广西科学院学报 2006,22(3):141～147
Journal of Guangxi Academy of Sciences Vol. 22, No. 3 August 2006

5个 Van der Waerden 数 $W(3,q)$ 的准确值*

Five Accurate Values for the Van der Waerden Number $W(3,q)$

苏文龙[1]**，罗海鹏**[2]**，黎贞崇**[2]**，何建东**[2]

SU Wen-long[1], LUO Hai-peng[2], LI Zhen-chong[2], HE Jian-dong[2]

(1. **广西大学梧州分校，广西梧州** 543002；2. **广西科学院，广西南宁** 530022)

(1. Guangxi University Wuzhou Branch, Wuzhou, Guangxi, 543002, China; 2. Guangxi Academy of Sciences, Nanning, Guangxi, 530022, China)

摘要：使用 3 个算法，给出 5 个 Van der Waerden 数 $W(3,q)$ 的准确值：$W(3,4)=18$，$W(3,5)=22$，$W(3,6)=32$，$W(3,7)=46$，$W(3,8)=58$.

关键词：Van der Waerden 数　n-部分拆　下界　上界　准确值

中图法分类号：O157.5；TP312　　**文献标识码**：A　　**文章编号**：1002-7378(2006)03-0141-07

Abstract: Three algorithms are used, and five accurate values for Van der Waerden number $W(3,q)$ are obtained as follows: $W(3,4)=18$, $W(3,5)=22$, $W(3,6)=32$, $W(3,7)=46$, $W(3,8)=58$.

Key words: Van der Waerden number, n-partition, lower bound, upper bound, accurate value

1　本文的主要结果

定义 1　Van der Waerden 数是具有下述性质的最小正整数 w：给定 $n \geqslant 2$ 个正整数 $k_1, k_2, \cdots, k_n$，在自然数集 $[w]=\{1,2,\cdots,w\}$ 的任意 n-部分拆 $\pi_n([w])=\{S_1,S_2,\cdots,S_n\}$ 中，必有一个子集 $S_i(1\leqslant i\leqslant n)$ 含 k_i 个元素构成的等差数列.

Van der Waerden 定理[1~3]证明了上述正整数 w 的存在性. 我们记这样的 Van der Waerden 数为 $W(k_1,k_2,\cdots,k_n)$. 特别地，当 $k_1=k_2=\cdots=k_n=k$ 时，一般的文献都记为 $W(k,n)$. 在这里，我们参照文献[4]中关于 Ramsey 数的记法，简记为 $W_n(k)$.

Van der Waerden 定理是 Ramsey 理论的重要组成部分，Van der Waerden 数的计算也是组合数学中非常困难的问题. 虽然迄今尚未见到学术界在 Ramsey 数的计算方面有取得重大突破的迹象，但探索小 Ramsey 数的准确值及其上、下界却是学术界的一个热门课题，迄今记录小 Ramsey 数研究成果的动态综述论文[4]已更新至 #10 版. 关于小 Van der Waerden 数的研究，理应像 Ramsey 数那样获得学术界的重视，但实际上其研究进展却极其缓慢. 迄今已知的非平凡的小 Van der Waerden 数的准确值只有如下 5 个，并且都是 $W_n(k)$ 类型的：$W_2(3)=9$，$W_2(4)=35$，$W_2(5)=178$，$W_3(3)=27$，$W_4(3)=76$.

对于 $W_2(k)$ 的界，文献[3]收录有如下公式：

$$W_2(k)\geqslant \frac{2^k}{2ek}-\frac{1}{k}.$$

$$W_2(k)\leqslant T(2W_2(k-1)).$$

其中 $T(n)$ 称为 2 的塔幂函数，定义为 $T(1)=2$，$T(n)=2^{T(n-1)}$.

R. L. Graham[2] 猜想 $W_2(k)\leqslant T(k)$. 即使这一猜想获得证实，由于 $T(n)$ 的递增速度远超过指数函数，因而 $W_2(k)$ 的这个上界和已知的准确值或下界比较起来相差极大. 例如 $T(3)=2^4=16>W_2(3)=9$，$T(4)=2^{16}=65536>W_2(4)=35$，而 $T(5)=2^{65536}$ 则远远大于 $W_2(5)=178$. Van der Waerden 定理中肯定其存在的数 $W_2(k)$ 的定量性质仍然是对数

收稿日期：2005-01-12

作者简介：苏文龙(1947-)，男，广西梧州人，研究员，主要从事组合数学与算法研究工作。

* 国家自然科学基金(60563008)、广西自然科学基金项目(桂科自0640037)、梧州市科研基金资助项目(梧科字[2005]第 35 号)资助。

学家的一大挑战[3].

至于 $k_1 \neq k_2$ 的小 Van der Waerden 数 $W(k_1,k_2)$ 的准确值或其上下界，迄今尚未见有文献报道. 本文的主要结果是

定理 1　$W(3,4)=18, W(3,5)=22, W(3,6)=32, W(3,7)=46, W(3,8)=58$.

