_2]$,使得每个子句 $C\in F$,要么 var(C)$\subseteq V_1$,要么 var(C)$\subseteq V_2$.

这里假定了 $n=|V|$是偶数,显然不会失去一般性.同时我们注意到,2-正则可分离是一种完全分离.例如公式 $F=\{\{x_1,\overline{x}_2,\overline{x}_3\},\{\overline{x}_1,x_2\},\{x_2,x_3\},\{x_4,x_5,\overline{x}_6\},\{\overline{x}_4,x_6\},\{\overline{x}_5,x_6\}\}$,变量集 $V=\{x_1,x_2,x_3,x_4,x_5,x_6\}$存在一个平均 2-划分$[\{x_1,x_2,x_3\},\{x_4,x_5,x_6\}]$,使得每个子句的变量都完全包含在其中一个划分子集中,所以 F 是 2-正则可分离的,且两个子公式 $F_1=\{\{x_1,\overline{x}_2,\overline{x}_3\},\{\overline{x}_1,x_2\},\{x_2,x_3\}\}$ 和 $F_2=\{\{x_4,x_5,\overline{x}_6\},\{\overline{x}_4,x_6\},\{\overline{x}_5,x_6\}\}$ 是完全独立的,因此公式 F 是可完全分离的.

1.2.2 3-正则可分离定义

定义 2. CNF 公式 F 称为是 3-正则可分离的,如果存在变量集 V=var(F)的平均 3-划分$[V_1,V_2,V_3]$,使得每个子句 $C\in F$,至少有一个划分子集不含有子句 C 的变量.

不失一般性,假定 $n=|V|$是 3 的倍数.显然,3-正则可分离是一种不完全分离.例如变量集 $V=\{x_1,x_2,x_3,x_4,x_5,x_6\}$上的 CNF 公式 $F=\{\{x_1,\overline{x}_2,\overline{x}_3\},\{\overline{x}_1,x_2,\overline{x}_4\},\{x_3,x_5\},\{x_4,x_5,\overline{x}_6\},\{\overline{x}_1,x_6\},\{x_2,\overline{x}_5,x_6\}\}$,取变量集 V 的平均 3-划分$[\{x_1,x_2\},\{x_3,x_4\},\{x_5,x_6\}]$,容易验证每个子句都与其中的一个子集不相关,因此,F 是 3-正则可分离的.其分离子公式分别为 $F_1=\{\{x_1,\overline{x}_2,\overline{x}_3\},\{\overline{x}_1,x_2,\overline{x}_4\}\}$,$F_2=\{\{x_3,x_5\},\{x_4,x_5,\overline{x}_6\}\}$,$F_3=\{\{\overline{x}_1,x_6\},\{x_2,\overline{x}_5,x_6\}\}$.变量集的划分不必是惟一的,如平均 3-划分$[\{x_1,x_4\},\{x_2,x_3\},\{x_5,x_6\}]$,也可将公式 3-正则分离.

容易验证,变量集 V 的任何划分都不能完全分离公式 F,即 F 是不可完全分离的,更不是 2-正则可分离的.如果在公式中再增加两个子句,即设 CNF 公式 $F'=F\cup\{\{\overline{x}_1,x_4,\overline{x}_6\},\{\overline{x}_3,x_4,\overline{x}_5\}\}$,则找不到变量集 V 的任何平均 3-划分,使得 F'可 3-正则分离,即说明 F'不是 3-正则可分离的.

从上述讨论我们看到:3-正则可分离虽然不是完全分离,但也不是所有的 CNF 公式都具备的属性,因此对 CNF 公式有严格的限制.从本文第 2 节将知道,3-正则可分离 CNF 公式可以很方便地表示成布尔矩阵的乘积,这样就能应用矩阵乘积算法的研究成果,更有效地解决这类 SAT 问题.

1.3 3-正则可分离SAT问题

3-正则可分离 SAT 问题(3-regular separable SAT,简称 3-RSSAT)是一类特殊的 SAT 问题,即,要判定 3-正则可分离公式 F 的可满足性.下面的定理描述了该问题的复杂性.

定理 1. 3-RSSAT 问题是 NP 完全问题.

证明:显然,3-RSSAT 是 NP 问题,所以我们只要证明一般 SAT 问题可多项式时间归约为 3-RSSAT 问题,本定理即能得证.

