# 政党选举中的必要总统

Katarína Cechlárová 科希策
帕沃尔·约瑟夫·沙法里克大学，斯洛伐克
katarina.cechlarova@upjs.sk

Ildikó Schlotter
ELTE经济与区域研究中心，布达佩斯，匈牙利；布达佩斯技术与经济大学，布达佩斯，匈牙利 schlotter.ildiko@krtk.elte.hu

arXiv:2602.10601v1 [cs.GT] 11 Feb 2026

## 摘要
考虑一场选举，其中候选人集合被划分为若干政党，每个政党必须恰好选出一名候选人，提名其参加由所有提名人参与的选举。必要总统问题询问：如果某候选人获得提名，那么无论其他政党如何提名，该候选人是否都会赢得选举。

我们研究了若干投票规则下必要总统问题的计算复杂性。我们证明，对于博尔达、最大最小法和科普兰，该问题对每个 $\alpha \in [0,1]$ 均可在多项式时间内求解；但对于包含 $\ell$-批准制和 $\ell$-否决制在内的一般位置计分规则类别，该问题是 coNP 完全的，即使政党最大规模为二也是如此。对于这类位置计分规则，我们证明，以政党数量为参数时，必要总统问题是 W[2]-困难的；但以选民类型数量为参数时，它是固定参数可处理的。此外，我们证明，即使政党最大规模为二，排序对下的必要总统问题也是 coNP 完全的；以政党数量为参数时，该问题是 W[1]-困难的。值得注意的是，这两项结果即使在选民数量为常数时仍然成立。

## 关键词
选举、政党、候选人提名、必要总统、计算复杂性、固定参数可处理性

## 1 引言
大多数政治选举之前，都会经历一段动荡而激烈的时期，各政党希望决定提名哪位候选人参加选举。提名过程可能采取初选的形式，也可能涉及更复杂的策略性决策；这些决策不仅基于政党成员对候选人特质的看法，也会考虑选民偏好。事实上，随着选举临近，政党可能会发现，此前选定的候选人——例如初选的获胜者——与其他政党的提名人相比，在民调中的支持率较低，因此需要更换。

在近期此类策略性提名的案例中，最著名的或许是乔·拜登在2024年美国总统选举 [33]之前被卡玛拉·哈里斯取代。在波兰，最大的反对派团体公民纲领党在 Małgorzata Kidawa-Błońska 的支持率大幅下滑后，于2020年总统选举 [19]中以 Rafał Trzaskowski 取代了她这位初选获胜者。值得注意的是，这两种情况下，替代候选人都未能确保获胜。

有时，政党的主要目的并非赢得选举本身，而是阻止某位候选人获胜，由此产生各种形式的策略性提名。2022年匈牙利选举中便出现了这种做法：反对党决定合作，建立一份联合提名人名单，以便击败执政党 Fidesz [34]。类似的合作也出现在2023年土耳其选举中，反对派组建了六党联盟，提名 Kemal Kılıçdaroğlu，希望击败执政的主席 Recep Tayyip Erdoğan [24]。这两次努力均未成功。

在这类情形下，会引出几个有趣的问题。本文探讨其中最基本的问题之一：能否通过谨慎选择其余所有政党的提名，使某位指定提名人 $p$在选举中落败？还是说，无论其他政党选择哪些提名人，$p$ 都会必然获胜？

为研究这一主题，我们采用 Faliszewski 等人提出的候选人提名形式化模型 [11]：政党被视为候选人集合，每个政党必须从中恰好提名一位候选人参加即将举行的选举。Faliszewski 等人假设政党了解选民对所有 潜在候选人的偏好，并研究了这一设定下的两个问题：可能当选总统问题询问，一个政党能否以某种方式提名一位候选人，使其在其他政党的某些 提名组合下成为选举的获胜者；而必要总统问题则询问，该政党提名的某位候选人是否无论其他政党如何提名都会成为获胜者。

Faliszewski 等人 [11] 将研究重点放在相对多数制上。对于这一投票规则，他们证明了：当选民的偏好不受限制时，可能当选总统问题是 NP完全的，而必要总统问题是 coNP 完全的。受这些困难性结果的启发，他们转而研究结构化偏好，并证明对于单峰偏好剖面，必要总统问题可以通过多项式时间算法求解。相比之下，即使在一维欧几里得偏好剖面上，可能当选总统问题仍然是 NP完全的；但如果将选举限制为这样的单峰偏好剖面，即每个政党的候选人在社会轴上连续出现，则该问题可以通过多项式时间算法求解。

与针对多种不同投票规则的可能总统问题在经典计算复杂性和参数化计算复杂性方面不断增长的研究相比，关于必要总统问题的已知结果并不多。Cechlárová 等人将 Faliszewski 等人的难解性结果加以推广，证明对于带决选的相对多数制，以及对于每个正整数 $\ell$ 对应的 $\ell$-Approval 和 $\ell$-否决制投票规则，必要总统问题均为 coNP 完全；所有这些

本文是已被第*25*届自主智能体与多智能体系统国际会议（*AAMAS 2026*）接收的论文的完整版。

**表 1：必要总统问题的研究结果。“参数/常数”列列出所考虑的参数或将其限制为常数后的情形；“有效计算”表示能够在多项式时间内高效计算选举获胜者的投票规则。对于科普兰$^{\alpha}$，$\alpha$属于 $[0,1]$。**

| 投票规则 | 参数/常数 | 复杂度 | 参考文献 |
|---|---|---|---|
| 博尔达 | – | in P | 定理 3.1 |
| 简短 | $s=2$ | coNP 完全 | 定理 3.2 |
| | $t$ | W[2]-难，在 XP 中 | 定理 3.4，观察 2 |
| | $\tau$ | 属于固定参数可 tractable | 定理 3.6 |
| 否决型 | $s=2$ | coNP 完全 | 定理 3.3 |
| | $t$ | W[2]-困难，属于 XP | 定理 3.5，观察 2 |
| | $\tau$ | 固定参数可 tractable | 定理 3.7 |
| 科普兰$^{\alpha}$ | – | in P | 定理 4.1 |
| 最大最小法 | – | in P | 定理 4.2 |
| 排序对 | $s=2, \lvert V\rvert=12$ | coNP 完全 | 定理 4.3 |
| | $t, \lvert V\rvert=20$ | W[1]-难，在 XP 中 | 定理 4.4，观察 2 |
| 有效计算 | $s,t$ | 属于固定参数可 tractable 类 | 观察 2 |

结果在每个政党至多有两名候选人的受限情形下仍然成立。作者还针对多种投票规则给出了必要总统问题的整数规划，并利用真实和合成的选举数据进行了计算实验。作为对 Faliszewski 等人关于单峰偏好剖面下简单多数制的必要总统问题可解性结果的补充，Misra 证明，同一问题在单交叉剖面下可在多项式时间内求解。

## 1.1 我们的贡献

我们研究了若干位置计分规则和多种孔多塞一致投票规则下必要总统问题的计算复杂性；我们的结果总结于表 1。我们考察的位置计分规则包括博尔达规则，以及 Schlotter 等人 [31]提出的所有所谓短式和否决型投票规则。此外，我们还考察了一些最著名的孔多塞一致投票规则：科普兰$^{\alpha}$ 对所有 $\alpha \in [0,1]$、最大最小法和排序对。

对于上述每种投票规则，我们将必要总统问题的复杂性归类为多项式时间可解或 coNP 完全。此外，我们运用参数化复杂性框架来处理计算上难以求解的情形：我们考察给定实例的某些自然参数如何影响必要总统问题的计算复杂性。我们考虑的参数如下：

- $t$：政党数量；
- $s$：政党的最大规模；
- $\tau$：选民类型的数量；若两名选民对候选人的偏好相同，则他们属于同一类型；• $|V|$：选民数量。

对于所有必要总统问题被证明为 coNP 完全的投票规则，我们确定其在上述四个参数的每种可能组合下的参数化复杂度，结果分别为：(i) 固定参数可处理（FPT）；(ii) W[1]-困难或 W[2]-困难，且属于 XP；或 (iii) para-NP 困难。

## 1.2 相关工作

与本文最密切相关的研究方向——选举前由政党进行的策略性候选人提名问题——由 Faliszewski 等人 [11] 开创，除前文提及的论文外，[3, 11], 其研究主要聚焦于可能当选总统问题；参见第 1.2.1 节。第 1.2.2 节简要讨论了从算法角度描述政党选举的不同模型。另一个相关主题是策略性候选，在这种情形下，候选人彼此独立，并完全有权决定是否参加选举；参见第 1.2.3 节。

候选人提名也可以视为候选人集合不固定这一更广泛的选举议题的一部分；Bartholdi 等人 [18] 的开创性论文中已出现了这类情形，即由选举主持人通过增加或删除候选人来进行选举控制。关于此类选举控制的其他研究成果综述，例如参见 Faliszewski 和 Rothe [13], 、 Chen 等人 [4] 或 Erdélyi 等人 [10]。

*1.2.1* 可能主席问题。Faliszewski 等人 [11] 关于相对多数制投票下可能主席问题的研究结果，已扩展至其他投票规则。Lisowski [22] 研究了竞赛解，并证明相对于孔多塞规则（若存在孔多塞胜者，则选出该胜者；否则选出空集），可能主席问题可在多项式时间内求解；而对于未覆盖集规则，该问题是 NP完全的。Cechlárová等人 [3] 研究了位置计分投票规则下的问题，包括 $\ell$-批准制、 $\ell$-否决制和博尔达规则，以及科普兰、Llull 和最大最小法等孔多塞一致规则。此外，本文还针对所有研究的投票规则，为可能主席问题和必要总统问题提供了整数规划，并通过计算实验，将这些整数规划应用于真实选举和合成选举。

Misra [23] 首先研究了可能主席问题的参数化复杂性，并通过证明相对多数制下的可能主席问题即使在偏好剖面同时满足单峰和单交叉时，按最大政党规模参数化仍为 para-NP-hard，强化了 Faliszewski 等人 [11] 的结果；她还证明，按政党数量参数化时，该问题在一般偏好下为 W[1]-hard 且属于 XP，而在一维欧几里得域上则属于固定参数可 tractable。

Schlotter 等人 [31] 对若干类位置计分规则下的可能主席问题进行了详细的多变量复杂性分析，研究了本文所考察的四个参数下这些问题的参数化复杂性。Schlotter 和 Cechlárová [30] 也采用了相同的多变量分析方法，研究了两类孔多塞一致投票规则：适用于每个 $\alpha \in [0,1]$ 的科普兰$^{\alpha}$，以及最大最小法。

*1.2.2* 涉及政党的选举。 Harrenstein 等人 [15] 提出了一种模型，其中选民和候选人都由其在实数轴（用于表示政治光谱）上的位置来描述，并假设每位选民都会投票给距离自己最近的提名人，各政党必须从各自的潜在候选人中选出提名人。作者证明，即使在两党博弈中，纳什均衡（NE）也不一定存在；一般情况下，寻找纳什均衡是 NP完全问题，但对于两个政党，可以在线性时间内计算得出。

Deligkas 等人 [6]扩展了上述模型：他们为每位候选人关联了一个成本，并研究了均衡计算问题的参数化

复杂性，考察了若干自然参数，例如候选人不同位置的数量、提名人的所谓差异度和跨度，以及政党的最大重叠度。

Harrenstein和Turrini [16] 研究了基于选区的选举。在每个选区中，选民对提名的候选人进行排序，并选出相对多数制获胜者；政党则必须策略性地将候选人安排到各个选区，以最大化当选的提名人数量。作者证明，当政党规模有常数上界时，判定这些博弈是否存在纯纳什均衡是NP完全的；一般情况下则为 $\Sigma_2^P$完全。

Perek等人 [27] 提出了一种模型，其中被划分为政党的不是候选人，而是选民；同一政党的选民以相同的方式投票。作者提出，通过以下两项指标衡量对所谓领先政党 $P$——即其青睐的候选人预计将在选举中获胜的政党——的威胁：在不改变选举获胜者的情况下，可以转投其他政党的选民最大数量；以及为确保获胜者发生变化而必须离开的选民最少数量。Perek等人 [27] 以及后续论文中的Guo等人 [14] 研究了这些问题在若干不同投票规则下的计算复杂性。

*1.2.3* 策略性候选。在策略性参选博弈中，假设选民和候选人都对可能的选举结果有偏好。候选人会策略性地选择参加或退出选举。Lang 等人 [21] 展示了这一现象在现实中的表现：在2017年法国总统选举中，中间派候选人 Bayrou 退出竞选，以帮助 Macron 成功晋级第二轮；绿党候选人 Jadot 也退出竞选，以帮助社会党候选人 Hamon 晋级，但未能成功。类似情况也出现在斯洛伐克总统选举史上：2019年2月，Robert Mistrík 退选并支持 Zuzana Čaputová [32], ；2024年3月，议会副主席 Andrej Danko 退出总统候选资格，转而支持前司法部长 Štefan Harabin [26]。