这些结果是本文首次报道的.

2　计算 $W(k_1,k_2)$ 下界的算法

定义 2　给定整数 $p \geqslant 5$ 与自然数集 $[p]=\{1,2,\cdots,p\}$ 的 2- 部分拆 $\pi_2([p])=\{S_1,S_2\}$，其中 S_1 与 S_2 均非空集. 对于任一 $a_j \in S_i$，如果在 S_i 中最多有 $l \geqslant 3$ 个数 $a_j < a_{j+1} < \cdots < a_{j+l-1}$ 构成一个等差数列，我们就记为 $l_i(a_j)=l$. 约定，如果以 a_j 为首项的等差数列不存在，那么当 $|S_i|=1$ 时令 $l(a_j)=|S_i|$，当 $|S_i| \geqslant 2$ 时令 $l(a_j)=2$. 令 $l(S_i)=\max\{l(a_j) \mid a_j \in S_i\}$ 并称之为 S_i 的赋值.

显然，对于任意非空数集 $S_i \subset [p]$ 以及任一 $a_j \in S_i$，$l(a_j)$ 与 $l(S_i)$ 都是明确定义的. 根据定义 1 与定义 2，显然有

定理 2　对于数集 $[p]$ 的一个 2- 部分拆 $\pi_2([p])=S_1 \cup S_2$，如果 $c_1=l(S_1), c_2=l(S_2)$，则有 $W(c_1+1,c_2+1) \geqslant p+1$. //

据此，我们有

算法 1　该算法是计算 $W(k_1,k_2)$ 下界的算法，步骤如下：

步骤 1：给定整数 $p \geqslant 5$ 与参数集 $S_1 \subset [p]$. 令 $i=1$.

步骤 2：如果 $s=|S_i| \leqslant 2$，令 $c_i=s$，转到步骤 7；否则，把 S_i 中的元素按从小到大的顺序排列：$a_1 < a_2 < \cdots < a_s$，作成全序集 $(S_i, \prec)$. 令 $c_i=0, j=1$.

步骤 3：令 $d=a_{j+1}-a_j, a=d+a_{j+1}, k=j+2, c=2$.

步骤 4：如果 $a=a_k$，令 $c=c+1, a=a+d, k=k+1$. 如果 $k \leqslant s$，转到步骤 4.

步骤 5：如果 $c_i < c$，令 $c_i=c$. 如果 $c \geqslant 3$，打印首项为 a_j 公差为 d 的 c 项等差数列.

步骤 6：令 $j=j+1$，如果 $j \leqslant s-2$，转到步骤 3.

步骤 7：令 $l(S_i)=c_i$.

步骤 8：令 $i=i+1$. 如果 $i=2$，令 $S_2=[p]-S_1$ 转到步骤 2.

步骤 9：打印 $W(c_1+1,c_2+1) \geqslant p+1$. 运算结束.

在算法 1 中，由步骤 4 输出的 c 便是 a_j 的赋值 $l(a_j)$，进入算法步骤 7 中的 c_i 便是 S_i 的赋值 $l(S_i)$. 在步骤 5 中，如果 $c \geqslant 3$，那么打印出来的首项为 a_j 公差为 d 的 c_i 项等项数列是全序集 $(S_i, \prec)$ 中第一个长为 c_i 的等差数列.

容易看出，步骤 7 给出的 $l(S_i)=c_i$ 便是 S_i 的赋值. 以下在引用算法 1 的步骤 2 至步骤 7 计算 $l(S_i)$ 时，简称"计算 $l(S_i)$".

例 1　令 $p=5, S_1=\{1,2,4,5\}$，则 $S_2=[5]-S_1=\{3\}$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=1$，据定理 2 有 $W(3,2) \geqslant 6$.

不难证明，对于数集 $[6]$ 的任意 2- 部分拆 $\pi_2([6])=\{S_1,S_2\}$，都有 $l(S_1) \geqslant 3$ 或者 $l(S_2) \geqslant 2$，故有 $W(3,2) \leqslant 6$. 即得 $W(3,2)=6$.

我们以这个结论为起点，在假设已知 $W(k_1,k_2-1)$ 的情况下，探索 $W(k_1,k_2)$ 的下界、上界和准确值.

3　P 的子集

给定整数 $k_2 \geqslant k_1 \geqslant 3$，以及 $p > W(k_1,k_2-1)$. 记 $[p]$ 的所有非空真子集的集为 P. 据 $W(k_1,k_2)$ 的定义，显然有

引理 1　如果对于任意 $S_1 \subset P$ 与 $S_2=[p]-S_1$，都有 $l(S_1) \geqslant k_1$ 或者 $l(S_2) \geqslant k_2$，那么 $W(k_1,k_2) \leqslant p$. //

注意到 $|P|=2^p-2$，当 p 稍大时 $|P|$ 就已很庞大了，因此据引理 1 求 $W(k_1,k_2)$ 的上界是很困难的. 在此，我们用 P 的子集取代引理 1 中的 P.