若 SAT 问题的实例公式 F 是 3-正则可分离的,则该问题就是 3-RSSAT 问题;否则,取变量集 V=var(F)的一个 3-划分$[V_1,V_2,V_3]$,使得$|V_1|=n/2,|V_2|=|V_3|=n/4$,其中,$n=|V|$(不妨假定 n 是 4 的倍数).令:

$$F_1=\{C\in F|\text{var}(C)\cap V_1\neq\varnothing\text{且 var}(C)\cap V_2\neq\varnothing\text{且 var}(C)\cap V_3\neq\varnothing\}\subseteq F.$$

其中,每个子句都同时含有 V_1,V_2,V_3 中的变量.

设 $V_1=\{v_1,v_2,\ldots,v_{n/2}\}$,任取 V_1 的一个 2-划分$[V_{1x},V_{1y}]=[\{v_1,\ldots,v_{n/4}\},\{v_{n/4+1},\ldots,v_{n/2}\}]$.与 V_{1x},V_{1y} 中的变量相对应,

增加新变量集 $X=\{x_1,\ldots,x_{n/4}\}$和 $Y=\{y_1,\ldots,y_{n/4}\}$,使得 $x_i=v_i,y_i=v_{i+n/4},i=1,2,\ldots,n/4$.

令 $V'=V\cup X\cup Y,V_1'=V_1,V_2'=V_2\cup X,V_3'=V_3\cup Y$,则易知 $[V_1',V_2',V_3']$ 是变量集 V'的平均 3-划分,且 $|V_1'|=|V_2'|=|V_3'|=n/2=|V'|/3$.下面构造变量集 V'上的 CNF 公式 F',其来源分为 3 个部分:

1) $F-F_1$ 中的所有子句全部加入到 F'中.对每个子句 $C\in F-F_1$,存在 $i\in\{1,2,3\}$,使得 $\mathrm{var}(C)\cap V_i=\varnothing$.又 $\mathrm{var}(C)\cap X=\varnothing$且 $\mathrm{var}(C)\cap Y=\varnothing$,所以 $\mathrm{var}(C)\cap V_i'=\varnothing$;

2) 若 $F_1\neq\varnothing$,对每个子句 $C\in F_1$,$\mathrm{var}(C)\cap V_1\neq\varnothing$,把每个变量 $v\in\mathrm{var}(C)\cap V_1$ 换成其对应的 X 中的变量 x 或 Y 中的变量 y,这样得到新子句 C',把 C'加入到 F'中.显然,$\mathrm{var}(C')\cap V_1=\varnothing$,即 $\mathrm{var}(C')\cap V_1'=\varnothing$;

3) 为保证 X 和 Y 中的变量与 V_1 中对应的变量相等,在 F'中新增如下子句集:

$$F_x=\{x_i\vee\overline{v}_i,\overline{x}_i\vee v_i\mid i=1,\ldots,n/4\}\text{和}F_y=\{y_i\vee\overline{v}_{i+n/4},\overline{y}_i\vee v_{i+n/4}\mid i=1,\ldots,n/4\}\ .$$

显然,对每个子句 $C\in F_x$, $\mathrm{var}(C)\cap V_3'=\varnothing$;对每个子句 $C\in F_y$, $\mathrm{var}(C)\cap V_2'=\varnothing$.

从上述构造过程可知:对每个子句 $C\in F'$,总是存在 V'的一个划分子集 $V_i'\in[V_1',V_2',V_3']$,使得 $\mathrm{var}(C)\cap V_i'=\varnothing$, 因此由定义 2,公式 F'是 3-正则可分离的.

下面证明公式 F 和公式 F'在可满足意义上等价.

事实上,

- 若公式 F'是可满足的,即存在指派 $\alpha':V'\to\{0,1\}^{|V'|}$,使得 $\alpha'(F')=1$.取 $\alpha=\alpha'|_V$,显然 $\alpha(F-F_1)=1$;而 $\alpha'(F_x)=1$, $\alpha'(F_y)=1$ 且 $\alpha'(F')=1$ 又保证了 $\alpha(F_1)=1$,所以公式 F 可满足;
- 反之,若公式 F 可满足,即存在指派 $\alpha:V\to\{0,1\}^{|V|}$,使得 $\alpha(F)=1$.取指派 $\alpha':V'\to\{0,1\}^{|V'|}$,使得 $\alpha'|_V=\alpha,\alpha'|_X=\alpha|_{V_{1x}},\alpha'|_Y=\alpha|_{V_{1y}}$,显然,α'是 F'的满足指派.