Dutta等人 [9] 考虑了这样一个框架：其中有一个有限的选民集合和潜在候选人集合，且部分候选人也可能是选民。每个人（包括候选人）都对所有候选人有偏好，并且假设每位候选人都比起其他候选人更偏好自己。作者考察了一个两阶段程序：第一阶段由候选人决定是否参加选举，第二阶段则实施投票程序，从参选者中选出获胜者。如果对于所有候选人都参加的情况总是构成纳什均衡（简称 NE），则称投票程序为候选人稳定。作者证明，如果选民集合与候选人集合不相交，那么唯一候选人稳定的投票程序就是独裁程序。当候选人集合与选民集合存在交集时，则存在满足候选人稳定性和一致性的非独裁投票程序。

Brill 和 Conitzer [2] 将分析扩展到选民也采取策略性行动的情形。首先，他们研究投票规则满足多数一致性、但不满足孔多塞一致性时，具有单峰偏好的策略性候选。他们定义了几种稳定性概念，并证明当候选人采取强均衡策略时且选民如实投票，结果将是孔多塞胜者。对于逐次淘汰投票，他们证明计算候选人稳定集是 NP完全的。

Lang 等人 [21] 针对一系列常见投票程序对此类博弈进行了分析，其中包括位置计分规则（包括相对多数制、否决制和博尔达）、孔多塞一致的投票规则（最大最小法、未覆盖集和科普兰）、带决选的相对多数制以及单一可转移投票制。所研究的问题是，此类博弈是否拥有一种纯策略纳什均衡，其结果与所有候选人均参选时相同（称为真正均衡）。由此得到若干负面结果：除非候选人数量较少，否则可能存在不具备此类稳定结果的博弈；但对于Copeland规则，在任意数量的候选人以及奇数选民数量的情况下，真正均衡的存在性均可得到保证。此外，还确立了候选人提名博弈的均衡与一种通过增加或移除候选人实施的投票控制形式之间的紧密关系，其中候选人必须同意增加或删除。

其他研究假定，候选人的偏好不仅取决于选举结果，还会考虑开展竞选活动的金钱成本与声誉成本。Obraztsova 等人 [25]假定候选人是懒惰的，即他们从开展竞选活动中获得的额外价值为负（竞选会产生一定成本）；而Lang 等人 [20] 则假定该价值为正（参与选举使候选人有机会宣传其政党或政治纲领，从而提升其知名度/声誉）。

Polukarov 等人 [29] 和 Obraztsova 等人 [25] 的另一条研究路线探讨候选人提名博弈中的均衡动态：在每一轮过程中，候选人都可以策略性地决定参选或退出提名，直到过程收敛到稳定状态（如果能够收敛的话）。

## 2 预备知识

我们用记号 $[i] = \{1、2、\ldots\ldots、i\}$ 表示每个正整数 $i$。

一次选举 $\mathrm{E} = (C, V, \{\succ_v\}_{v \in V})$ 由有限的候选人集合 $C$、有限的选民集合 $V$ 以及选民对候选人集合 $C$ 的偏好构成。我们假设，选民 $v$ 的偏好由候选人集合 $C$ 上的严格线性序 $\succ_v$ 表示，其中 $c \succ_v c'$ 表示选民 $v$ 更偏好候选人 $c$ 而非候选人 $c'$。如果两名选民的偏好相同，则称他们属于同一类型；$V$ 中的选民类型数量记为 $\tau$。

我们将候选人集合 $C$ 上的所有选举所构成的集合记为 $\mathbb{E}_C$。投票规则 $\mathrm{R}: \mathbb{E}_C \to 2^C$ 会从该选举中选出一组获胜者 。

我们还假设给定了候选人集合 $C$ 的一个划分 $\mathrm{P} = \{P_1, \ldots\ldots, P_t\}$；每个集合 $P_j$ 都被视为一个政党，它必须决定从其潜在候选人中提名谁参加选举。形式上，在每个政党都提名了一位唯一的候选人后，就产生一个缩减选举，并形成一个提名人集合 $C' \subseteq C$，使得 $|C' \cap P_j| = 1$ 对每个 $j \in [t]$ 均成立。在缩减选举 $\mathrm{E}_{C'} = (C', V, \{\succ'_v\}_{v \in V})$ 中，每位选民 $v \in V$ 将其在 $C$ 上的原始偏好关系 $\succ_v$ 限制到 $C'$，从而得到 $\succ'_v$。

现在，我们来表述本文研究的问题。

**问题** 投票规则 R 下的必要总统。**实例：** 一个元组 $I = (\mathrm{E}, \mathrm{P}\ p)$，其中 $\mathrm{E} = (C,\ V,\ \{\succ_v\}_{v\in V})$ 是一次选举，候选人集合为 $C$，选民集合为 $V$；P 是 $C$ 的一个政党划分；另有一名指定候选人 $p \in C$。**问题：** $p$ 是否为必要总统？也就是说，对于所有来自不包含 $p$ 的政党的可能提名，若由此产生的提名人集合为 $C'$，且 $p \in C'$，指定候选人 $p$ 是否为缩减选举 $\mathrm{E}_{C'}$ 在 $C'$ 中的获胜者？

请注意，我们考虑的是非唯一获胜者模型，因此，如果 $p$ 在所有包含 $p$的可能约化选举中都属于获胜者之一，我们就称其为必要主席。包含指定候选人 $p$ 的政党或许还包含其他候选人，但在必要主席问题中，这些候选人无关紧要。

例 *2.1*。 考虑一场有三个政党的选举：$A = \{a_1,\ a_2\}$、$B = \{b_1,\ b_2\}$和 $P = \{p\}$。假设每个政党都必须为即将举行的一场选举恰好提名一名候选人，且该选举采用博尔达投票规则。设选民集合为$\{v_1,\ v_2,\ v_3\}$，他们的偏好如下。

$$\begin{aligned} v_1 &: \ p,\ \ a_1,\ \ b_1,\ \ a_2,\ \ b_2 \\ v_2 &: \ a_1,\ \ p,\ \ b_1,\ \ a_2,\ \ b_2 \\ v_3 &: \ b_1,\ \ b_2,\ \ a_2,\ \ p,\ \ a_1 \end{aligned}$$

请注意，无论政党 $A$ 提名 $a_1$ 还是 $a_2$，$p$ 都能从选民 $v_1$ 那里获得 2 分，并从 $v_2$ 和 $v_3$那里再各获得 2 分：若提名 $a_1$，$p$ 会从这两位选民那里各获得 1 分；若提名 $a_2$，则 $p$ 会从 $v_2$ 那里获得 2 分，但无法从 $v_3$那里得分。因此，无论其他政党如何提名，$p$在缩减选举中的得分都是 4。由此可见，$p$ 是必要主席，因为其他候选人都无法获得超过 4 分；若要超过 4 分，政党 $A$ 或 $B$ 的某位候选人就必须从两位不同的选民（即 $v_2$ 和 $v_3$）那里各获得 2 分，而这是不可能的。⌟

为了验证给定候选人 $p$ 不是必要总统，只需给出一个包含 $p$、且其中 $p$ 不是获胜者的缩减选举。因此，我们有以下事实。

观察 1。 对于每种能够在多项式时间内完成获胜者判定的投票规则，必要总统问题都属于 coNP 。

正如 Schlotter 等人 [31],等人对可能主席问题的观察所示，政党最多有 $s^t$ 种可能的提名方式，其中 $s$ 和 $t$ 分别表示最大规模和政党数量。因此，简单的蛮力法可得：

观察 2。 对于每种能够在多项式时间内完成获胜者判定的投票规则，以政党数量 $t$ 为参数时，必要总统问题属于 XP ；以 $t$ 和政党最大规模 $s$ 为参数时，则属于 固定参数可 tractable 。

## 2.1 投票规则

本文将讨论两类投票规则：位置计分规则和孔多塞一致规则。对于所有考察的投票规则，任何选举中的获胜者都可以高效计算（即在多项式时间内计算），因此根据观察 1，我们知道，对于本文研究的所有投票规则，必要总统问题属于 coNP。[1]

[1]我们指出，在排序对规则中，高效的获胜者判定需要采用某种破同票方法。

*2.1.1* 位置计分规则。对于涉及 $t$ 个候选人的选举，位置计分规则由一个计分向量$(a_1, a_2,\ \ldots\ldots,\ a_t)$ 确定，其中 $a_1 \geq a_2 \geq \cdots \geq a_t$，且至少一个不等式为严格不等式。对于每位候选人 $c$，该规则会根据每位选民将 $c$ 列在其偏好列表第 $i^{\text{th}}$ 位，给 $c$ 计 $a_i$ 分。选举的获胜者是得分最高的候选人；得分即所获总分。我们用 $\mathrm{scr}_{\mathcal{E}}(c)$ 表示候选人 $c$ 在选举 E 中的得分。

我们讨论以下（类别的）位置计分规则。短计分规则由 Schlotter 等人在 [31], 中提出，其计分向量只有常数个非零位置，即形式为（$a_1,\ a_2,\ \ldots\ldots,\ a_\ell,\ 0,\ \ldots\ldots,\ 0$），其中 $\ell$ 为某个常数。换言之，在此类选举中，选民只给自己最偏好的 $\ell$ 个候选人分配分数。这类投票规则包含广为人知的计分规则：当 $\ell$ 固定时的 $\ell$-赞成制，其计分向量的前 $\ell$ 个位置为 1，之后均为 0。当 $\ell = 1$ 时，该规则为相对多数制，选民只给自己最偏好的候选人分配 1 分。

类否决计分规则的计分向量在除最后 $\ell$ 个位置以外的每个位置上都包含某个值 $a$，其中 $\ell$ 为某个常数；也就是说，其形式为（$a,\ \ldots\ldots,\ a,\ a_1,\ a_2,\ \ldots\ldots,\ a_\ell$），其中 $\ell \geq 1$ 和 $a > a_1$ 为常数。换言之，在此类选举中，选民只区分自己最不喜欢的 $\ell$ 个候选人。类否决计分规则包括 $\ell$-否决制，其计分向量为（1，1，……，1，0，0，……，0），其中恰有 $\ell$ 个零；当 $\ell = 1$ 时，该规则称为否决制。

最后，我们还考虑博尔达投票规则，其由计分向量 $(t-1,\ t-2,\ \ldots\ldots 1,\ 0)$ 描述。因此，候选人 $c$ 从选民 $v \in V$ 那里获得的分数，等于在 $v$的偏好列表中排名低于 $c$ 的候选人数量。

*2.1.2* 孔多塞一致规则。对于两个候选人 $c,\ c' \in C$，令 $N_{\mathcal{E}}(c, c')$ 表示在选举 E 中偏好候选人 $c$而非候选人 $c'$ 的选民数量；当 E 的含义可由上下文确定时，我们将省略下标。如果 $N_{\mathcal{E}}(c, c') > N_{\mathcal{E}}(c',\ c)$，则称候选人 $c$击败候选人 $c'$；如果 $N_{\mathcal{E}}(c, c') = N_{\mathcal{E}}(c',\ c)$ 且 $c \neq c'$，则候选人 $c$和 $c'$在 E 中平票。孔多塞胜者是击败所有其他候选人的候选人；如果孔多塞胜者存在时，投票规则总是选出该胜者，则称该规则为孔多塞一致。

科普兰$^{\alpha}$投票规则由 Faliszewski 等人定义，[12] 其中常数为 $\alpha \in [0, 1]$。此投票规则考察所有不同候选人组成的配对 $(c、c')$。根据两人的一对一比较，击败对方的候选人得 1 分，被击败的候选人得 0 分；平票时，两位候选人均得 $\alpha$ 分。形式上，在 E 中，根据 $c$ 与 $c'$ 的一对一比较，$c$ 获得的得分为

$$\mathrm{Cpl}^{\alpha}_{\mathcal{E}}(c, c') = \begin{cases} 1 & \text{if } c \text{ defeats } c' \text{ in } \mathcal{E}; \\ \alpha & \text{if } c \text{ and } c' \text{ are tied in } \mathcal{E}; \\ 0 & \text{if } c \text{ is defeated by } c' \text{ in } \mathcal{E}. \end{cases} \tag{1}$$

然后，候选人 $c$的科普兰$^{\alpha}$得分计算为 $\mathrm{Cpl}^{\alpha}_{\mathcal{E}}(c) = \sum_{c' \in C\setminus\{c\}} \mathrm{Cpl}^{\alpha}_{\mathcal{E}}(c、\ c')$ 的总和。E 的获胜者是所有得分最高的候选人。参数为 $\alpha = 1$时的科普兰$^{\alpha}$规则称为 *Llull* 规则，而参数为 $\alpha = 0$时的规则称为科普兰规则。[2]

在*Maximin*投票规则中，选举 E 中候选人 $c$ 的 Maximin 分数为 $\mathrm{MM}_{\mathcal{E}}(c) = \min_{c' \in C\setminus\{c\}} N_{\mathcal{E}}(c, c')$。换言之，