定义 3　设 $S_1=\{a_1,a_2,\cdots,a_s\} \in P$，称 $|S_1|=s$ 为 S_1 的长度，称 $j(S_1)=\max\{a_{j+1}-a_j \mid 0 \leqslant j \leqslant s\}$ 为 S_1 的步长. 这里 $a_1 < a_2 < \cdots < a_s$，并约定 $a_0=0$，$a_{s+1}=p+1$.

定义 4　对于数集 $[p]$ 的任意一个 2- 部分拆 $\pi_2([p])=S_1 \cup S_2$，设 $a \in S_2$，令 $S_i^*=S_i \cup \{a\}$，$S_2^*=S_2-\{a\}$，称为从 $\pi_2([p])=S_1 \cup S_2$ 到 $\pi_2([p])=S_i^* \cup S_2^*$ 的一次调整，简称为对 S_i 增加长度的调整.

显然，增加 S_1 的长度，不会减小其赋值，也不会增大其步长，并且不会增大 S_2^* 的赋值. 即有 $l(S_1^*) \geqslant l(S_1), j(S_1^*) \leqslant j(S_1), l(S_2^*) \leqslant l(S_2)$. 容易证明

引理 2　如果 $S_1 \in P$ 满足 $l(S_1) \leqslant k_1-1$ 并且可作增加长度的调整使 $l(S_1^*)$ 达到或者保持为 k_1-1，那么这样的调整只可作有限次，此后的任何增加

长度的调整都会使 $l(S_1^*) > k_1 - 1$. //

定义 5 对于任意 $S_1 \in P$,$S_i \in G$ 当且仅当它满足条件

(1) $j(S_1) \leqslant k_2$;

(2) $l(S_1) = k_1 - 1$,并且对 S_1 的任何增加长度的调整都会增加其赋值. 即对于任何 $a \notin S_1$,令 $S_i^* = S_i \cup \{a\}$,都有 $l(S_1^*) > l(S_1)$.

引理 3 如果对于任意 $S_1 \in G$,数集 $S_2 = [p] - S_1$ 的赋值 $l(S_2) \geqslant k_2$,那么对于$[p]$的任一 2- 部分拆 $\pi_2([p]) = S_1 \cup S_2$,必有 $l(S_1) \geqslant k_1$ 或者 $l(S_2) \geqslant k_2$.

证明 用反证法. 如若不然,假设存在 $\pi_2([p]) = S_1 \cup S_2$ 使 $l(S_1) \leqslant k_1 - 1$ 并且 $l(S_2) \leqslant k_2 - 1$,则由引理 2 的给定条件可知 $S_1 \notin G$. 有如下情形:

i) S_1 满足定义 5 的条件(2) 而不满足条件(1),即 $j(S_1) \geqslant k_2 + 1$. 那么,至少有一个 j(其中 $0 \leqslant j \leqslant s$) 满足 $a_{j+1} - a_j \geqslant k_2 + 1$,此时令 $C = [a_j + 1, a_{j+1} - 1]$,则 $C \subset S_2$,并且数集 C 中的 $|C|$ 个数构成公差为 1 的等差数列,故有

$$l(S_2) \geqslant |C| = (a_{j+1} - 1) - (a_j + 1) + 1 = (a_{j+1} - a_j) - 1 \geqslant (k_2 + 1) - 1 = k_2.$$

这与假设的 $l(S_2) \leqslant k_2 - 1$ 矛盾.

ii) S_1 满足定义 5 的条件(1) 而不满足条件(2),即 $j(S_1) \leqslant k_2$,$l(S_1) \leqslant k_1 - 1$,并且对 S_1 可作增加长度的调整使 $l(S_1^*) \leqslant k_1 - 1$.

据引理 2,对 S_1 作有限次增加长度的调整,由 2- 部分拆 $\pi_2([p]) = S_1 \cup S_2$ 得到 $\pi_2([p]) = S_1^* \cup S_2^*$ 时,可使 $l(S_1^*) = k_1 - 1$ 并且有 $j(S_1^*) \leqslant j(S_1) \leqslant k_2$,$l(S_2^*) \leqslant l(S_2) \leqslant k_2 - 1$. 此后对 S_1^* 的任何增加长度的调整都会增加其赋值,于是 S_1^* 满足定义 5 的两个条件,即 $S_1^* \in G$. 由引理 3 的给定条件可知 $l(S_2^*) \geqslant k_2$. 矛盾.

iii) S_1 不满足定义 5 的两个条件. 此时可对 S_1 作有限次增加长度的调整, 注意到 $l(S_1^*) \geqslant l(S_1)$,$j(S_1^*) \leqslant j(S_1)$,总可使 S_1^* 满足定义 5 的两个条件之一. 仿上述讨论,即可导致矛盾.