上述转化过程中,花费的时间主要体现在新增变量 $n/2$ 个,新增子句 $2\times n/4+2\times n/4=n$ 个,全部时间是多项式时间的.证毕. □

3-RSSAT 问题增加了对实例公式的 3-正则可分离性约束,但定理 1 表明,其复杂性并没有降低.引入该问题可以起到过渡的作用:一方面,3-RSSAT 问题与原问题难度接近,是否有可以接受的时间复杂度进行二者之间的转化;另一方面,3-RSSAT 问题便于布尔矩阵表示,可以应用矩阵乘积的很好结果,以期寻求 SAT 问题与子句长度无关的算法.

2 3-RSSAT 问题的布尔矩阵表示

2.1 矩阵乘积

对于正数 m,用粗体大写字母表示实数域 $R^{m\times m}$ 上的 $m\times m$ 矩阵,如 $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$.用相对应的带下标的小写字母表示矩阵元素,如 $A=(a_{ij})_{m\times m}$,其中,$a_{ij}\in R(i,j=1,\ldots,m)$表示矩阵 $\boldsymbol{A}$ 中的第 i 行第 j 列元素.布尔矩阵(或称为逻辑矩阵)指的是元素全为 0 或 1 的矩阵.

两个 $m\times m$ 实数矩阵 $\boldsymbol{A},\boldsymbol{B}$ 的乘积定义为

$$\boldsymbol{A}\times\boldsymbol{B}=\left(\sum_{j=1}^{m}a_{ij}b_{jk}\right)_{i,k=1,\ldots,m}\tag{1}$$

布尔矩阵也是实数矩阵,两个 $m\times m$ 布尔矩阵 $\boldsymbol{A},\boldsymbol{B}$ 的乘积可以按实数矩阵计算,即:

$$\boldsymbol{A}\times\boldsymbol{B}=\left(\bigvee_{j=1}^{m}(a_{ij}\wedge b_{jk})\right)_{i,k=1,\ldots,m}=\left(\sum_{j=1}^{m}a_{ij}b_{jk}>0\right)_{i,k=1,\ldots,m}\tag{2}$$

此外还定义两个 $m\times m$ 布尔矩阵 $\boldsymbol{A},\boldsymbol{B}$ 的与($\wedge$)运算为

$$\boldsymbol{A}\wedge\boldsymbol{B}=(a_{ij}\wedge b_{ij})_{i,j=1,\ldots,m}\tag{3}$$

对于两个 $m\times m$ 矩阵的乘法,传统算法需要花费时间 $O(m^3)$.直到 1969 年,Strassen[12]首先改变了这个现状,用一种巧妙而简单的方法将时间复杂度降到 3 次以下达到了 $O(m^{2.808})$.此后的研究就在 $O(m^{\omega})$上不断刷新 m 的指数 $\omega(\omega\in[2,3])$,使其更接近于 2.到 1989 年,已经由 Coppersmith 和 Winograd[13]改进到了 $\omega<2.376$.这一结果保

持了 20 多年,后来并无大的改进.矩阵乘积算法已有了很多应用,如图顶点最短路径算法的改进[14]、Max-2-SAT 问题的求解[15]等.

2.2 3-正则可分离公式的判定矩阵

对于给定的 3-正则可分离 CNF 公式 F,其变量集 V=var(F)必存在一个平均 3-划分[V_1,V_2,V_3],其中,$|V_1|=|V_2|=|V_3|=n/3$,不妨假定 $n=|V|$是 3 的倍数,使得公式 F 可 3-正则分离,3 个分离子公式记为

$$F_{12}=\{C\in F|\text{var}(C)\subseteq V_1\cup V_2\},F_{23}=\{C\in F|\text{var}(C)\subseteq V_2\cup V_3\},F_{13}=\{C\in F|\text{var}(C)\subseteq V_1\cup V_3\} \tag{4}$$

令 $m=2^{n/3}$,设 V_1,V_2,V_3 上的所有真值指派分别为

$$\left.\begin{aligned} V_1 &\to \{0,1\}^{n/3}:\{\alpha_0,\alpha_1,...,\alpha_{m-1}\}=\{0,1\}^{n/3}\\ V_2 &\to \{0,1\}^{n/3}:\{\beta_0,\beta_1,...,\beta_{m-1}\}=\{0,1\}^{n/3}\\ V_1 &\to \{0,1\}^{n/3}:\{\gamma_0,\gamma_1,...,\gamma_{m-1}\}=\{0,1\}^{n/3}\end{aligned}\right\} \tag{5}$$

对任意的$\alpha\in\{\alpha_0,\alpha_1,\ldots,\alpha_{m-1}\}$,$\beta\in\{\beta_0,\beta_1,\ldots,\beta_{m-1}\}$,$\gamma\in\{\gamma_0,\gamma_1,\ldots,\gamma_{m-1}\}$,通过联合分别得到变量子集 $V_1\cup V_2$,$V_2\cup V_3$,$V_1\cup V_3$ 上的指派为$\alpha\beta$,$\beta\gamma$,$\alpha\gamma$.变量集 V 上的指派为$\alpha\beta\gamma$.