[2]请注意，有些论文，尤其是 [12],，将科普兰规则称为科普兰$^{0.5}$。

候选人的 Maximin 分数 $c$ 是满足以下条件的最大整数 $r$：对于每个其他候选人 $c'$，都存在 $r$ 名选民更偏好 $c$ 而非 $c'$。同样，E 的获胜者是得分最高的候选人。

排序对投票规则使用选举的所谓多数图：一个有向图 $D_{\mathcal{E}} = (C, A)$，其中顶点是候选人，并且对于两个不同的候选人 $c$和 $c'$，$(c, c')$ 是一条弧，当且仅当 $c$在选举 E 的两两比较中击败 $c'$。排序对的获胜者判定过程通过考察 $A$ 中的候选人对集合来构建一个无环的[3]弧集，并按其权重从高到低依次考察，其中 $(c, c') \in A$的权重为 $N_{\mathcal{E}}(c, c')$。该过程从一个空弧集 $F$开始，检查当前考察的弧能否加入 $F$ 而不产生任何环；如果可以，就将其加入 $F$；随后继续考察下一条弧。为简单起见，我们只考虑多数图 $D_{\mathcal{E}}$ 中存在的弧，因此不考虑平票候选人之间的弧。不过，这一过程通常需要采用某种破同票方法，以确定权重相同的弧的排序。获胜者是 $F$中所有没有入弧的候选人。我们指出，我们的结果不依赖于任何特定的破同票方法，而是适用于任意破同票方法。

## 2.2 参数化复杂性

参数化问题 $Q$ 为每个输入实例 $I$关联一个整数参数 $k$，使我们能够将求解 $Q$的算法的运行时间表示为输入长度 $|I|$ 和参数的函数。若 $Q$关于参数 $k$ 存在一个运行时间为 $f(k)|I|^{O(1)}$的算法，其中 $f$是某个可计算函数，则称 $Q$是固定参数可处理（FPT）的；这样的算法称为关于参数 $k$的*FPT*算法 。

相比之下，若某个算法的运行时间为 $|I|^{f(k)}$，其中 $f$是某个可计算函数，则称该算法为关于参数 $k$的 XP算法；存在此类算法的参数化问题称为属于复杂性类XP。若通过参数化归约证明某个参数化问题 $Q$ 是 W[1]-困难（甚至是 W[2]-困难）的，则意味着除非标准复杂性理论假设FPT $\subsetneq W[1] \subsetneq W[2]$ 不成立，否则 $Q$关于给定参数不是固定参数可处理的。若某个参数化问题在参数取某个常数值时就已经是 NP-困难，则这是难解性的更强表征；此时，$Q$称为*para*-NP-困难。

关于参数化复杂性的更多背景知识，请参阅书籍 [5, 8]。

# 3 位置计分规则的结果

本节介绍我们关于三类位置计分规则下必要总统问题的参数化复杂性结果。第 3.1 节讨论博尔达规则，第 3.2 节则研究短规则和类否决规则。

## 3.1 博尔达规则

我们首先证明，对于博尔达投票规则，必要总统问题是多项式时间可解的。鉴于密切相关的博尔达可能当选总统问题在计算上是困难的，这一易处理性结果颇为出人意料 [31]。

[3]弧集 $F$是无环的，当且仅当生成有向图$(C, F)$中不存在有向环；基本图论术语请参见 Diestel 的教材 [7]。

定理 3.1. 博尔达规则下的必要总统可在多项式时间内求解。

证明。我们提出一种多项式时间算法，用于求解博尔达投票规则下的必要总统问题；伪代码见 NP-Borda 算法。令 $I = (\mathrm{E}, \mathrm{P}, p)$ 为我们的输入实例。

假设存在一组 $C' \ni p$ 提名候选人，使得在缩减选举 $\mathrm{E}_{C'}$（候选人集合为 $C'$）中，$p$ 不是获胜者。设包含 $p$ 的政党为 $P$。NP-Borda 算法首先猜测一个候选人 $w \in P_w \in \mathrm{P} \setminus \{P\}$，其在 $\mathrm{E}_{C'}$ 中的得分高于 $p$ 的得分；随后，从每个剩余政党中贪心地提名一名候选人。注意，这里的“猜测” $w$ 指遍历所有可能的候选人选择。

NP-Borda 算法通过为每个候选人 $c \in C \setminus (P \cup P_w)$ 计算以下值来执行：

$$\Delta(c) = |\{v : v \in V, w \succ_v c \succ_v p\}| - |\{v : v \in V, p \succ_v c \succ_v w\}|. \quad (2)$$

注意，数量 $\Delta(c)$ 表示提名候选人 $c$ 后，$w$ 的得分超过 $p$ 得分的幅度。

接下来，对于每个政党 $\widetilde{P} \in \mathcal{P} \setminus \{P, P_w\}$，NP-Borda 算法提名一个使 $\Delta(c_{\widetilde{P}})$ 最大化的候选人 $c_{\widetilde{P}} \in P$，并检查在由此得到的选举中 $p$ 是否不是获胜者；若是，则输出“否”。如果算法已遍历 $w$ 的所有可能猜测，却仍未返回“否”，则返回“是”。

**NP-Borda 算法**求解博尔达的必要总统问题。

**输入：** 一个必要总统问题实例（E, P, $p$），其中包含候选人集合 $C$ 和 $p \in P \in \mathrm{P}$。

**对于所有** $w \in C \setminus P$，**执行**
  令 $P_w$ 为 P 中包含 $w$的政党。
  **对于所有** $c' \in C \setminus (P \cup P_w)$，**执行**
    按照（2）计算 $\Delta(c)$。
  设定 $\widetilde{C} = \{p, w\}$。
  **对于所有** $\widetilde{P} \in \mathcal{P} \setminus \{P, P_w\}$，**执行**
    将某个候选人 $c_{\widetilde{P}} \in \arg\max_{c \in \widetilde{P}} \Delta(c)$ 添加到 $\widetilde{C}$。
  **如果** $p$ 不是 $\mathrm{E}_{\widetilde{C}}$ 中的获胜者，**则返回** “否”。
**返回**“是”。

让我们证明 NP-Borda 算法的正确性。显然，每当 NP-Borda 算法返回“否”时，它都是正确的，因为对于集合 $C$（其中恰好包含来自 P 中每个政党的一个提名候选人），由政党 $P$ 提名的候选人 $p$ 在缩减选举 $\mathrm{E}_{\widetilde{C}}$ 中并非获胜者，其中 $\widetilde{C}$。

因此，接下来只需证明：每当输入是必要总统问题的一个“否”实例时，NP-Borda 算法都会返回“否”。设 $C'$ 为某个缩减选举 $\mathrm{E}_{C'}$中的提名人集合，其中 $p$ 是一名提名人，但不是获胜者。于是存在某个候选人 $w \in P_w \in \mathrm{P} \setminus \{P\}$，使得 $\mathrm{scr}_{\mathcal{E}_{C'}}(w) > \mathrm{scr}_{\mathcal{E}_{C'}}(p)$。我们将偏好 $p$ 胜于 $w$的选民集合记为 $V^{p \succ w}$，将其余选民记为$V^{w \succ p} = V \setminus V^{p \succ w}$。

我们的关键观察如下：如果在集合 $\mathrm{E}_{C'}$ 中，对于选民 $v \in V^{w \succ p}$，恰好有 $i$ 位提名人是其偏好高于 $p$、但不高于 $w$ 的对象，那么由于选民 $v$，博尔达规则会在 $\mathrm{E}_{C'}$ 中给 $w$ 分配的分数比给 $p$ 多 $i+1$ 分。类似地，如果对于某位选民 $v \in V^{p \succ w}$，在集合 $\mathrm{E}_{C'}$ 中恰好有 $i$ 位提名人是其偏好高于 $w$、但不高于 $p$ 的对象，那么

由于选民 $v$，博尔达规则给 $p$ 分配的分数比给 $w$ 多 $i+1$ 分。因此，我们得到

$$
\begin{aligned}
0 &< \mathrm{scr}_{\mathcal{E}_{C'}}(w) - \mathrm{scr}_{\mathcal{E}_{C'}}(p) = \\
&= \sum_{v \in V^{w \succ p}} \left( \left| \bigcup_{\substack{c \in C', \\ w \succ_v c \succ_v p}} \{c\} \right| + 1 \right) - \sum_{v \in V^{p \succ w}} \left( \left| \bigcup_{\substack{c \in C', \\ p \succ_v c \succ_v w}} \{c\} \right| + 1 \right) \\
&= \sum_{c \in C' \setminus \{p,w\}} \Delta(c) + |\{v : v \in V, w \succ_v p\}| - |\{v : v \in V, p \succ_v w\}| \\
&\leq \sum_{c \in \widetilde{C} \setminus \{p,w\}} \Delta(c) + |\{v : v \in V, w \succ_v p\}| - |\{v : v \in V, p \succ_v w\}| \\
&= \sum_{v \in V^{w \succ p}} \left( \left| \bigcup_{\substack{c \in \widetilde{C}, \\ w \succ_v c \succ_v p}} \{c\} \right| + 1 \right) - \sum_{v \in V^{p \succ w}} \left( \left| \bigcup_{\substack{c \in \widetilde{C}, \\ p \succ_v c \succ_v w}} \{c\} \right| + 1 \right) \\
&= \mathrm{scr}_{\mathcal{E}_{\widetilde{C}}}(w) - \mathrm{scr}_{\mathcal{E}_{\widetilde{C}}}(p).
\end{aligned}
$$

其中，$C$ 是在第 3 行选出 $w$ 的那次迭代中，NP-Borda 算法在第 5–7 行计算得到的集合。注意，$\leq$ 不等式成立，是因为根据第 7 行的保证，每个提名人 $c \in C$ 在其所属政党的所有候选人中，使 $\Delta(c)$ 最大。由此可知，NP-Borda 算法在第 8 行会发现 $p$ 不是选举 $\mathrm{E}_{\widetilde{C}}$ 的获胜者，因为 $\mathrm{scr}^{\mathcal{E}_{\widetilde{C}}}(w) > \mathrm{scr}^{\mathcal{E}_{\widetilde{C}}}(p)$，因此会按要求输出“否”。

注意，选择 $w$ 的可能性少于 $|C|$ 种。$\Delta(c)$ 的值可以在 $O(|V| \cdot |C|)$ 时间内计算出来，随后也可以在 $O(|C|)$ 时间内构造 $\widetilde{C}$。最后，在候选人集合 $\widetilde{C}$ 上的缩减选举中，$p$ 和 $w$ 的得分也可以在 $O(|V| \cdot |C|)$ 时间内计算，因此 NP-Borda 算法的总运行时间为 $O(|C|^2 \cdot |V|)$。 □

## 3.2 短计分规则与类否决计分规则

本节将详细考察所有短计分规则和类否决计分规则下必要总统问题的计算复杂性。我们首先证明，即使最大政党规模为 $s = 2$，在此类投票规则下，必要总统问题仍然是 coNP 完全的。因此，对于短计分规则，我们推广了 Faliszewski 等人 [11] 的一项结果；他们证明了相对多数制下的必要总统问题是 coNP 完全的。

定理 3.2 和 3.3 的证明基于从 (2,2)-E3-SAT 到该问题的多项式归约。(2,2)-E3-SAT 要判定的是：给定一个 3-CNF 公式，其中每个变量恰好以正文字和负文字的形式各出现两次，该公式是否可满足。Berman 等人 [1] 已证明此问题是 NP 完全的。在陈述定理 3.2 之前，我们先正式定义 (2,2)-E3-SAT 问题：

> **问题** (2,2)-E3-SAT
> **实例：** 一个定义在变量集合 $X = \{x_1, \ldots, x_n\}$ 上的 CNF 布尔公式 $\varphi = C_1 \wedge \cdots \wedge C_m$，其中每个子句 $C_j$ 恰好包含三个互不相同的文字，且每个变量都恰好以正文字和负文字各出现两次。**问题：** $\varphi$ 是否可满足？

此后，对于每个 $j \in [m]$，我们用 $y_j^1$、$y_j^2$ 和 $y_j^3$ 表示 $C_j$ 中的三个文字。

定理 3.2。设 $\mathcal{R}$ 是基于位置计分向量的一种短投票规则，其形式为 $(a_1, a_2, \ldots, a_\ell, 0, \ldots, 0)$，其中某个常数 $\ell \geq 1$ 满足 $a_\ell > 0$。那么，即使最大政党规模为 $s = 2$，针对 $\mathcal{R}$ 的必要总统问题仍是 coNP-完全的。

证明。设 $\varphi = C_1 \wedge \cdots \wedge C_m$ 是变量集 $X = \{x_1, \ldots, x_n\}$ 上的一个 (2,2)-E3-SAT 实例。(2,2)-E3-SAT 布尔公式的结构意味着 $m \geq 2$。

我们将构造一个 $I$ 实例，其中必要总统的指定候选人 $p$ 属于政党 $P = \{p\}$，并使得：对于其他政党的每一种可能提名集合，$p$ 是缩减选举的唯一获胜者，当且仅当 $\varphi$ 不可 满足。