综上述,引理 3 得证. //

显然有 $G \subset P$. 容易证明,对于 $G^* \subset G$,用 G^* 取代引理 3 中的 G,不能肯定得到引理 3 的结论. 因此集 G 是使引理 3 成立的 P 的子集中的阶数最小者. 但容易证明,任何包含 G 作为子集的集都可取代引理 2 中的集 G,即得

引理 4 设 $G \subset H \subset P$,如果对于任意 $S_1 \in H$,数集 $S_2 = [p] - S_1$ 的赋值 $l(S_2) \geqslant k_2$,那么对于$[p]$ 的任一 2- 部分拆 $\pi_2([p]) = S_1 \cup S_2$,必有 $l(S_1) \geqslant k_1$ 或者 $l(S_2) \geqslant k_2$. //

在全序集($[p]$, $\prec$) 中,P 的元可由$[p]$ 的子集按通常的字典排列法作出并且进行排序,第 j 个元记为 P_j. 以下约定 P 及其一切子集(例如下述的 A、B、H、G) 的元都按字典排列法排序,其中第 j 个元以下标 j 记之. 易知 $P_1 = \{1\}$,$P_{|P|} = \{p\}$.

注意到,虽然定义 5 明确定义了集 G,但在实际操作中构造集 G 时,要对每个 $S_1 \in P$ 都判断是否"对 S_1 的任何增加长度的调整都会增加其赋值"却遇到很大的运算量. 采用操作简便的逐步逼近法构造一个包含集 G 的集 H,可以减小运算量.

定义 6 对于任意 $S_1 \in P$,$S_i \in A$ 当且仅当它满足条件

(1) $j(S_1) \leqslant k_2$;

(2) $l(S_1) = k_1 - 1$.

显然 $G \subset A \subset P$,并且 A 中有许多不属于 G 的元. 例如,给定 $k_1 = 3$,$k_2 = 7$,$p = 46$,则 A 中前几个元是

$A_1 = \{1,2,4,5,10,12,17,21,26,27,34,36,39,43\}$,

$A_2 = \{1,2,4,5,10,12,17,21,26,27,34,39,43\}$,

$A_3 = \{1,2,4,5,10,12,17,21,26,27,34,39,43,45\}$,

$A_4 = \{1,2,4,5,10,12,17,21,26,27,34,39,43,45,46\}$,

$A_5 = \{1,2,4,5,10,12,17,21,26,27,34,39,43\}$,

$A_6 = \{1,2,4,5,10,12,17,21,26,27,34,39,43,46\}$.

我们只须比较 A 中相继的两个元 A_i 与 A_{i+1},就可以用简单的方法淘汰这些明显不属于 G 的元. 例如上述 A_2 与 A_3 比较,由 $A_2 \cup \{45\} = A_3$ 可知,对 A_2 作增加长度的调整可得到 A_3,故有 $A_2 \notin G$,于是就可淘汰 A_2. 类似地,A_3 与 A_4 比较,$A_3 \notin G$,可淘汰 A_3. A_5 与 A_6 比较,$A_5 \notin G$,可淘汰 A_5. 让 i 从 1 跑过 $|A| - 1$,淘汰那些不属于G的元 A_i. 最后,把 A 中未被淘汰的元依次编号,就得到 B. 一般地,我们有

定义 7 考察 A 中相继的两个元,如果 $|A_i| = |A_{i+1}| - 1$ 并且 $A_i = \{a_1, a_2, \cdots, a_s\}$,$A_{i+1} = \{a_1, a_2, \cdots, a_s, a_{s+1}\}$,其中 $1 \leqslant i \leqslant |A| - 1$,那么就淘汰 A_i. 把 A 中所有未被淘汰的元作成集 B.

显然 $G \subset B \subset A$，并且 B 中还有许多不属于 G 的元. 例如，给定 $k_1 = 3, k_2 = 7, p = 46$，则 B 中前几个元是

$B_1 = \{1,2,4,5,10,12,17,21,26,27,34,36,39,43\}$，

$B_2 = \{1,2,4,5,10,12,17,21,26,27,34,39,43,45,46\}$，

$B_3 = \{1,2,4,5,10,12,17,21,26,27,34,39,43,46\}$，

$B_4 = \{1,2,4,5,10,12,17,21,26,27,34,39,45,46\}$，

$B_5 = \{1,2,4,5,10,12,17,21,26,27,34,39,46\}$，

$B_6 = \{1,2,4,5,10,12,17,21,26,28,34,36,43,45\}$.

其中 B_3, B_4, B_5 就不属于 G. 我们只须比较 B 中相继的两个元，就可以用简单的方法淘汰这些不属于 G 的元. 例如上述 B_2 与 B_3 比较，显然 $B_3 \notin G$，可以淘汰 B_3. B_2 与 B_4 比较，显然 $B_4 \notin G$，可淘汰 B_4. B_2 与 B_5 比较，显然 $B_5 \notin G$，可淘汰 B_5. 淘汰那些明显不属于 G 的元后，把 B 中未被淘汰的元依次编号，就得到 H. 一般地，我们有

定义 8 在 B 中依次考察的两个元，设 $B_i = \{a_1, a_2, \cdots, a_r\}$，$B_{i+t} = \{b_1, b_2, \cdots, b_s\}$. 下述条件(1)～(7)称为淘汰条件. 按下述算法2由集 B 构造数集 H.