定义 3. 令 $m=2^{n/3}$,$m\times m$ 布尔矩阵 $\boldsymbol{A}=(a_{ij})_{m\times m}$,$\boldsymbol{B}=(b_{jk})_{m\times m}$,$\boldsymbol{C}=(c_{ik})_{m\times m}$ 称为 3-正则可分离公式 F 的分离子公式(2-4)的布尔矩阵表示,其中,

$$a_{ij}=(\alpha_i\beta_j)(F_{12}),b_{jk}=(\beta_j\gamma_k)(F_{23}),c_{ik}=(\alpha_i\gamma_k)(F_{13}),i,j,k=0,1,\ldots,m-1 \tag{6}$$

从定义 3 可以看出,子公式的可满足性完全蕴含在其对应的布尔矩阵表示中.例如,子公式 F_{12} 的布尔矩阵表示 $A=(a_{ij})_{m\times m}$ 中,若其中某个元素 $a_{ij}=(\alpha_i\beta_j)(F_{12})=1$,则表明子公式 F_{12} 在指派$\alpha_i\beta_j$ 下满足;反之亦然.

定义 4. 设布尔矩阵 $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$ 是 3-正则可分离公式 F 的分离子公式(4)的布尔矩阵表示,则布尔矩阵 $\boldsymbol{P}=(\boldsymbol{A}\times\boldsymbol{B})\wedge\boldsymbol{C}$ 称为 3-正则可分离公式 F 的判定矩阵.

布尔矩阵的乘积运算、与运算参见公式(2)和公式(3).为了进一步分析判定矩阵,先给出如下引理.

引理 1. 3-正则可分离公式 F,对应变量集 V 的 3-正则分离划分是[V_1,V_2,V_3],ρ是 V 上的真值指派,取其部分指派 $\alpha=\rho|_{V_1},\beta=\rho|_{V_2},\gamma=\rho|_{V_3}$,则$\rho(F)=(\alpha\beta)(F_{12})\wedge(\beta\gamma)(F_{23})\wedge(\alpha\gamma)(F_{13})$.

证明:显然$\rho=\alpha\beta\gamma$.假设$\rho(F)=1$,若$(\alpha\beta)(F_{12})\wedge(\beta\gamma)(F_{23})\wedge(\alpha\gamma)(F_{13})=0$,不妨设$(\alpha\beta)(F_{12})=0$,则必有子句 $C\in F_{12}$,使得$(\alpha\beta)(C)=0$,根据公式(4)知 C 与指派γ无关,所以$\rho(C)=(\alpha\beta\gamma)(C)==(\alpha\beta)(C)=0$,与假设矛盾.

假设$\rho(F)=0$,则必有一个子句 $C\in F$,使得$\rho(C)=0$.由公式(4),不妨设 $C\in F_{12}$,所以$(\alpha\beta)(C)=0$,即$(\alpha\beta)(F_{12})=0$,因此$(\alpha\beta)(F_{12})\wedge(\beta\gamma)(F_{23})\wedge(\alpha\gamma)(F_{13})=0$. □

下面的定理说明了判定矩阵这个名称的含义,也是我们求解 3-RSSAT 问题的重要工具:

定理 2. 3-正则可分离公式 F 是可满足的,当且仅当它的判定矩阵 $\boldsymbol{P}$ 有非零元素.

证明:设 3-正则可分离公式 F 的判定矩阵 $\boldsymbol{P}=(p_{ik})_{(m+1)\times(m+1)}$,这里 $m=2^{n/3}$,根据定义 4,由公式(2)、公式(3)可得:

$$p_{ik}=\left(\bigvee_{j=0}^{m}(a_{ij}\wedge b_{jk})\right)\wedge c_{ik}=\bigvee_{j=0}^{m}(a_{ij}\wedge b_{jk}\wedge c_{ik}),i,k=0,1,...,m-1.$$

由公式(6)及引理 1 得:

$$p_{ik}=\bigvee_{j=0}^{m}(a_{ij}\wedge b_{jk}\wedge c_{ik})=\bigvee_{j=0}^{m}((\alpha_i\beta_j)(F_{12})\wedge(\beta_j\gamma_k)(F_{23})\wedge(\alpha_i\gamma_k)(F_{13}))=\bigvee_{j=0}^{m}((\alpha_i\beta_j\gamma_k)(F)),$$

其中,$i,k=0,1,\ldots,m-1$.