除 $P$ 外，我们还为每个变量 $x_i \in X$ 定义政党 $P_w = \{w\}$ 和政党 $\overline{X_i} = \{x_i, \overline{x_i}\}$。此外，我们还定义一个包含 $|D_h| = \ell - 1$ 个虚拟候选人的集合 $D_1 \cup D_2 \cup D_3 \cup D_4$，其中 $h = 1$、2、3、4 对应的每个虚拟候选人都单独组成一个政党。注意，政党数量为 $n + 4\ell - 2$，且任何政党的最大规模为 $s = 2$。

选民集合为

$$V = W \cup U \cup \overline{U} \cup V_0 \cup V_0' \qquad |V_0| = \{\}, \text{其中，} |V_0'| = 2m+1 \qquad U = u_1 \ldots u_m \quad \overline{U} = \overline{u_1} \ldots \overline{u_m}$$

和 {„}，{„}。因此，共有 $6m + 3$ 名选民。偏好剖面如下所示。请注意，在此证明及其他证明中，$[\cdots]$ 表示其余候选人（偏好列表中未明确列出），并按任意严格顺序排列；此外，当我们在偏好列表中写出候选人的一个子集时，表示该集合中的候选人按任意严格顺序排列。

| | |
|---|---|
| $w$： | $D_1 \succ w \succ [\cdots]$； |
| $v \in V_0$： | $w \succ D_2 \succ [\cdots]$； |
| $v' \in V_0'$： | $p \succ D_3 \succ [\cdots]$； |
| $u_j, \overline{u_j}$ for $j \in [m]$： | $y_j^1 \succ y_j^2 \succ y_j^3 \succ D_4 \succ p \succ [\cdots]$. |

考虑一个缩减选举 E。我们看到，候选人 $w$ 从 $V_0$ 中的每位选民处获得 $a_1$ 分，从选民 $w$ 处获得 $a_\ell$ 分；而由于虚拟候选人的存在，在其他地方均不得分。因此，我们得到 $\mathrm{scr}_{\mathcal{E}}(w) = (2m+1)a_1 + a_\ell$。候选人 $p$ 从 $V_0'$ 中的选民处获得 $(2m+1)a_1$ 分。由于每个文字在 $\varphi$ 中至多出现两次，$X_i$ 和 $\overline{X_i}$ 中的两个候选人从 $U \cup \overline{U}$ 中的选民处获得的分数都不超过 $4a_1$。来自 „ 和 „ 的虚拟候选人（仅当 $\ell \geq 2$ 时才存在）分别至多获得 $a_1$、$2ma_1$、$(2m+1)a_2$ 和 $(2m+1)a_2$ 分。这些数值均小于 $\mathrm{scr}_{\mathcal{E}}(w)$。

现在假设存在一个使 $\varphi$ 成立的满足赋值。让每个政党 $X_i$ 提名 $\{x_i, \overline{x_i}\}$ 中与真文字相对应的候选人。由于每个子句至少包含一个真文字，我们知道，对于每个 $j \in [m]$，$\{y_j^1, y_j^2, y_j^3\}$ 中至少有一名候选人获得提名。因此，候选人 $p$ 不会从 $U \cup \overline{U}$ 中的选民那里获得额外分数，并且在由此产生的选举中，她不是获胜者，因为 $\mathrm{scr}_{\mathcal{E}}(p) = (2m+1)a_1 < \mathrm{scr}_{\mathcal{E}}(w)$。

反过来，假设 $\varphi$ 不存在满足赋值。因此，无论政党 $X_i$、$\overline{X_i}$、$i \in [n]$ 如何提名，都至少存在一个子句 $C_j$，其中 $\{y_j^1, y_j^2, y_j^3\}$ 中没有任何文字被赋为真，这意味着选民 $u_j$ 和 $\overline{u_j}$ 都会给候选人 $p$ 分配 $a_\ell$ 额外分数。这保证了对于任何由此得到的缩减选举 E，只要 $\mathrm{scr}_{\mathcal{E}}(p) = (2m+1)a_1 + 2a_\ell$，$p$ 就是唯一获胜者；因此，她是 $I$ 中的必要总统。□

现在让我们把注意力转向否决型计分规则。定理 3.3 的证明也给出了一个从 (2,2)-E3-SAT 问题出发的归约；回想一下本节开头给出的 (2,2)-E3-SAT 定义之后的记号。

定理 3.3。设 $\mathcal{R}$ 是一个否决型计分投票规则，其基于的一个位置计分向量形式为 $(a, \ldots, a, a_1, a_2, \ldots, a_\ell)$，其中某个常数 $\ell \geq 1$ 满足 $a > a_\ell$。那么，即使最大政党规模为 $s = 2$，关于 $\mathcal{R}$ 的必要总统问题仍是 coNP-完全的。

证明。我们再次给出从 (2,2)-E3-SAT 出发的多项式归约。我们将构造一个必要总统问题实例 $I$，其中指定候选人 $p$ 属于政党 $P = \{p\}$，使得对于其他政党的每一种可能提名集合，$p$ 都是缩减选举的获胜者，当且仅当 $\varphi$ 是不可 满足的。

除 $P$ 外，我们还定义政党 $P_w = \{w\}$，并为每个变量 $x_i \in X$ 创建一个政党 $X_i = \{x_i, \bar{x}_i\}$。此外，我们创建一个包含 $\ell - 1$ 个虚拟候选人的集合 $D$，每个虚拟候选人各自组成一个单候选人政党。注意，政党数量为 $n + \ell + 1$，且任何政党的最大规模为 $s = 2$。

选民集合定义为 $V = \{v_0\} \cup W \cup \bigcup_{j\in[m]} V_j$ 其中 $W = \bigcup_{i\in[n]} W_i$，$|W_i| = 2$ 对每个 $i \in [n]$ 以及 $|V_j| = 2$ 对每个 $j \in [m]$ 均如此。因此，选民数量为 $2n + 2m + 1$，他们的偏好如下：

$$
\begin{array}{ll}
v_0: & [\cdots] \succ p \succ D; \\
w \in W_i \text{ for } i \in [n]: & [\cdots] \succ x_i \succ \bar{x}_i \succ D; \\
v \in V_j \text{ for } j \in [m]: & [\cdots] \succ w \succ y_j^1 \succ y_j^2 \succ y_j^3 \succ D.
\end{array}
$$

注意，$\mathrm{scr}_{\mathcal{E}}(p) = (2n + 2m)a + a_1$，并且由于政党 $X_i$ 的每位提名人从 $W_i$ 中的选民处仅获得 $a_1$ 分，因此 $\mathrm{scr}_{\mathcal{E}}(x_i)$ 以及 $\mathrm{scr}_{\mathcal{E}}(\bar{x}_i)$ 至多为 $(2n + 2m - 1)a + 2a_1$，严格小于 $\mathrm{scr}_{\mathcal{E}}(p)$，所以任何政党 $X_i$ 的提名人都不可能成为任何缩减选举的获胜者。此外，候选人 $w$ 从 $\{v_0\} \cup W$ 中的选民处获得 $(2n + 1)a$ 分。

现在假设存在一个满足 $\varphi$ 的真值赋值。让每个政党 $X_i$ 提名 $\{x_i, \bar{x}_i\}$ 中与真文字对应的候选人。由于每个子句都至少包含一个真文字，因此对于每个 $j \in [m]$，$\{y_j^1, y_j^2, y_j^3\}$ 中至少有一名候选人获得提名。因此，$\mathrm{scr}_{\mathcal{E}}(w) = (2n + 2m + 1)a > \mathrm{scr}_{\mathcal{E}}(p)$，所以在这些提名下，候选人 $w$ 是由此产生的缩减选举的获胜者。

反过来，假设 $\varphi$ 不存在满足赋值。因此，无论政党 $X_i$、$i \in [n]$ 提名什么，都至少存在一个子句 $C_j$，其中 $\{y_j^1, y_j^2, y_j^3\}$ 中没有任何文字被赋为真。因此，候选人 $w$ 从 $V_j$ 中的选民那里获得的分数少于 $a$，所以 $\mathrm{scr}_{\mathcal{E}}(w) \leq (2n + 2m - 1)a + 2a_1 < \mathrm{scr}_{\mathcal{E}}(p)$。因此，$p$ 是任何缩减选举中的唯一获胜者，所以她是 $I$ 中的必要总统。□

鉴于定理 3.2 和 3.3 中的难解性结果，我们研究短投票规则和类否决计分规则下必要总统问题的参数化复杂性。具体而言，我们将政党数量 $t$ 和选民类型数量 $\tau$ 作为参数。

*3.2.1* 按政党数量参数化。这里我们证明，对于短计分规则和类否决计分规则，必要总统在参数 $t$ 下都是 W[2]-困难的。我们的证明分别采用了与 [31], 中定理 1 和 6 的证明类似的思路。对于这两类计分规则，我们都给出了一个参数化归约，来自经典击中集问题，该问题在以所需击中集的大小为参数时是 W[2]-困难的（参见 Cygan 等人 [5]）。下面我们正式定义该问题：

**问题：** 击中集
**实例：** 一个元组 $H = (S, \mathrm{F}, k)$，其中 $S = \{s_1, s_2, \ldots\ldots, s_n\}$ 是元素集合，$\mathrm{F} = \{F_1, F_2, \ldots\ldots, F_m\} \subseteq 2^S$ 是 $S$ 的子集族，且 $k$ 是一个整数。
**问题：** 是否存在一个集合 $S' \subseteq S$ 使得 $|S'| \leq k$，并且对每个 $j \in [m]$ 都有 $S' \cap F_j \neq \emptyset$？

定理 3.4。设 $\mathcal{R}$ 是一个短投票规则，其基于如下形式的位置计分向量：$(a_1, a_2, \ldots, a_\ell, 0, \ldots, 0)$，其中 $\ell \geq 1$ 为某个常数，且满足 $a_\ell > 0$。那么，当以政党数量 $t$ 为参数时，$\mathcal{R}$ 下的必要总统问题是 W[2]-难的。

证明。我们给出一个从击中集问题出发的参数化归约。

对于击中集的任意实例 $H = (S, \mathrm{F}, k)$，我们将构造一个必要总统问题的实例 $I$，其中指定候选人 $p$ 属于政党 $P = \{p\}$，使得对于其他政党的任意可能提名，$p$ 都是缩减选举的获胜者，当且仅当 $H$ 是击中集问题的“否”实例。

除了 $P$，我们还定义政党 $P_w = \{w\}$，以及对每个 $i \in [k]$ 定义政党 $P_i = \{s_1^i, \ldots\ldots, s_n^i\}$，其中候选人 $s_r^i$ 代表元素 $s_r \in S$ 的第 $i^{\text{th}}$ 个副本，且对某个 $r \in [n]$ 成立。我们还用记号 $F_j^i = \{s^i : s \in F_j\}$ 表示包含在 $F_j, j \in [m]$ 中的 $D_1 \cup D_2 \cup D_3 \cup D_4$ 的元素的 $i^{\text{th}}$ 个副本所构成的集合。此外，我们还定义一组虚拟候选人，其中 $|D_h| = \ell - 1$ 分别对应 $h = 1, 2, 3, 4$，每个虚拟候选人各自组成一个仅有一名候选人的政党。因此，政党总数为 $t = k + 4\ell - 2$。

选民集合为 $V = W \cup U \cup U' \cup V_0 \cup V_0'$，$|V_0| = \{\}$，$|V_0'| = 2m + 1$，$U = u_1 \ldots u_m$，$U' = u_1' \ldots u_m'$，其中 ，和 {,}，{,}。因此，共有 $6m + 3$ 名选民。偏好剖面如下所示。

$$
\begin{array}{ll}
w: & D_1 \succ w \succ [\cdots]; \\
u_j, u_j' \text{ for } j \in [m]: & F_j^1 \succ F_j^2 \succ \cdots \succ F_j^k \succ D_2 \succ p \succ [\cdots]; \\
v \in V_0: & w \succ D_3 \succ [\cdots]; \\
v' \in V_0': & p \succ D_4 \succ [\cdots].
\end{array}
$$

回顾一下，$\ell$ 是一个常数，因此政党数量仅取决于 $k$，所以这里给出的归约是一个参数化归约。它同时也是多项式时间归约，因此根据击中集的 NP-困难性，我们的证明不仅得到参数为 $t$ 时的 $W[2]$-困难性，还得到 NP-困难性。

考虑一个缩减选举 E。我们看到，候选人 $w$ 从 $V_0$ 中的每位选民处获得 $a_1$ 分，从选民 $w$ 处获得 $a_\ell$ 分；此外，由于虚拟候选人的存在，她在其他地方均不得分。因此，我们得到 $\mathrm{scr}_{\mathcal{E}}(w) = (2m + 1)a_1 + a_\ell$。对于候选人 $p$，我们有 $\mathrm{scr}_{\mathcal{E}}(p) = (2m + 1)a_1$；她的所有得分都来自 $V_0'$ 中的选民。每位候选人 $s_r^i$ 从 $U \cup U'$ 中的选民处获得的分数至多为 $2ma_1$。来自 $D_1, D_2, D_3$ 和 $D_4$ 的虚拟候选人（仅当 $\ell \geq 2$ 时才存在）分别至多获得 $a_1$、$2ma_1$、$(2m + 1)a_2$ 和 $(2m + 1)a_2$ 分。这些数值都小于 $\mathrm{scr}_{\mathcal{E}}(w)$。