(1) $r = s+1, a_j = b_j (j \in [1, s-1])$ 并且 $a_r = b_s$.

(2) $r = s+1, a_j = b_j (j \in [1, s-2])$ 并且 $a_{r-1} = b_{s-1}, a_r = b_s$.

(3) $r = s+1, a_j = b_j (j \in [1, s-3])$ 并且 $a_{r-2} = b_{s-2}, a_{r-1} = b_{s-1}, a_r = b_s$.

(4) $r = s+2, a_j = b_j (j \in [1, s-1])$ 并且 $a_r = b_s$.

(5) $r = s+2, a_j = b_j (j \in [1, s-2])$ 并且 $a_{r-1} = b_{s-1}, a_r = b_s$.

(6) $r = s+2, a_j = b_j (j \in [1, s-2])$ 并且 $a_{r-2} = b_{s-1}, a_r = b_s$.

(7) $r = s+3, a_j = b_j (j \in [1, s-1])$ 并且 $a_r = b_s$.

算法 2 该算法是由集 B 构造集 H 的算法，步骤如下：

步骤1：给定 P 的子集 B. 令 $i = 1, k = 0$.

步骤2：设 $B_i = \{a_1, a_2, \cdots, a_r\}$，令 $t = 1, k = k+1, H_k = B_i$. 如果 $i = |B|$，转到步骤5.

步骤3：设 $B_{i+t} = \{b_1, b_2, \cdots, b_s\}$，如果 B_i 与 B_{i+t} 符合淘汰条件之一，令 $t = t+1$，转到步骤3.

步骤4：令 $i = i+t$，如果 $i \leqslant |B|$，转到步骤2.

步骤5：运算结束.

定义8与算法2只须比较 B 中相继(B_{i+t} 被淘汰后，B_{i+t+1} 就视为 B_i 的后继元)的两个元，其优点是简单、快捷，用计算机运算时占用内存少. 缺点是淘汰得不够充分，故有 $|G| > |H|$，即由定义8不能淘汰 B 中所有不属于 G 的元. 改进的方法是在定义8中增加一些合理的淘汰条件，可以让所得到的 H 进一步逼近 G，此时 H 仍然使引理3成立. 不过，增加淘汰条件虽然使 $|H|$ 减小而节约了计算 $l(S_2)$ 的时间，但却增加了判断淘汰条件的执行时间，甚至不必淘汰的情形也要进行判断. 因此这种改进未必能提高运算效率.

显然 $G \subset H \subset B$. 由上述 A、B、H 的构造方法，我们有

引理 5 $G \subset H \subset B \subset A \subset P$.

4 计算 $W(k_1, k_2)$ 上界的算法

由引理4、引理5与定义1即得

定理 3 设 H 是满足条件 $G \subset H \subset P$ 的集，如果对于任意 $S_1 \in H$，数集 $S_2 = [p] - S_1$ 的赋值 $l(S_2) \geqslant k_2$，那么 $W(k_1, k_2) \leqslant p$. //

据引理5和定理3，我们有如下计算 Van der Waerden 数 $W(k_1, k_2)$ 上界的算法. 注意到，集 P 可由集 $[p]$ 的所有非空真子集按照字典排列法作出，集 A、B、H 可由定义6、7、8的规定构造，它们都不难操作，因此在下述算法3中我们省略它们构造方法与构造过程的描述.

算法 3 该算法是计算 $W(k_1, k_2)$ 上界的算法，步骤如下：

步骤1：给定整数 $k_2 \geqslant k_1 \geqslant 3$，以及 $p > W(k_1, k_2 - 1)$. 令 $h = 1$.

步骤2：按照字典排列法构造数集 $[p]$ 的非空真子集的集 P.

步骤3：据定义6构造集 A.

步骤4：据定义7构造集 B.

步骤5：据定义8构造集 H. 设集 H 的第 h 个元记为 H_h.

步骤6：令 $S_1 = H_h, S_2 = [p] - S_1$，计算 $l(S_2)$.

步骤7：如果 $l(S_2) < k_2$，打印 $W(k_1, l(S_2)+1)$

$\geqslant p+1$. 令 $p=p+1$,转到步骤 2.

步骤 8:令 $h=h+1$. 如果 $h\leqslant|H|$,转到步骤 5.

步骤 9:打印 $W(k_1,k_2)\leqslant p$. 运算结束.

实际上,在上述步骤 2 至步骤 5 由集 P 开始构造集 A、B、H 的过程中,每确定 H 的一个元 H_h,就立刻进入步骤 6 至步骤 8 的运算过程,不必等到集 H 的元全部确定完毕后才进入步骤 6(即算法 1 的步骤 2 至步骤 7),这就节省了计算机的存储空间和数组 H_h 的存、取时间.

在算法 3 的步骤 1 中,给定整数 k_1,k_2 与 p 的三种情形,有如下结果.