下面从两个方向证明定理.

(1) 若 3-正则可分离公式 F 是可满足的,即存在指派$\rho:V\to\{0,1\}^n$ 使得$\rho(F)=1$,则存在 $i,j,k\in\{0,1,\ldots,m-1\}$,使得 $\alpha_i=\rho|_{V_1},\beta_j=\rho|_{V_2},\gamma_k=\rho|_{V_3}$.则有 $p_{ik}=\bigvee_{j=0}^{m}((\alpha_i\beta_j\gamma_k)(F))=1$,即 F 的判定矩阵 $\boldsymbol{P}$ 中必有非零元素;

(2) 若 3-正则可分离公式 F 的判定矩阵 $\boldsymbol{P}$ 有非零元素,不妨设 $p_{ik}=1,i,k=0,1,\ldots,m-1$,则必存在 $j\in\{0,1,\ldots,m-1\}$,使得$(\alpha_i\beta_j\gamma_k)(F)=1$.即有变量集 V 上的指派$\rho=\alpha_i\beta_j\gamma_k$,使得$\rho(F)=1$,即 F 满足.

证毕. □

定理 2 表明:如果求出了 3-正则可分离 CNF 公式的判定矩阵,实际上就可以判定 3-RSSAT 问题.如果判定矩阵 $\boldsymbol{P}$ 有非零元素则满足,否则不可满足.

3 CNF 公式的正则可分离性判定

从上节讨论可知,定理 2 的前提条件是 3-正则可分离 CNF 公式,所以对于任意给定的 CNF 公式,我们首先必须判定它是否是 3-正则可分离的.对于 3-正则可分离 CNF 公式,为了能够得出其布尔矩阵表示及其判定矩阵,我们还必须求出其变量集的 3-正则分离划分.本节给出 3-正则可分离性的判定,并给出 3-正则可分离 CNF 公式的正则分离划分.

3.1 2-正则可分离性判定

首先讨论 2-正则可分离性问题.即:对于 CNF 公式 F,是否能给出变量集 V=var(F)的平均 2-划分[V_1,V_2],使得每个子句 $C\in F$,var(C)$\subseteq V_1$ 或 var(C)$\subseteq V_2$.

注意到变量集的平均 2-划分与分离公式 F 与变量的不同文字表示无关,所以把每个子句当成是变量集 V 的子集,即得子集簇:

$$F'=\{C'=\text{var}(C)|C\in F\}\subseteq 2^V \tag{7}$$

要分离公式 F 就转化为分离子集簇 F'.考虑到如果两个子集 $C'_1,C'_2\in F'$ 包含有相同变量,那么这两个子集必在 V 的同一个划分子集中,合并这样的子集,直到任意两个子集都不相交.从公式(7)最后得到的子集簇记为

$$F'=\{C'_1,C'_2,...,C'_p\} \tag{8}$$

其中,1<p=|F'|≤n≤|F|,不妨假定 n=|V|是偶数.

为了得到变量集的平均划分,只需考虑每个子集的大小,于是得到一个多重集:

$$S=\{s_1,s_2,\ldots,s_p\} \tag{9}$$

其中,$s_i=|C'_i|$ 是正整数,且 $1\leqslant s_i\leqslant n,i=1,\ldots,p$.显然,$s_1+s_2+\ldots+s_m=n$.这里的多重集是指集合中包含有值相同的元素,如 S={1,1,2,3,2}.

这说明 2-正则可分离性判定问题实质上是子集和问题.一般的子集和问题是 NP 完全问题[16],文献[17]采用动态规划给出了子集和问题的一个伪多项式算法,其时间复杂度为 $O(rN)$,其中,r 是集合中元素个数,N 是集合中所有元素的和.于是我们有如下引理:

引理 2[17]. 对于公式(9)定义的多重集合 S,给定正整数 t,存在算法在 $O(n^2)$时间内判定 S 是否有多重子集和等于 t.