现在假设，对于实例 $H$ 存在一个击中集 $S' = \{s_{(1)}, \ldots\ldots, s_{(k)}\}$，其中 $s_{(i)}$ 表示 $S'$ 中按某一固定顺序的第 $i^{\text{th}}$ 个元素。令每个政党 $P_i$ 提名对应的候选人

即元素 $s_{(i)} \in S'$ 的 $i^{\text{th}}$ 副本所对应者，也就是候选人 $s^i$ $(i)$ 。由于对每个 $j \in [m]$都有$S' \cap F_j \neq \emptyset$，这确保集合 $F_j^1 \cup F_j^2 \cdots \cup F_j^k$中至少有一名候选人被提名。因此，候选人 $p$不会从 $U \cup U$,中的选民那里获得额外分数，并且她不属于由此得到的缩减选举的获胜者。

反过来，假设 $H$不存在大小为 $k$的击中集。因此，对于每个大小至多为 $k$的子集 $S' \subseteq S$，都存在至少一个集合 $F_j \in \mathrm{F}$，使得 $S' \cap F_j = \emptyset$。这又意味着，对于政党 $P_i, i \in [k]$的任意提名，都存在至少一个索引 $j$，使得没有候选人被提名到 $F_j^1 \cup F_j^2 \cup \cdots \cup F_j^k$中，因此，选民 $u_j$和 $u_j'$都会向候选人 $p$分配 $a_\ell$额外分数，从而确保 $p$是所得缩减选举 E 中的唯一获胜者，且 $\mathrm{scr}_{\mathcal{E}}(p) = (2m+1)a_1 + 2a_\ell$。

因此，当且仅当 $H$是击中集的否实例时，$p$才是任意缩减选举的获胜者，也就是 $I$中的必要获胜者。 □

接下来，我们给出定理 3.4 在否决型投票规则下的对应版本。

定理 3.5。 设$\mathcal{R}$是一个类否决投票规则，其基于如下形式的计分向量：$(a, \ldots, a, a_1, a_2, \ldots, a_\ell)$，其中某个常数 $\ell \geq 1$满足 $a > a_1$。那么，以政党数量为参数的$\mathcal{R}$的必要总统问题是W[2]-难的。

证明。我们再次给出一个从击中集归约的参数化归约。设 $H$是击中集的一个实例，我们构造必要总统问题的一个实例 $I$。

同样，$p$是我们的指定候选人，包含在单元素政党 $P = \{p\}$中；对每个 $i \in [k]$，我们定义政党 $P_w = \{w\}$和 $P_i = \{s_1^i, \ldots, s_n^i\}$。对于每个 $j \in [m]$，我们用$F_j^i = \{s^i\colon s \in F_j\}$表示包含在 $F_j$中的元素的 $i^{\text{th}}$ 个副本所构成的集合。此外，我们创建一个由 $\ell - 1$ 个虚拟候选人组成的集合 $D$，每个虚拟候选人各自构成一个单候选人政党。因此，政党数量为 $t = k + \ell + 1$。由于 $\ell$是一个常数，所给出的归约是一个参数化归约，因此政党数量仅取决于 $k$。

选民集合为

$$V = \{v_0\} \cup W \cup V'$$

其中，对于每个 $i \in [k]$ 和每个 $j \in [m]$，W=$\bigcup_{i\in[m]} W_i$, $V$, $= \bigcup_{j\in[k]} V_j$, $|W_i| = 2$。因此，选民数量为 $2k + 2m + 1$。他们的偏好如下：

$$\begin{aligned}
&v_0 : && [\cdots] \succ p \succ D;\\
&w \in W_i \text{ for } i \in [k] : && [\cdots] \succ s_1^i \succ s_2^i \succ \cdots \succ s_n^i \succ D;\\
&v \in V_j \text{ for } j \in [m] : && [\cdots] \succ w \succ F_j^1 \succ F_j^2 \succ \cdots \succ F_j^k \succ D.
\end{aligned}$$

很容易看出，对于任意缩减选举 E，我们都有

$\mathrm{scr}_{\mathcal{E}}(p) = (2k + 2m)a + a_1$；对于任意 $d \in D$，$\mathrm{scr}_{\mathcal{E}}(d) \leq (2k + 2m + 1)a_2$；对于任意 $s_r^i$，$\mathrm{scr}_{\mathcal{E}}(s_r^i) \leq (2m + 1)a + 2ka_1$；$i \in [k]$，$r \in [n]$,

这意味着，对于每个虚拟候选人 $d \in D$，$\mathrm{scr}_{\mathcal{E}}(d) < scr(p)$；对于政党 $P_i$的每个提名人 $i \in [k]$，$\mathrm{scr}_{\mathcal{E}}(p_i) < scr(p)$。候选人 $w$ 从 $\{v_0\}$ 中的选民处获得 $(2k + 1)a$ 分。

现在假设实例 $H$ 存在一个击中集 $S' = \{s_{(1)}, s_{(2)}, \ldots\ldots, s_{(k)}\} \subseteq S$。令政党 $P_i$ 对每个 $i \in [k]$ 提名其成员 $s^i{}_{(i)}$。那么候选人 $w$ 会出现在 $V'$ 中每位选民的偏好列表里，且其后至少有 $\ell$ 位其他候选人，因此她的得分变为 $\mathrm{scr}_{\mathcal{E}}(w) = (2k + 2m + 1)a$，因此，在这些提名下，她是 E 的唯一获胜者。因此，$p$ 不是必要总统。

反之，假设 $H$不存在击中集。则至少存在一个索引 $j \in [m]$满足 $S' \cap F_j = \emptyset$。这意味着，对于来自政党 $P_i$的任何可能提名，都存在某个$j \in [m]$，使得在 $V_j$中选民的偏好里，虚拟候选人紧接在候选人 $w$之后，因此$\mathrm{scr}_{\mathcal{E}}(w) \leq (2k + 1)a + (2m - 2)a + a_1$。该值严格小于$\mathrm{scr}_{\mathcal{E}}(p)$，因此候选人 $p$是 $I$中的必要总统。□

*3.2.2* 按选民类型数量进行参数化。接下来，我们证明：当以选民类型的数量 $\tau$作为参数时，必要总统问题对于短位置计分规则和否决型位置计分规则都是固定参数可处理的。这与我们按政党数量或最大规模进行参数化时所得到的难解性结果形成鲜明对比，如定理3.2–3.5所示。

定理 3.6。 设 $\mathcal{R}$是一个短投票规则，其依据的计分向量形式为 $(a_1, a_2, \ldots, a_\ell, 0, \ldots, 0)$，其中 $\ell \geq 1$ 满足 $a_\ell > 0$。那么，当以选民类型的数量 $\tau$ 为参数进行参数化时，规则 $\mathcal{R}$的必要总统属于 固定参数可 tractable 。

证明。我们给出 NP-Short 算法，以固定参数可 tractable 时间求解规则 R 的必要总统的一个实例$I = (\mathrm{E}, \mathrm{P}, p)$，参数为 $\tau$，即 E 中不同选民 $V = V_1 \cup \cdots \cup V_\tau$ 类型的数量。设 $\cdots$ 是按照选民类型对选民进行的划分；也就是说，对于某个 $i \in [\tau]$，$V_i$中的所有选民对 $C$都有相同的偏好。

假设存在一组 $C' \ni p$ 提名候选人，使得在关于 $C'$ 的缩减选举 $\mathrm{E}_{C'}$ 中，$p$ 并非获胜者。设 $P$ 是包含 $p$ 的政党。NP-Short 算法猜测关于 $\mathrm{E}_{C'}$ 的以下信息：

- 一个候选人 $w \in C'$在 $\mathrm{E}_{C'}$ 中的得分高于 $p$。选择 $w$有 $|C \setminus P|$种可能。
- $\mathrm{E}_{C'}$的结构定义如下。首先定义函数 prt: $[\tau] \times [\ell] \to \mathrm{P}$，如下所示。对于某个选民类型 $i \in [\tau]$ 和索引 $j \in [\ell]$，令 $\mathrm{prt}(i, j)$ 表示在缩减选举 $\mathrm{E}_{C'}$ 中，来自 $V_i$ 的选票中位于第 $j$位的候选人所属的政党。对于 $[\tau] \times [\ell]$ 中的某些有序对 $(i, j)$ 和 $(i' j')$，若 $\mathrm{prt}(i, j) = \mathrm{prt}(i' j')$，则记作 $(i, j) \sim (i' j')$。$\mathrm{E}_{C'}$的结构是关系 ∼的等价类族 Q，它是 $[\tau] \times [\ell]$的一个划分。注意，选择 Q 至多有 $(\tau\ell)^{\tau\ell}$ 种可能。
- 等价类 $Q_w \in \mathrm{Q}$ 包含所有满足 $\mathrm{prt}(i, j) = P_w$ 的有序对 $(i, j)$。选择 $Q_w$ 有 $|\mathrm{Q}| \leq \tau\ell$ 种可能。
- 集合 $Q_p$ 包含所有满足 $\mathrm{prt}(i, j) = P$ 的有序对 $(i, j)$。注意，$Q_p \in \mathrm{Q}$ 或 $Q_p = \emptyset$。因此，选择 $Q_p$ 有 $|\mathrm{Q} \setminus \{Q_w\}| + 1 \leq \tau\ell$ 种可能。

在猜测完上述关于 $\mathrm{E}_{C'}$ 的信息后（这里“猜测”是指尝试所有可能性），NP-Short 算法接下来按如下方式构造一个辅助二分图 $G$。$G$的顶点集为 $\hat{}\mathrm{P} \cup \hat{}\mathrm{Q}$，其中 $\hat{}\mathrm{P} = \mathrm{P} \setminus \{P, P_w\}$，$\hat{}\mathrm{Q} = \mathrm{Q} \setminus \{Q_p, Q_w\}$。对于每个 $Q \in \mathrm{Q}$，定义

$$Q^+ = Q \cup \{(i, \ell + 1) : i \in [\tau], \nexists j \in [\ell] \text{ such that } (i, j) \in Q\}.$$

直观而言，$Q^+$ 表示这样一种情形：某政党的提名人 (i) 在来自 $V_i$ 的选民对 $\mathrm{E}_{C'}$ 中 $j^{\text{th}}$ 所投选票的排序中获得

对于每个 $(i,\ j) \in Q$，其中 $j \in [\ell]$，以及 (ii) 在来自 $V_i$ 的选民对 $\mathrm{E}_{C'}$ 中 $\ell^{\text{th}}$ 所投选票的排序中排在其后，对于每个 $(i,\ \ell+1) \in Q^+$。

候选人 $c \in C \setminus (P \cup P_w)$ 相对于放置得当（关于 $p$ˆ和 $w$）对于 $Q \in \mathrm{Q}$，当每个 $i \in [\tau]$ 都满足以下条件时成立：

(i) 如果 $(i,\ j) \in Q^+$ 和 $(i,\ j') \in Q_p^+$ 对于某些 $j$ 和 $j'$ 成立，那么 $j = j' = \ell + 1$ 或 $c$ 在来自 $V_i$ 的选民偏好中排在 $p$ 之前，当且仅当 $j < j'$；(ii) 如果 $(i,\ j) \in Q^+$ 和 $(i,\ j'') \in Q_w^+$ 对于某些 $j$ 和 $j''$成立，那么 $j = j'' = \ell + 1$ 或 $c$ 在来自 $V_i$ 的选民偏好中排在 $w$ 之前，当且仅当 $j < j''$。

~ ˆ 现在我们可以定义辅助图 $G$：某个政党 $P \in \mathrm{P}$ 与某个 $Q \in \mathrm{Q}$ 在 $G$ 中由一条边相连，当且仅当~存在一个候选人 $c \in P$，其相对于 $p$ 和 $w$对于 $Q$放置得当。给定候选人 $c$ 和某个 $Q \in$ ˆQ，可以在$O(\tau\ell)$ 时间内检查 $c$对于 $Q$是否放置得当，因此可以在 $O(|C|\tau^2\ell^2)$ 时间内计算出图 $G$。

计算图 $G$后，NP-Short 算法还会计算集合 $\mathrm{S} \subseteq \mathrm{P}$，其中包含安全政党，即至少包含一名安全候选人的政党 $S$。所谓安全候选人，是指对于每个 $i \in [\tau]$，如果 $(i, \ell+1) \notin Q_w^+$，则 $V_i$ 中的选民更偏好 $w$而非 $c_S$；我们为每个安全政党 $S$选定这样一名候选人 $c_S$。直观而言，这些候选人可以安全地获得提名，因为他们不会阻碍 $w$ 在 E 中获得所需的分数。