情形 Ⅰ:$W(k_1,k_2-1)<p<W(k_1,k_2)$. 注意到定理 3 的逆否命题也成立,即

"设 H 是满足条件 $G\subset H\subset P$ 的集. 如果 $W(k_1,k_2)>p$,那么存在一个 $S_1\in H$,使数集 $S_2=[p]-S_1$ 的赋值 $l(S_2)<k_2$." 对此作如下分析.

首先,据 H 的定义可知,当 $S_1\in H$ 时有 $l(S_1)=k_1-1$.

其次,据定义 1 可知,$W(k_1,k_2-1)<p$ 给出:$l(S_1)\geqslant k_1$ 或者 $l(S_2)\geqslant k_2-1$. 特别地,当 $l(S_1)=k_1-1$ 时只能有 $l(S_2)\geqslant k_2-1$.

因此,综合上述逆否命题说的"存在一个 $S_1\in H$,使数集 $S_2=[p]-S_1$ 的赋值 $l(S_2)<k_2$(即 $l(S_2)\leqslant k_2-1$)",就只能有 $l(S_2)=k_2-1$.

于是据定理 2 得到一个下界 $W(k_1,k_2)\geqslant p+1$. 此时在步骤 7 中令 $p=p+1$,然后转到步骤 2 继续计算 $W(k_1,k_2)$ 的上界. 经过有限次从步骤 7 转到步骤 2 的运算过程,p 值增加到 $W(k_1,k_2)$ 的时候,就归结到情形 Ⅱ.

情形 Ⅱ:$p=W(k_1,k_2)$. 此时算法 3 的变量 h 从 1 跑过 $|H|$ 而进入步骤 9,我们就证明了:对于每个 $S_1\in H$,集 $S_2=[p]-S_1$ 的赋值 $l(S_2)\geqslant k_2$. 据定理 3 即得 $W(k_1,k_2)\leqslant p$. 于是算法 3 确定了 $W(k_1,k_2)$ 的准确值.

情形 Ⅲ:$p>W(k_1,k_2)$. 此时不存在 $S_1\in H$,使数集 $S_2=[p]-S_1$ 的赋值 $l(S_2)<k_2$. 因此在算法 3 中步骤 7 的条件不能满足,变量 h 从 1 跑过 $|H|$ 而进入步骤 9,我们就得到一个上界 $W(k_1,k_2)\leqslant p$. 但这不是上确界,此时算法 3 不能得到准确值.

例 2 已知 $W(3,2)=6$,我们计算 $W(3,3)$. 给定整数 $k_1=3$,$k_2=3$,$p=7$. 则

$H=\{\{1,2,4,5\},\cdots\cdots\}$. 令 $S_1=H_1=\{1,2,4,5\}$,即得 $l(S_2)=2$,据定理 2 有 $W(3,3)\geqslant 8$. 由算法 3 的 7° 给出 $p=8$,转到 2° 继续计算 $W(3,3)$ 的上界. 此时

$H=\{\{1,2,5,6\},\cdots\cdots\}$. 令 $S_1=H_1=\{1,2,5,6\}$,即得 $l(S_2)=2$,据定理 2 有 $W(3,3)\geqslant 9$. 由算法 3 的步骤 7 给出 $p=9$,转到步骤 2 继续计算 $W(3,3)$ 的上界. 此时

$H=\{\{1,2,5,7\},\{1,3,6,7\},\{1,4,5,8\},\{1,4,6,9\},\{2,3,6,7\},\{2,4,5,7\},\{2,4,7,8\},\{2,5,6,9\},\{2,5,7\},\{3,4,6,7\},\{3,4,7,8\},\{3,5,6,8\},\{3,5,8,9\},\{3,6,7\}\}$.

这里 $|H|=14$. 由算法 3 得到:对于任一 $h\in[1,14]$,令 $S_1=H_h$,$S_2=[9]-S_1$,都有 $l(S_2)\geqslant k_2=3$,据定理 3 有 $W(3,3)\leqslant 9$.

综上述即得 $W(3,3)=9$,简记为 $W_2(3)=9$,即文献[1～3]说的 $W(3,2)=9$. 在例 2 中,当 $k_1=3$,$k_2=3$,$p=9$ 时,H 有 14 个元,其中 $a_1=1$ 的有 4 个,$a_1=2$ 的有 5 个,$a_1=3$ 的有 5 个. 我们把这种情况记为 $|H|=14(4,5,5)$,以下仿此.

顺便指出,在例 2 中当 $k_1=3$,$k_2=3$,$p=9$ 时有 $B=H$ 而 $G=H-\{\{2,5,7\},\{3,6,7\}\}$. 但我们不能象定义 8 那样,用简单快捷的方法(仅比较相继的两个元)在 B 中淘汰不属于 G 的 $\{2,5,7\}$ 与 $\{3,6,7\}$.