其中的算法在判定的同时也能给出这个子集划分,即子集簇公式(8)的划分.这其实已经解决了 2-正则可分离性判定以及正则划分问题,即有下面的引理:

引理 3. 给定一个 CNF 公式 F,存在一种算法能够在多项式时间内判定 F 的 2-正则可分离性.如果 F 是 2-正则可分离的,则同时给出其变量集 V=var(F)的正则 2-划分[V_1,V_2].

3.2 3-正则可分离性的判定

给定 CNF 公式 F,V=var(F),不妨设 n=|V|是 3 的倍数.如果公式 F 是 3-正则可分离的,则根据定义 2 一定存在变量集 V 的平均 3-划分,记为 $[\tilde{V}_1,\tilde{V}_2,\tilde{V}_3]$,且 $|\tilde{V}_1|=|\tilde{V}_2|=|\tilde{V}_3|=n/3$,使得每个 $C\in F$,至少有一个划分子集不含有子句 C 的变量,即 $\text{var}(C)\cap\tilde{V}_1=\varnothing$或$\text{var}(C)\cap\tilde{V}_2=\varnothing$或$\text{var}(C)\cap\tilde{V}_3=\varnothing$.

如果能找到这样的平均 3-划分,则说明 F 是 3-正则可分离的,否则不是.遗憾的是,目前还没有有效的办法获得这样的平均 3-划分.简单直观的方法就是蛮力搜索,穷尽所有的平均 3-划分方案,对每种方案验证 F 的所有子句.显然,枚举算法的时间复杂度是 $O(3^n)$.本文给出的解决方案是采用部分枚举,使得时间复杂度大为降低.算法的基本思路是:首先枚举所有的 $n/3$ 个变量的组合形成划分子集 V_1,然后对剩下的 $2n/3$ 个变量应用第 3.1 节中的 2-正则分离算法得到平均 2-划分[V_2,V_3].算法过程描述见算法 1.

算法 1. 3-正则划分算法.

$[V_1,V_2,V_3]$=3-*regularPartition*(F,V,n)

输入:CNF 公式 F,V=var(F),n=|V|是 3 的倍数;

输出:若公式 F 是 3-正则可分离的,则返回正则分离划分$[V_1,V_2,V_3]$;否则返回 0.

1. V_1←枚举变量集 V 中 n/3 个变量的所有组合;
2. F'←{C'←var(C)∩(V–V_1)|C∈F 且 var(C)∩V_1≠∅且 var(C)∩(V–V_1)≠∅};
   //只考虑没有被 V_1 和 V–V_1 分离的子公式,且只关心变量而忽略文字形式.
3. $[V_1,V_2]$←2-*regularPartition*(F',V–V_1,2n/3);
   //调用 2-正则分离算法返回子公式 F'的变量集 2-正则划分;如果不存在,返回 0.
4. 如果成功,则返回划分$[V_1,V_2,V_3]$;
5. 否则,如果变量集 V 中还有 n/3 个变量的不同组合,则转步骤 1;
6. 否则,返回 0.

引理 4. 对于任意的 CNF 公式 F,算法 1 能在 $O(1.890^n)$时间内判定 F 的 3-正则可分离性,且当 F 是 3-正则可分离时,返回正确的正则分离划分.

证明:CNF 公式 F 的变量集 V=var(F),不妨设 n=|V|是 3 的倍数.算法 1 首先在步骤 1 划分出 V_1 且保证了|V_1|=n/3,且|V–V_1|=2n/3 是偶数.这时,未被 2-划分$[V_1,V-V_1]$分离的子句集为:

$$F'=\{C'\leftarrow \mathrm{var}(C)\cap(V-V_1)\mid C\in F \text{ 且 } \mathrm{var}(C)\cap V_1\neq\varnothing \text{且 } \mathrm{var}(C)\cap(V-V_1)\neq\varnothing\}.$$

实际上,F'是与子句集对应的变量子集簇,且 $var(F')\subseteq V-V_1$.若有变量 $v\in V-V_1$ 但 $v\notin var(F')$,则令 $F'=F'\cup\{\{v\}\}$,使得 $var(F')\subseteq V-V_1$.