接下来，NP-Short 算法会计算 $M$中的一个匹配 $G$，使其覆盖 $\mathrm{P} \setminus \mathrm{S}$ 中所有非安全政党以及 Q 中所有等价类；如果不存在这样的匹配，它就会舍弃当前的猜测集合。~ˆ ~每个由边 $(P,\ Q) \in M$ 覆盖的政党 $P \in \mathrm{P}$，都会提名一名对 $Q$放置得当的候选人；而每个未被 $M$ 中的边覆盖的政党 $S$（这必定是安全政党）都会提名候选人 $c_S$。最后，算法会检查这些提名与 $p$ 和 $w$ 是否共同构成一个缩减选举 $\mathrm{E}^\star$，使得 $p$ 不是获胜者；如果是，则返回“no”。如果算法遍历了所有可能的猜测，但仍未返回“no”，则返回“yes”。

注意，选择 $w$、Q $Q_p$ 和 $Q_w$，最多有 $|C| \cdot (\tau\ell)^{\tau\ell+2}$ 种可能性。一旦这些猜测确定下来，就可以在 $O(|C|\tau^2\ell^2)$ 时间内计算出二分辅助图 $G$，正如我们此前所论证的那样。每次迭代的瓶颈是计算算法 NP-Short 第 12 行中的匹配。由于 $G$ 至多有 $|\mathrm{P}| + \tau\ell \le |C| + \tau\ell$ 个顶点，我们可以通过 Hopcroft–Karp 算法 [17]，在 $O((|C| + \tau\ell)^{2.5})$ 时间内计算出 $G$ 中覆盖所需顶点集的匹配。因此，总运行时间为 $O(|C|^{3.5} \cdot (\tau\ell)^{\tau\ell+4.5})$，关于 $\tau$ 固定参数可处理，因为 $\ell$ 是常数。

下面证明算法 NP-Short 的正确性。显然，只要算法 NP-Short 返回“否”，其判断就是正确的，因为对于一个包含 $p$，并且对 $\mathrm{P} \setminus \{P\}$ 中的每个政党恰好包含一名候选人的提名候选人集合 $C_M$，候选人 $p$ 不是 $C_M$ 上缩减选举的获胜者，因此也不是必要总统。

因此，接下来只需证明：当输入是必要总统问题的一个“否”实例时，NP-Short 算法会返回“否”。令 $C'$ 为缩减选举 $\mathrm{E}_{C'}$ 中的提名人集合，其中 $P$ 不是获胜者。于是存在某个候选人 $w$，使得 $\mathrm{scr}_{\mathcal{E}}(w) > \mathrm{scr}_{\mathcal{E}}(p)$。再令 $P_w \in \mathrm{P}$ 包含 $w$。令 Q 为

**NP-Short 算法**求解必要总统的简写

投票规则。

**输入：** 一个关于候选人、P $p$的必要总统实例 (E 集合 $C$ 和 $p \in P \in$ P。

1: **对所有** $w \in C \setminus P$ **执行**
2: **对所有** ▹ Q 的划分 $[\tau] \times [\ell]$ **执行**： E 的结构
3: **对所有** $Q_w \in$ Q 和 $Q_p \in (\mathrm{Q} \setminus \{Q_w\}) \cup \{\emptyset\}$ **执行**
4: 令 $P_w$ 为 P 类中包含 $w$的政党。
5: 令 ˆP =P 类 $\setminus \{P, P_w\}$ 和 ˆQ =Q $\setminus \{Q_p, Q_w\}$。
6: 令 $E = \emptyset$ 和 S $= \emptyset$。
7: **对所有** $c \in P_c \in \hat{\mathrm{P}}$ 类和 $Q \in \hat{\mathcal{Q}}$ **do**
8: **如果** $c$对 $Q$放置得当，**则**将 $(P_c,\ Q)$ 添加到 $E$。
9: 创建图 $G = (\hat{\mathrm{P}} \cup \hat{\mathrm{Q}}\ E)$。
10: **对于所有** $S \in \hat{\mathcal{P}}$ **do**
11: **如果** ∃ 是安全候选 $c_S \in S$，**则**将 $S$添加到 S。
12: **如果** ∃ 一个匹配 $M$ 在 $G$ 覆盖 $\hat{\mathrm{Q}} \cup \hat{\mathrm{P}} \setminus \mathrm{S})$ **则**
13: 集合 $C_M = \{p,\ w\}$。
14: **对于所有**（ $\widetilde{P},\ Q) \in M$ **执行**
15: 将 $C_M$中一个对于 $Q$放置得当的候选人添加到 $\widetilde{P}$。
16: **对于所有** $S \in$S 未被 $M$ 覆盖的情况**执行** 将 $c_S$ 添加到 $C_M$。
17: **如果** $p$不是 $\mathrm{E}_C$ 中的获胜者 $_M$ **那么返回**“否”。
18: **返回**“是”。

E 的结构，并令 $Q_w \in$Q 和 $Q_p \in$Q $\cup \{\emptyset\}$按照前文定义。考虑算法在第 1–3 行选择候选人 $w$、结构 Q 以及集合 $Q_p$ 和 $Q_w$的这次迭代 $\iota$。

ˆ 注意，对于每个 $(i,\ j) \in Q$，其中某个 $Q \in$ Q 满足该条件，在 $\mathrm{E}_{C'}$ 中，来自 $V_i$ 的选票中位于第 $j^{\text{th}}$ 位的候选人 $c$ 对于 $Q$ 放置得当（这是由 $\mathrm{E}_{C'}$ 的结构 Q 的定义所决定的），因此，提名 $c$ 的政党与 $Q$ 之间在辅助图 $G$ 中由一条边相连。令 $M^\star$ 表示所有此类边构成的集合；那么 $M^\star$ 是 $G$ 中的一个匹配，它覆盖了 ˆQ 中的每个等价类。此外，注意，在 E 中总得分为 0 的每个政党都有一位在 $\mathrm{E}_{C'}$ 中安全的提名人。因此，匹配 $M^\star$ 覆盖了所有非安全政党。由此，NP-Short 算法将在第 12 行发现，存在某个匹配 $M$，覆盖 ˆQ $\cup$ ( ˆP \S)。

现在我们证明，在第 13–16 行的 NP-Short 算法计算出的候选人集合 $C_M$ 上的简化选举 $\mathrm{E}_{C_M}$ 中，$p$ 不可能是获胜者。为此，先考虑某个 $i \in [\tau]$，在 $\mathrm{E}_{C'}$ 中，$p$ 因 $V_i$ 中的每位选民而获得 $a_j > 0$ 分。令 $Q_1$、…、$Q_{j-1} \in$Q 为分别包含数对 $(i, 1)$ … $(i,\ j-1)$ 的等价类。首先，若对某个 $h \in [j-1]$ 有 $Q_h = Q_w$，则根据 $Q_p$、$Q_w$ 的定义以及第 $\iota$ 次迭代中的猜测，$w$ 在 $V_i$ 中选民的偏好排序中位于 $p$ 之前。此外，对每个 $h \in [j-1]$，匹配 $M$ 都包含一条覆盖 $Q_h$ 的边 $(P_h,\ Q_h)$，因此 $C_M$ 包含一个由 $P_h$ 提名、且对 $Q_h$ 放置得当的候选人 $c_h$。根据定义，这意味着在 $\mathrm{E}_{C_M}$ 中，$c_h$ 位于 $p$ 之前。由于这对每个 $h \in [j-1]$ 都成立，因此我们得到 $\mathrm{scr}_{\mathcal{E}_{C_M}}(p) \le \mathrm{scr}_{\mathcal{E}_{C'}}(p)$。

类似地，考虑某个 $i \in [\tau]$，它在 $\mathrm{E}_{C'}$ 中因 $V_i$ 的每位选民而获得 $w$ $a_j > 0$ 分。令 $Q_1, \ldots, Q_{j-1} \in$Q 为分别包含这些有序对 $(i,1)\ldots,(i, j-1)$ 的等价类。一方面，如果某位候选人 $c$ 被 $\mathrm{E}_{C'}$ 中某个稳固政党提名

，那么 $c$ 是安全的，也就是说，$V_i$ 中的选民更偏好 $w$ 而非 $c$（因为 $(i, j) \in Q_w$）。另一方面，如果 $c$ 是~ ~由某个政党 $P$ 提名，且该政党被 $M$ 中某个属于 $Q \in$^Q 的边 $(P, Q)$ 覆盖，那么，要么 $Q \in \{Q_1, \ldots, Q_{j-1}\}$，要么在 $V_i$ 中选民的偏好排序里，$c$ 不排在 $w$ 之前，因为对于 $Q$ 而言，$c$ 放置得当。这意味着在 $\mathrm{E}^{C_M}$ 中，排在 $w$ 之前的提名人至多有 $j-1$ 位，因此 $\mathrm{scr}^{\mathcal{E}_{C_M}}(w) \geq \mathrm{scr}^{\mathcal{E}_{C'}}(w)$。

根据前面两段，我们得到

$$\mathrm{scr}_{\mathcal{E}_{C_M}}(p) \leq \mathrm{scr}_{\mathcal{E}_{C'}}(p) < \mathrm{scr}_{\mathcal{E}_{C'}}(w) \leq \mathrm{scr}_{\mathcal{E}_{C_M}}(w),$$

因此，正如所保证的，$p$ 不是 $\mathrm{E}^{C_M}$中的获胜者。因此，NP-Short 算法在第 $\iota$次迭代的第 17 行返回“否”，从而证明了该算法的正确性。

□

接下来，我们给出一种算法：当以 $\tau$为参数时，它能在固定参数可处理的时间内求解类否决计分规则下的必要总统问题。

定理3.7. 设$\mathcal{R}$ 为一种类否决投票规则，其计分向量的形式为$(a, \ldots, a, a_1, a_2, \ldots, a_\ell)$，其中 $\ell \geq 1$ 为某个常数，且 $a > a_1$。那么，当以选民类型的数量 $\tau$为参数时，必要总统问题对于$\mathcal{R}$ 属于固定参数可 tractable 。

证明。设 $(\mathrm{E}, \mathrm{P}\ p)$ 是必要总统问题的输入实例。首先，若 $|\mathrm{P}| \leq \ell\tau$，则我们将 R 视为针对 $\ell' = |\mathrm{P}| \leq \tau\ell$ 的 $\ell'$-短投票规则，并应用 NP-Short 算法。如定理 3.6 的证明所示，运行时间为 $O(|C|^{3.5} \cdot (\tau^2\ell)^{\tau^2\ell+4.5})$；由于 $\ell$ 是常数，因此该时间对于 $\tau$ 是固定参数可处理的。

其次，如果 $|\mathrm{P}| > \ell\tau$，那么在每个缩减选举 $\mathrm{E}'$ 中，至少有一位被提名候选人能够获得 $|V| \cdot a$ 的最高可能得分。这意味着，当且仅当 $p$ 的得分低于 $|V| \cdot a$ 时，$p$ 才不是 $\mathrm{E}'$ 中的获胜者；也就是说，对于某位选民而言，它排在最不受欢迎的被提名人之列。要判断这种情况是否可能，只需检查

$$|\{\widetilde{P} \in \mathcal{P} : c \succ_v p \text{ for some } c \in \widetilde{P}\}| \geq |\mathcal{P}| - \ell \qquad (3)$$

对于某位选民 $v \in V$ 是否成立。不难看出，这可以在多项式时间内完成检查。如果对于某个 $v \in V$，不等式 (3) 成立，则返回“否”；否则返回“是”。

为说明该算法的正确性，假设对于某位选民 $v \in V$，不等式 (3) 成立。那么，提名 $p$，并在其他所有可能的政党中提名一位比 $p$ 更受偏好的候选人，由此可知，在最终选举 $\mathrm{E}'$ 中，按照 $v$ 的偏好，$p$ 前面至少有 $|\mathrm{P}| - \ell$ 位提名人。因此，$v$ 给 $p$ 分配的得分少于 $a$ 分，得到 $\mathrm{scr}_{\mathcal{E}'}(p) < |V| \cdot a$，进而意味着 $p$ 不是 $\mathrm{E}'$ 中的获胜者。现在假设对于任何选民 $v \in V$，不等式 (3) 都不成立。那么，无论各政党的提名如何，对于每位选民 $v \in V$，其偏好中排在 $p$ 前面的提名人所属政党都少于 $|\mathrm{P}| - \ell$ 个。因此，$p$ 从每位选民处获得 $a$ 分，总得分为 $|V| \cdot a$，这是可获得的最高得分，从而确保 $p$ 是最终选举中的获胜者。□

## 4 孔多塞一致投票规则的结果

本节讨论三种孔多塞一致投票规则。我们证明，科普兰$^\alpha$ 和最大最小法的必要总统问题均可在多项式时间内求解。相比之下，必要排序对的必要主席问题即使在选民数量为常数时，计算上仍然是困难的。