注意到,数集[9]的非空真子集有 $2^9-2=510$ 个,相应地[9]的 2-部分拆 $\pi_2([9])=S_1\cup S_2$ 有 510 种情形. 用引理 1 对每种情形都计算 $l(S_1)$ 与 $l(S_2)$ 是不胜其烦的. 在例 2 中我们只须考察 14 种情形就可以了. 在以下关于 $W(3,8)\leqslant 58$ 的证明中,我们只须考虑 $|H|=1.7\times10^7$ 种情形,这 1.7×10^7 远小于 $|P|=2^{58}-2=2.8\times10^{17}$. 可见我们的算法具有较高的运算效率.

5 定理 1 的证明

在证明定理 1 的结论时,为了简便,我们不是像例 2 那样从 $W(3,q-1)+1$ 开始,就每个 $W(3,q)$ 用算法 3 确定其下界、上界与准确值,而是用算法 1 和算法 3 确定分别一批 $W(3,q)$ 的下界和上界,并在计算下界时写出全序集$(S_2,\prec)$中的第一个最长的等差数列.

据算法 1,我们有

引理 6 $W(3,4)\geqslant 18$,$W(3,5)\geqslant 22$,$W(3,6)\geqslant 32$,$W(3,7)\geqslant 46$,$W(3,8)\geqslant 58$.

证明 (1) 令 $p=17$,$S_1=\{4,5,7,11,12,14\}$,则 $S_2=[17]-S_1=\{1,2,3,6,8,9,10,13,15,$

$16,17\}$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=3$，并且全序集$(S_2,\prec)$中第一个长为 c_2 的等差数列是 $1\prec 2\prec 3$，据定理 2 有 $W(3,4)\geqslant 18$.

顺便指出，这里的 S_1 就是在算法 3 中，给定整数 $k_1=3, k_2=4, p=17$ 时的 H 的第 122 个元(其中 $a_1=1$ 的有 47 个元，$a_1=2$ 的有 41 个元，$a_1=3$ 的有 33 个元，这 S_1 是 $a_1=4$ 的第 1 个元).

(2) 令 $p=21, S_1=\{1,2,6,7,9,14,15,18,20\}$，则 $S_2=[21]-S_1=\{3,4,5,8,10,11,12,13,16,17,19,21\}$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=4$，并且全序集$(S_2,\prec)$中第一个长为 c_2 的等差数列是 $4\prec 8\prec 12\prec 16$，据定理 2 有 $W(3,5)\geqslant 22$.

顺便指出，这里的 S_1 就是在算法 3 中，给定整数 $k_1=3, k_2=5, p=21$ 时的 H 的第 31 个元.

为了简便，以下不再写出 S_2 的各项元素.

(3) 令 $p=31, S_1=\{1,2,7,9,14,15,18,24,25,31\}$，则 $S_2=[31]-S_1$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=5$，并且全序集$(S_2,\prec)$中第一个长为 c_2 的等差数列是 $4\prec 6\prec 8\prec 10\prec 12$，据定理 2 有 $W(3,6)\geqslant 32$.

顺便指出，这里的 S_1 就是在算法 3 中，给定整数 $k_1=3, k_2=6, p=31$ 时的 H 的第 507 个元.

(4) 令 $p=45, S_1=\{1,3,8,11,17,18,22,29,30,32,37,39\}$，则 $S_2=[45]-S_1$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=6$，并且全序集$(S_2,\prec)$中第一个长为 c_2 的等差数列是 $4\prec 7\prec 10\prec 13\prec 16\prec 19$，据定理 2 有 $W(3,7)\geqslant 46$.

顺便指出，这里的 S_1 就是在算法 3 中，给定整数 $k_1=3, k_2=7, p=45$ 时的 H 的第 32066 个元.

(5) 令 $p=57, S_1=\{2,5,10,12,17,21,27,28,34,38,43,45,50,53\}$，则 $S_2=[57]-S_1$，由算法 1 得到 $c_1=l(S_1)=2, c_2=l(S_2)=7$，并且全序集$(S_2,\prec)$中第一个长为 c_2 的等差数列是 $1\prec 7\prec 13\prec 19\prec 25\prec 31\prec 37$，据定理 2 有 $W(3,8)\geqslant 58$.

顺便指出，这里的 S_1 就是在算法 3 中，给定整数 $k_1=3, k_2=8, p=57$ 时 H 的第 3987524 个元(其中 $a_1=1$ 的有 2976707 个元，这 S_1 是 $a_1=2$ 的第 1010817 个元). //

据算法 3，我们有

引理 7 $W(3,4)\leqslant 18, W(3,5)\leqslant 22, W(3,6)\leqslant 32, W(3,7)\leqslant 46, W(3,8)\leqslant 58$.

证明 为了简便，以下仅分别写出 H 的排序在最前的与最后的 3 个元.

(1) 令 $k_1=3, k_2=4, p=18$. 则从定义 6 到定义 8 由 P 构造 A、B、H，有 $r=|H|=173(52,47,41,33)$，其中

$H_1=\{1,2,4,8,10,11,13,17\}$,

$H_2=\{1,2,4,8,11,13,16,17\}$,

$H_3=\{1,2,5,7,10,11,14,16\}$,

…,

$H_{r-2}=\{4,8,10,14,15,17\}$,

$H_{r-1}=\{4,8,11,13,15,17\}$,

$H_r=\{4,8,11,15,16\}$.