(1) 若公式 F 是 3-正则可分离的,则必存在正则分离划分 $[\tilde{V}_1,\tilde{V}_2,\tilde{V}_3]$.如果算法 1 在步骤 1 划分出 V_1 后得到的公式 F'不是 2-正则可分离的,则根据引理 3,子算法 2-*regularPartition*(F',V–V_1,2n/3)返回 0,由步骤 5 返回步骤 1 重新选择下一个组合划分出新的 V_1,直至遍历完所有可能的组合.其中,必定有一次取到 $V_1=\tilde{V}_1$,这时得到的公式 F'一定能被 2-划分 $[\tilde{V}_2,\tilde{V}_3]$ 正则分离,即公式 F'是 2-正则可分离的.由引理 3 可知,子算法 2-*regularPartition*(F',V–V_1,2n/3)返回公式 F'的正则分离划分$[V_2,V_3]$.从而算法 1 步骤 4 返回公式 F 的正则分离划分$[V_1,V_2,V_3]$,算法终止;

(2) 若公式 F 不是 3-正则可分离的,则算法 1 在步骤 1 不论哪一次划分出的 V_1,所得到的公式 F'都不会是 2-正则可分离的.根据引理 3,子算法 2-*regularPartition*(F',V–V_1,2n/3)总是返回 0.当 V_1 穷尽所有可能的组合后,算法 1 执行到步骤 6,返回 0 并终止.

根据算法 1,除了步骤 1 以外,其他步骤都是多项式时间.而步骤 1 的时间复杂度是从 n 个变量中取出 n/3 个的组合数,由 Stirling 估算公式得:

$$\binom{n}{n/3}=\frac{n!}{(n/3)!(2n/3)!}\sim\frac{\sqrt{2\pi n}(n/e)^n}{\sqrt{2\pi n/3}(n/3e)^{n/3}\sqrt{4\pi n/3}(2n/3e)^{2n/3}}=\frac{3}{2\sqrt{\pi n}}\left(\frac{3}{2^{2/3}}\right)^n\leqslant O(poly(n)\cdot 1.890^n).$$

引理得证. □

4 3-RSSAT 问题的求解

对于 3-RSSAT 问题,它的每个实例都是 3-正则可分离 CNF 公式,求解算法的基本思路是:首先求出 3-正则划分,然后求得布尔矩阵表示及其判定矩阵,再应用矩阵乘积的优化算法计算出判定矩阵.下面描述主算法.

算法 2. 3- RSSAT 问题求解主算法.

Solving-3-*RSSAT*(F,V,n)

输入:3-正则可分离 CNF 公式 F,V=var(F),n=|V|是 3 的倍数;

输出:若公式 F 可满足则返回 1,否则返回 0.

1. $[V_1,V_2,V_3]$=3-*regularPartition*(F,V,n);

2. 根据定义 3 构建布尔矩阵 $\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$;

3. $\boldsymbol{Q}\leftarrow\Pi_{\omega}(\boldsymbol{A},\boldsymbol{B})$; //应用矩阵相乘算法$\Pi_{\omega}$计算矩阵 $\boldsymbol{A},\boldsymbol{B}$ 的普通乘积 $\boldsymbol{A}\times\boldsymbol{B}$,得到布尔矩阵 $\boldsymbol{Q}$.

4. $\boldsymbol{P}\leftarrow\boldsymbol{Q}\wedge\boldsymbol{C}$; //计算布尔矩阵 $\boldsymbol{Q},\boldsymbol{C}$ 的与运算(见公式(3))得到判定矩阵 $\boldsymbol{P}$.

5. 若矩阵 $\boldsymbol{P}$ 有非零元素则返回 1,否则返回 0.结束.

算法 2 中,Π_{ω}表示时间复杂度不超过 $O(m^{\omega})$的矩阵乘积算法,$m\times m$ 是矩阵的规模.布尔矩阵的普通乘积转化为实数矩阵计算(见公式(2)),然后将乘积矩阵转换为布尔矩阵 $\boldsymbol{Q}$.

定理 3. 存在算法能够在 $O(poly(n)\cdot 1.890^n)$时间内判定 3-RSSAT 问题.

证明:首先证明算法 2 能够判定 3-RSSAT 问题.设 F 是 3-RSSAT 问题的任意实例公式,则 F 是 3-正则可分离 CNF 公式.由引理 4,算法 2 步骤 1 返回 F 的正则分离划分$[V_1,V_2,V_3]$.于是,根据定义 3 和定义 4 能够得到 F 的判定矩阵 $\boldsymbol{P}$.由定理 2 知,算法 2 步骤 5 能够判定 F 的可满足性.