定理4.1. 对于每个 $\alpha \in [0, 1]$，科普兰必要总统问题$^\alpha$ 可在多项式时间内求解。

证明。我们提出 NP-Copeland 算法，用于在多项式时间内求解科普兰$^\alpha$ 投票中的必要总统问题。设 $(\mathrm{E}, \mathrm{P}\ p)$ 为我们的输入实例，其中选举为 $\mathrm{E} = (C, V, \{\succ_v\}_{v \in V})$，政党 $P$ 包含 $p$。

**NP-Copeland 算法** 求解 必要总统问题的
对于某个 $\alpha \in [0, 1]$ 的科普兰$^\alpha$。

**输入：** 一个必要总统问题的实例 $(\mathrm{E}, \mathrm{P}\ p)$，其中包含候选人集合 $C$ 和 $p \in P \in \mathrm{P}$ 类。

1: **对于所有** $p' \in C \setminus P$，**执行**
2: 令 $P'$ 为 P 类中包含 $p'$的政党。
3: **对于所有** $c \in C \setminus (P \cup P')$，**执行** 计算 $\Delta_c(p', p)$。
4: 设 $\widetilde{C} = \{p, p'\}$。
5: **对于所有** $\widetilde{P} \in$ P 类$\setminus \{P, P'\}$ **确实**
6: 向 $\widetilde{C}$ 添加候选人 $c(\widetilde{P}) \in \arg\max_{c \in \widetilde{P}} \Delta_c(p'p)$。
7: **如果** $p$ 不是 $\mathrm{E}^{\widetilde{C}}$ 中的获胜者，**则返回**“否”。
8：**返回**“是”。

假设存在一个 $C' \ni p$ 提名候选人集合，使得 $p$ 在关于 $C'$ 的约化选举 $\mathrm{E}_{C'}$ 中不是获胜者。NP-Copeland 算法首先猜测一个候选人 $w \in C' \setminus P$，使得 $\mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(w) > \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(p)$。令 $P_w$ 为包含 $w$ 的政党。

回忆一下，对于两个候选人 $c$ 和 $c'$，$\mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(c, c')$ 是 $c$ 在 $\mathrm{E}_{C'}$ 中与 $c'$ 进行一对一比较后获得的得分，定义见公式（1）。现在定义$\Delta(w, p) = \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(w, p) - \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(p, w)$；注意，对于所有同时包含 $p$和 $w$ 的约化选举 E，此值都相同，而且可以根据选举 E 轻松计算得出。于是，$\mathrm{E}_{C'}$ 中 $w$与 $p$ 的科普兰得分之差可表示为

$$\mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(w) - \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(p) = \Delta(w, p) + \sum_{c \in C' \setminus \{p, w\}} \Delta_c(w, p) \qquad (4)$$

其中，对于在 E 中获得提名的每个候选人 $c \in C' \setminus \{p, w\}$，$\Delta_c(w, p) = \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(w, c) - \mathrm{Cpl}^\alpha_{\mathcal{E}_{C'}}(p, c)$。因此，$\Delta_c(w, p)$ 的值反映了 $w$ 和 $p$ 与某个候选人 $c \in C' \setminus \{p, w\}$ 比较后得分的差异。请注意，对于所有包含 $p$、$w$ 和 $c$ 的 E 的约化选举，$\Delta_c(w, p)$ 都相同，并且可根据以下情形计算：

$(i)$ 若 $w$击败 $c$ 且 $c$击败 $p$，则 $\Delta_c(w, p) = 1$；$(ii)$ 若 $w$击败 $c$ 且 $p$与 $c$平票，则 $\Delta_c(w, p) = 1 - \alpha$；$(iii)$ 若 $w$与 $c$平票且 $p$被 $c$击败，则 $\Delta_c(w, p) = \alpha$；$(iv)$ 若 $w$与 $c$平票且 $p$击败 $c$，则 $\Delta_c(w, p) = \alpha - 1$；$(v)$ 若 $w$被 $c$击败且 $p$与 $c$平票，则 $\Delta_c(w, p) = -\alpha$；$(vi)$ 若 $w$被 $c$击败且 $p$击败 $c$，则 $\Delta_c(w, p) = -1$；$(vii)$ 若 $p$ 和 $w$ 均击败 $c$，或均与 $c$ 平票，或均被 $c$ 击败，则 $\Delta_c(w, p) = 0$。

猜测 $w$后，NP-Copeland 算法会为每个政党 $P \in \mathrm{P} \setminus \{P, P_w\}$ 提名一名候选人~ ~date $c_{\widetilde{P}} \in P$，使 $\Delta_c(w, p)$ 在 $\widetilde{P}$ 上最大化。最后，算法计算 $\mathrm{Cpl}^\alpha_{\mathcal{E}_{\widetilde{C}}}(w) - \mathrm{Cpl}^\alpha \mathrm{E}_{\widetilde{C}} f(p)$

根据公式（4）计算所得缩减选举 $\mathrm{E}_{\widehat{C}}$；如果该值为正，则返回“否”。如果算法已探索 $w$ 的所有可能猜测而仍未返回“否”，则返回“是”。

下面证明 NP-Copeland 算法的正确性。显然，每当 NP-Copeland 算法返回“否”时，它的判断都是正确的，因为 $\mathrm{Cpl}^{\alpha}_{\mathcal{E}_{\widehat{C}}}(w) > \mathrm{Cpl}^{\alpha}_{\mathcal{E}_{\widehat{C}}}(p)$，因此候选人 $p$ 不是该算法在候选人集合 $\widehat{C}$ 上构造的缩减选举 $\mathrm{E}_{\widehat{C}}$ 的获胜者。

反过来，假设候选人 $w$ 是某个缩减选举 $\mathrm{E}_{C'}$ 的获胜者，该选举的提名候选人集合为 $C' \ni p$，且其中 $p$ 不是获胜者。设 $P_w$ 是包含 $w$ 的政党，并设 $c_{\widetilde{P}}$ 表示 $\mathrm{E}_{C'}$ 中某个不同于 $P$ 或 $P_w$ 的政党 $\widetilde{P}$ 的提名人。由于 $w$ 是 $\mathrm{E}_{C'}$ 中的获胜者，而 $p$ 不是，因此有 $\mathrm{Cpl}^{\alpha}_{\mathcal{E}_{C'}}(w) > \mathrm{Cpl}^{\alpha}_{\mathcal{E}_{C'}}(p)$。利用公式（4）以及算法对提名候选人的选择，可得

$$\mathrm{Cpl}^{\alpha}_{\mathcal{E}_{\widehat{C}}}(w) - \mathrm{Cpl}^{\alpha}_{\mathcal{E}_{\widehat{C}}}(p) = \Delta(w,p) + \sum_{\widetilde{P}\in\widetilde{\mathcal{P}}} \max\{\Delta_c(w,p) : c \in \widetilde{P}\}$$
$$\geq \Delta(w,p) + \sum_{\widetilde{P}\in\widetilde{\mathcal{P}}} \Delta_{c_{\widetilde{P}}}(w,p) = \mathrm{Cpl}^{\alpha}_{\mathcal{E}_{C'}}(w) - \mathrm{Cpl}^{\alpha}_{\mathcal{E}_{C'}}(p) > 0$$

对于 $\widetilde{\mathrm{P}} = \mathrm{P} \setminus \{P, P_w\}$。因此，NP-Copeland 算法返回“否”。

为评估 NP-Copeland 算法的计算复杂性，注意到至多有 $|C|$ 种选择 $w$ 的可能性。可以在 $O(|V|\cdot|C|)$ 时间内计算值 $\Delta_c(w,p)$，这也足以找出提名候选人，并计算 $p$ 和 $w$ 最终的科普兰$^{\alpha}$得分。因此，NP-Copeland 算法的总运行时间为 $O(|C|^2\cdot|V|)$。□

定理 4.2。必要总统 的最大最小法问题可在多项式时间内求解。

证明。我们提出一种可在多项式时间内求解最大最小法下必要总统问题的算法；其伪代码见 NP-Maximin 算法。设 $(\mathrm{E}, \mathrm{P}, p)$ 是我们的输入实例，其中选举 $\mathrm{E} = (C, V, \{\succ_v\}_{v\in V})$，政党 $P$ 包含 $p$。

假设存在一个缩减选举 $\mathrm{E}'$，其中 $p$ 被提名但不是获胜者。那么，$\mathrm{E}'$ 中存在一位提名人 $w$，其 Maximin 分数高于 $p$ 的 Maximin 分数。令 $P_w$ 为包含 $w$ 的政党。NP-Maximin 算法首先猜测候选人 $w$，以及一位候选人 $\hat{c} \in C \setminus (P \cup P_w)$，后者决定了 $p$ 在 $\mathrm{E}'$ 中的 Maximin 分数，即满足 $\mathrm{MM}_{\mathcal{E}'}(p) = N(p, \hat{c})$ 的候选人。令 $\hat{s} = N(p, \hat{c})$，并令 $\hat{P}$ 为包含 $\hat{c}$ 的政党。[4]

首先，算法会检查其猜测是否有效，也就是说，是否满足 $N(w, \hat{c}) > \hat{s}$ 以及 $N(w, p) > \hat{s}$（这是 $\mathrm{MM}_{\mathcal{E}}(w) > \hat{s}$ 的必要条件）。接下来，算法会为每个政党 $\widetilde{P} \in \mathrm{P} \setminus \{P, P_w, \hat{P}\}$ 搜寻一位合适的提名人 $c_{\widetilde{P}}$，使其满足 $N(w, c_{\widetilde{P}}) > \hat{s}$。如果除 $P$、$P_w$ 和 $P$ 之外的每个政党都找到这样的候选人，NP-Maximin 就返回“否”。如果所有猜测都已穷尽，但算法仍未输出“否”，则返回“是”。

为了证明 NP-Maximin 的正确性，首先注意，假设猜测正确，第 5 行检查的条件必须成立

[4]注意到，对于任意一对候选人 $c, c'$，值 $N_{\mathcal{E}'}(c, c')$ 与其他提名无关，因此也与缩减选举无关。因此，在本文余下部分中，我们将省略表示（缩减）选举的下标。

**NP-Maximin 算法**求解 最大最小法下的必要总统，适用于最大最小法。

**输入：** 一个必要总统问题的实例 $(\mathrm{E}, \mathrm{P}, p)$，其中候选人集合为 $C$。

1: **对于所有** $w \in C \setminus P$，**执行**
2: 令 $P_w$ 为 P 类中包含 $w$ 的政党。
3: **对所有** $\hat{c} \in C \setminus (P \cup P_w)$，**执行**
4: 令 $\hat{s} = N(p, \hat{c})$，并令 $\hat{P}$ 为包含 $\hat{c}$ 的政党。
5: **如果** $N(w, \hat{c}) > \hat{s}$ 且 $N(w, p) > \hat{s}$，**则**
6: **if** $\forall \widetilde{P} \in \mathrm{P}$ 类 $\setminus \{P, P_w, \hat{P}\}$ $\exists c \in \widetilde{P}$: $N(w, c) > \hat{s}$ **那么**
7: **返回** “否”。
8: **返回**“是”。

对于 $p$、$w$ 和 $\hat{c}$，因为根据我们的定义

$$\hat{s} = N(p, \hat{c}) = \mathrm{MM}_{\mathcal{E}'}(p) < \mathrm{MM}_{\mathcal{E}'}(w) \leq \min\{N(w,p), N(w, \hat{c})\}.$$

同样，对于 E 中除这三个候选人之外的每位提名人 $c$，我们知道 $\hat{s} = \mathrm{MM}_{\mathcal{E}'}(p) < \mathrm{MM}_{\mathcal{E}'}(w) \leq N(w, c)$。因此，算法会在第 6 行找到一个满足要求的候选人 $c$：在每个不包含 $p$，$w$ 或 $\hat{c}$ 的政党中，都有 $\hat{s}$ 满足条件 $N(w, c) > \hat{s}$。因此，NP-Maximin 将在第 7 行返回“否”。

对于另一个方向，假设 NP-Maximin 在某次迭代中返回“否”。考虑由此得到的缩减选举：在该次迭代中猜测的提名人 $\hat{P}$、$P_w$ 和 $P$ 分别提名候选人 $p$、$w$ 和 $\hat{c}$，而每个其余政党 $P$ 都提名一个满足 $N(w, c_{\widetilde{P}}) > \hat{s}$ 的候选人 $c_{\widetilde{P}}$。注意，这样的候选人确实存在，因为算法在返回“否”的那次迭代中发现第 6 行的条件成立。因此，这种方法确实得到一个候选人集合为 $C'$ 的缩减选举 $\mathrm{E}_{C'}$。显然，我们关于提名人的假设确保 $\mathrm{MM}_{\mathcal{E}_{C'}}(w) = \min_{c\in C'\setminus\{w\}} N(w, c) > \hat{s} = N(p, \hat{c}) \geq \mathrm{MM}_{\mathcal{E}_{C'}}(p)$，因此 $p$ 不是 $\mathrm{E}_{C'}$ 中的获胜者。