由算法 3 得到：对于任一 $h\in[1,173]$，令 $S_1=H_h, S_2=[18]-S_1$，都有 $l(S_2)\geqslant k_2=4$，据定理 3 有 $W(3,4)\leqslant 18$.

(2) 令 $k_1=3, k_2=5, p=22$. 则从定义 6 到定义 8 由 P 构造 A、B、H，有 $r=|H|=1143(288,262,234,201,158)$，其中

$H_1=\{1,2,4,5,10,12,17,21\}$,

$H_2=\{1,2,4,5,10,13,17,20\}$,

$H_3=\{1,2,4,5,10,14,17,21,22\}$,

…,

$H_{r-2}=\{5,10,14,17,21,22\}$,

$H_{r-1}=\{5,10,14,19,20,22\}$,

$H_r=\{5,10,14,19,21,22\}$.

由算法 3 得到：对于任一 $h\in[1,1143]$，令 $S_1=H_h, S_2=[22]-S_1$，都有 $l(S_2)\geqslant k_2=5$，据定理 3 有 $W(3,5)\leqslant 22$.

(3) 令 $k_1=3, k_2=6, p=32$. 则从定义 6 到定义 8 由 P 构造 A、B、H，有 $r=|H|=19756(4472,3912,3581,3076,2596,2119)$，其中

$H_1=\{1,2,4,5,10,12,17,21,26,27\}$,

$H_2=\{1,2,4,5,10,12,17,21,27,28,31\}$,

$H_3=\{1,2,4,5,10,13,17,20,26,28,31\}$,

…,

$H_{r-2}=\{6,12,17,23,26,30,31\}$,

$H_{r-1}=\{6,12,17,23,26,31,32\}$,

$H_r=\{6,12,17,23,27,30,32\}$.

由算法 3 得到：对于任一 $h\in[1,19756]$，令 $S_1=H_h, S_2=[32]-S_1$，都有 $l(S_2)\geqslant k_2=6$，据定理 3 有 $W(3,6)\leqslant 32$.

(4) 令 $k_1=3, k_2=7, p=46$. 则从定义 6 到定义 8 由 P 构造 A、B、H，有 $r=|H|=659324(135768,117772,104143,90228,81463,70277,59673)$，其中

H_1 = {1,2,4,5,10,12,17,21,26,27,34,36,39,43},

H_2 = {1,2,4,5,10,12,17,21,26,27,34,39,43,45,46},

H_3 = {1,2,4,5,10,12,17,21,26,28,34,36,43,45},

…,

H_{r-2} = {7,14,20,27,32,39,41,42},

H_{r-1} = {7,14,20,27,32,39,42,43},

H_r = {7,14,20,27,32,39,43,45}.

由算法 3 得到:对于任一 $h \in [1,659324]$,令 $S_1 = H_h, S_2 = [46] - S_1$,都有 $l(S_2) \geqslant k_2 = 7$,据定理 3 有 $W(3,7) \leqslant 46$.

(5) 令 $k_1 = 3, k_2 = 8, p = 58$. 则从定义 6 到定义 8 由 P 构造 A、B、H,有 $r = |H| =$ 17839288(3439416,2976707,2623941,2281773,2009964,1733296,1500412,1273779),其中

H_1 = {1,2,4,5,10,11,14,22,25,31,32,38,41,47,55},

H_2 = {1,2,4,5,10,11,14,22,25,31,35,38,44,47,55},

H_3 = {1,2,4,5,10,11,14,22,25,31,35,41,44,50,54,55},

…,

H_{r-2} = {8,16,23,31,37,45,50,52,56,57},

H_{r-1} = {8,16,23,31,37,45,52,55,56},

H_r = {8,16,23,31,37,45,52,56,57}.

由算法 3 得到:对于任一 $h \in [1,17839288]$,令 $S_1 = H_h, S_2 = [58] - S_1$,都有 $l(S_2) \geqslant k_2 = 8$,据定理 3 有 $W(3,8) \leqslant 58$.

综合引理 6 与引理 7,就证明了定理 1 的结论. 我们在 AMD2400 的计算机上完成上述运算所用的时间是 5 h.

参考文献:

[1] VAN DER WAERDEN B L. Beweis einer banudetschen Vermutung[J]. Nieuw Archief Voor Wiskunde, 1927(15):212-216.

[2] GRAHAM R L, ROTHSCHILD B L, SPENCER J H. Ramsey theory[M]. New York: John Wiley & Sone, 1990.

[3] 李乔. 组合数学基础[M]. 北京:高等教育出版社,1993.

[4] RADZISZOWSKI S P. Small Ramsey numbers[J]. The Electronic Journal of Comb Inatorics, 2004, DS1, #10: 1-48.

(责任编辑:韦廷宗)