其次,不难看出算法 2 所花费的时间:步骤 1 花费时间由引理 4 得出不超过 $O(poly(n)\cdot 1.890^n)$;步骤 2 花费花费时间不超过 $O(poly(n)\cdot 2^{2n/3})$;步骤 3 花费时间由矩阵相乘算法得出为 $O(poly(n)\times 2^{\omega n/3})$,且$\omega<2.376$;步骤 4 和步骤 5 花费时间均不超过 $O(poly(n)\cdot 2^{2n/3})$.合计算法总时间不超过

$$O(poly(n)\cdot 1.890^n+poly(n)\cdot 2^{2n/3}+poly(n)\cdot 2^{\omega n/3})=O(poly(n)\cdot 1.890^n).$$

证毕. □

5 结 语

通过约束公式得到的一类特殊 SAT 问题——3-正则可分离 SAT 问题,它仍然是 NP 完全问题.这类问题的优势在于 3-正则可分离 CNF 公式可以有很好的布尔矩阵表示,并且能得出判定矩阵.根据判定矩阵,利用矩阵乘积算法得出 3-RSSAT 问题的常数基指数判定算法.为探索一般 SAT 问题的常数基指数上界算法迈开了一步.定理 3 表明,算法时间主要花在正则分离划分上,所以还有改进空间.算法 2 仅限于 3-正则可分离的 CNF 公式,离一般 SAT 问题差别很大.此外,CNF 公式的 3-正则可分离性判定问题的难度还是未知、判定算法优化等等这些问题都有待于后续进一步的研究.

References:

[1] Cook SA. The complexity of theorem-proving procedures. In: Proc. of the 3rd Annual ACM Symp. on Theory of Computing. 1971. 151−158. [doi: 10.1145/800157.805047]

[2] Levin L. Universal search problems. Problems of Information Transmission, 1973,9(3):265−266.

[3] Even S, Itai A, Shamir A. On the complexity of timetable and multi-commodity flow problems. In: Proc. of the Symp. on Foundations of Computer Science. IEEE Xplore, 1975. 184−193. [doi: 10.1109/SFCS.1975.21]

[4] Monien B, Speckenmeyer E. Solving satisfiabilty in less than 2n steps. Discrete Applied Mathematics, 1985,10(3):287−295.

[5] Paturi R, Pudlak P, Zane F. Satisfiability coding lemma. In: Proc. of the 38th IEEE Symp. on Foundations of Computing. 1997. 566−574.

[6] Paturi R, Pudlak P, Saks ME, Zane F. An improved exponential-time algorithm for *k*-SAT. In: Proc. of the 39th IEEE Symp. on Foundations of Computing. 1998. 628−637.

[7] Schöning U. A probabilistic algorithm for *k*-SAT and constraint satisfaction problems. In: Proc. of the 40th IEEE Symp. on Foundations of Computing. IEEE, 1999. 410−414.

[8] Hertli T. 3-SAT faster and simpler—Unique-SAT bounds for PPSZ hold in general. In: Proc. of the Foundations of Computer Science. IEEE, 2011. 277−284.

[9] Dantsin E, Goerdt A, Hirsch EA, *et al*. A deterministic $(2-2/(k+1))^n$ algorithm for *k*-SAT based on local search. Theoretical Computer Science, 2002,289:69−83.

[10] Moser RA, Scheder D. A full derandomization of Schöening's *k*-SAT algorithm. In: Proc. of the Annual ACM Symp. on Theory of Computing. 2010. 245−252.

[11] Makino K, Tamaki S, Yamamoto M. Derandomizing HSSW algorithm for 3-SAT. In: Proc. of the Int'l Conf. on Computing and Combinatorics. Springer-Verlag, 2011. 1−12.

[12] Strassen V. Gaussian elimination is not optimal. Numerische Mathematik, 1969,13:354−356.

[13] Coppersmith D, Winograd S. Matrix multiplication via arithmetic progressions. Journal of Symbolic Computation, 1990,9(3): 251−280.

[14] Alon N, Galil Z, Margalit O. On the exponent of the all-pairs shortest path problem. Journal of Computer and System Sciences, 1997,54:255−262.

[15] Williams R. Maximum two-satisfiability. In: Proc. of the Encyclopedia of Algorithms. 2008. 1−99.

[16] Karp RM. Reducibility among combinatorial problems. Journal of Symbolic Logic, 1972,1(4):85−103.

[17] Bellman R. Notes on the theory of dynamic programming iv-maximization over discrete sets. Naval Research Logistics Quarterly, 1956,3(1):67−70. [doi: 10.1002/nav.3800030107]

黄金贵(1964－),男,湖南临澧人,博士,教授,博士生导师,主要研究领域为计算机算法理论,NP 难问题优化.

王胜春(1977－),男,副教授,主要研究领域为机器学习,模式识别,图像处理,大数据处理.