为了估计该算法的计算复杂性，我们首先注意到，候选人 $w$ 和 $\hat{c}$ 至多有 $|C|^2$ 种可能的猜测。对于每一对候选人 $c$ 和 $c'$，计算得分 $N_{\mathcal{E}}(c, c')$ 需要 $O(|C|^2\cdot|V|)$ 步。对于某个固定的猜测，检查第 6 行中的条件需要 $O(|C|)$ 时间。因此，NP-Maximin 的总运行时间为 $O(|C|^2(|C|+|V|))$。□

最后，我们证明，与科普兰$^{\alpha}$和最大最小法不同，即使选民数量为常数，排序对中的必要总统在计算上也是困难的。定理 4.3 依赖于从 3-SAT 的归约，而定理 4.4 的证明给出了从 W[1]-困难的多彩团问题 [28]出发的参数化归约。

定理 4.3。 即使最大政党规模为 $s = 2$ 且选民数量为 $|V| = 12$，规则 $R$ 的必要总统对于排序对仍是 coNP-完全的。

证明。我们给出一个从 NP完全问题 (2,2)-E3-SAT 出发的归约。设我们的输入公式为 $\varphi = \bigwedge_{C\in\text{候选人集合}} C$，定义在一组 $X$ 变量上。我们使用记号 $X = \{x_1, \ldots\ldots, x_r\}$ 和 $\mathrm{C} = \{C_1, \ldots\ldots, C_q\}$。

为了定义排序对规则下必要总统问题的一个实例，我们令候选人集合为

$$\{p, w\} \cup \{x, \overline{x} : x \in X\} \cup \{C_i^j, C_i^{\neg j} : C_i \in C, j \in [3]\}$$

其中 $p$ 是指定候选人。我们将 $P=\{p\}$和 $P_w=\{w\}$定义为单元素政党，并且对于每个变量 $x \in X$，还加入政党 $P_x=\{x, \bar{x}\}$；对于每个子句 $C_i \in \mathbf{C}$，则加入三个政党 $P_{C_i}^j=\{C_i^j, C_i^{\neg j}\}$，$j \in [3]$。直观而言，政党的提名人对应于为变量 $x \in X$选择真值赋值，而为某个子句 $C_i \in \mathbf{C}$提名候选人 $C_i^j$则对应于将子句 $C_i$中的文字 $j^{\text{th}}$设为真。注意，最大政党规模确实为 $s=2$。

我们将选民集合定义为 $Y \cup Z$，其中 $Y=\{y_h, y_h' : h \in [4]\}$ 和 $Z=\{z_h, z_h' : h \in [2]\}$，因此选民数量为 $n=12$。为定义选民的偏好，我们定义以下候选人集合：$L=\{x, \bar{x} : x \in X\}$ 包含所有文字，并定义 $\mathbf{C}^+=\{C_i^j : C_i \in \mathbf{C}\ j \in [3]\}$ 和 $\mathbf{C}^-=\{C_i^{\neg j} : C_i \in \mathbf{C}\ j \in [3]\}$。此外，对于每个文字 $\ell \in L$，我们令

$$A(\ell)=\{C_i^j : \bar{\ell} \text{ is the } j^{\text{th}} \text{ literal in } C_i\}.$$

注意，集合 $A(\ell)$、$\ell \in L$构成了候选人集合 $\{C_i^j : j \in [3], C_i \in \mathbf{C}\}$ 的一个划分。

为定义选民的偏好，我们首先需要引入一些额外的记号。我们固定一个在所定义的所有候选人集合上的任意排序。对于候选人的任意子集 $S$，我们用 $\overrightarrow{S}$表示与此固定排序相对应的（严格）偏好排序；类似地，我们用 $\overleftarrow{S}$表示 $S$上的反向偏好排序。于是，选民的偏好如下：

$y_h$ 对于 $h \in [4]$：

$$p \succ \overrightarrow{L} \succ C_1^{\neg 1} \succ C_1^{\neg 2} \succ C_2^{\neg 1} \succ C_2^{\neg 2} \succ \cdots \succ C_q^{\neg 1} \succ C_q^{\neg 2} \succ C_1^{\neg 3} \succ C_2^{\neg 3} \succ \cdots \succ C_q^{\neg 3} \succ \overrightarrow{C^+} \succ w$$

$y_h'$ 对于 $h \in [2]$：

$$\overleftarrow{C^+} \succ w \succ C_q^{\neg 2} \succ C_q^{\neg 3} \succ C_{q-1}^{\neg 2} \succ C_{q-1}^{\neg 3} \succ \cdots \succ C_1^{\neg 2} \succ C_1^{\neg 3} \succ p \succ C_q^{\neg 1} \succ C_{q-1}^{\neg 1} \succ \cdots \succ C_1^{\neg 1} \succ \overleftarrow{L}$$

$y_h'$ 对于 $h \in \{3, 4\}$：

$$\overleftarrow{C^+} \succ C_q^{\neg 3} \succ C_{q-1}^{\neg 3} \succ \cdots \succ C_1^{\neg 3} \succ w \succ C_q^{\neg 1} \succ C_q^{\neg 2} \succ C_{q-1}^{\neg 1} \succ C_{q-1}^{\neg 2} \succ \cdots \succ C_1^{\neg 1} \succ C_1^{\neg 2} \succ p \succ \overleftarrow{L}$$

$z_h$ 对于 $h \in [2]$：

$$x_1 \succ A(x_1) \succ x_2 \succ A(x_2) \succ \cdots \succ x_r \succ A(x_r) \succ \bar{x}_1 \succ A(\bar{x}_1) \succ \bar{x}_2 \succ A(\bar{x}_2) \succ \cdots \succ \bar{x}_r \succ A(\bar{x}_r) \succ w \succ p \succ \overrightarrow{C^-}$$

$z_1'$：$\overleftarrow{C^-}$候选人集合 $\succ p \succ w$

$$\succ \bar{x}_r \succ A(\bar{x}_r) \succ \bar{x}_{r-1} \succ A(\bar{x}_{r-1}) \succ \cdots \succ \bar{x}_1 \succ A(\bar{x}_1) \succ x_r \succ A(x_r) \succ x_{r-1} \succ A(x_{r-1}) \succ \cdots \succ x_1 \succ A(x_1);$$

$z_2'$：$\overleftarrow{C^-}$候选人集合 $\succ w \succ p$

$$\succ \bar{x}_r \succ A(\bar{x}_r) \succ \bar{x}_{r-1} \succ A(\bar{x}_{r-1}) \succ \cdots \succ \bar{x}_1 \succ A(\bar{x}_1) \succ x_r \succ A(x_r) \succ x_{r-1} \succ A(x_{r-1}) \succ \cdots \succ x_1 \succ A(x_1).$$

至此，我们完成了必要总统问题实例 $I$的构造。接下来将证明：$I$存在一个缩减选举，其中$p$不是获胜者，当且仅当 $\varphi$ 可满足。

我们先计算 $I$中每一对不同候选人 $c$和 $c'$ 的值 $N(c, c')$。从选民偏好中可以直接观察到：

- $N(p, \ell)=10$ 对每个文字 $\ell \in L$；
- $N(C_i^j, w)=10$ 对每个子句 $C_i \in$候选人集合 和 $j \in [3]$；
- $N(p, C_i^{\neg 1})=N(C_i^{\neg 1}, C_i^{\neg 2})=N(C_i^{\neg 2}, C_i^{\neg 3})=N(C_i^{\neg 3}, w)=8$对每个子句 $C_i \in$候选人集合；
- $N(w, p)=7$；
- $N(a, a')=6$ 对于上文未列出、且满足 $(a, a') \notin (L \times$候选人集合$^+) \cup ($候选人集合$^+ \times L)$ 的每一对不同候选人 $(a, a')$。

现在考虑一个文字 $\ell \in L$ 以及某个候选人 $C_i^j \in$候选人集合$^+$。显然，$Y$中恰好有四名选民偏好 $\ell$胜过 $C_i^j$。此外，若 $C_i^j \in A(\ell)$，则 $Z$中的全部四名选民都偏好 $\ell$胜过 $C_i^j$；否则，$Z$中只有两名这样的选民。因此，我们得到

$$N(\ell, C_i^j)=\begin{cases} 8, & \text{if } C_i^j \in A(\ell), \\ 6, & \text{if } C_i^j \notin A(\ell). \end{cases} \tag{5}$$

设 $F$ 表示在某个候选人集合 $C'$上的缩减选举 $\mathrm{E}_{C'}$ 中，于排序对的获胜者确定过程所构造的多数图的无环子图 $D$中添加的边的集合；见图1。注意，弧集

$$F_0=(\{p\} \times L) \cup (C^+ \times \{w\}) \cup \{(p, C_i^{\neg 1}), (C_i^{\neg 1}, C_i^{\neg 2}), (C_i^{\neg 2}, C_i^{\neg 3}), (C_i^{\neg 3}, w) : C_i \in C\}$$

是无环的。事实上，甚至 $F_0 \cup (L \times$候选人集合$^+)$ 也是无环的。由于——根据我们对候选人之间两两比较的观察——该集合包含该实例的加权多数图中所有权重至少为 8 的弧，我们得到

$$F_0 \cap (C' \times C') \subseteq F \subseteq F_0 \cup (L \times C^+) \cup \{(w, p)\}.$$

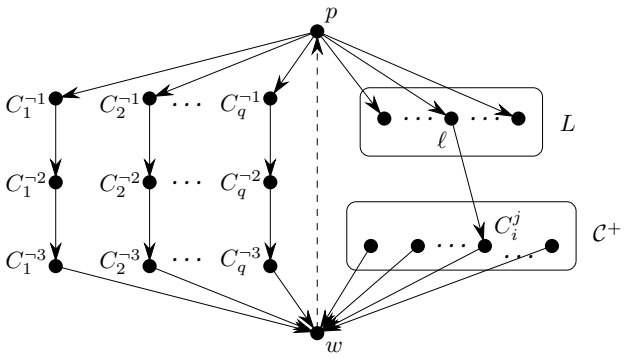


**图1：定理 4.3 的证明中构造的实例在排序对获胜者判定过程中形成的无环子图 $D$ 示意图。实线弧始终包含在 $D$中，而以虚线箭头表示的弧 $(w, p)$，在 $D$中可能存在，也可能不存在。图中假设 $C_i^j \in A(\ell)$。**

还要注意，$p$ 唯一可能的入弧是 $(w, p)$，这意味着当且仅当弧 $(w, p)$ 存在于 $F$ 中时，$p$才不是 约化选举 $\mathrm{E}_{C'}$ 中的获胜者。我们将证明，当且仅当 $\varphi$ 可满足时，这种情况才会发生。

首先，假设 $(w,\ p) \in F$。这显然意味着在 $D$ 中不存在从 $p$ 到 $w$ 的路径，因此尤其是，

(a) 对于每个 $C_i \in$ 候选人集合 C，都存在某个 $j \in [3]$，使得 $C_i^{\neg j} \notin C'$；(b) $F$不包含从 $L \times$ 候选人集合 $\mathrm{C}^+$出发的弧。

注意，$\mathrm{E}_{C'}$ 中的提名人自然确定了一个真值赋值，因为每个政党 $P_x = \{x,\ x\}$，$x \in X$，在 $\mathrm{E}_{C'}$中恰有一位提名人，即 $|C' \cap P_x| = 1$。令 $\alpha_{C'}$ 表示这样一种真值赋值：恰好将 $C' \cap L$中的文字赋值为真。

现在考虑某个子句 $C_i \in \mathrm{C}$。根据 (a)，存在某个$j \in [3]$，使得政党 $P^j_{C_i}$的提名人为 $\bar{C}_i^j$。令 $\ell$为 $C_i$中的 $j^{\text{th}}$文字；则根据定义，$C_i^j \in A(\ell)$。注意，$\ell \notin C'$，否则弧 $(\ell,\ C_i^j)$的两个端点都在 $C'$中，因此该弧会出现在 $F$中，这与 (b) 矛盾。由于 $\ell \notin C'$蕴含 $\ell \in C'$，我们得出结论：在真值赋值 $\alpha_{C'}$中，$\ell$被赋值为真，因此子句 $C_i$得到满足。由于这对 C 中的所有子句都成立，我们得出 $\alpha_{C'}$ 满足 $\varphi$。

其次，现在假设 $\alpha$是 $\varphi$的一个满足赋值。考虑缩减选举 $\mathrm{E}'$，其中每个政党 $P_x, x \in X$都提名在 $\alpha$中被赋值为真的文字，而每个政党 $P^j_{C_i}$当且仅当子句 $C_i$中的第 $j$个文字在 $\alpha$中被赋值为真时，才提名 $C_i^j$。

很容易验证，$p$ 不是 $\mathrm{E}'$中的获胜者。为此，考虑在 $\mathrm{E}'$中进行排序对获胜者判定过程时构造的多数图的无环子图 $D$。我们需要证明，$D$ 中仅包含权重至少为 8 的弧的子图不存在从 $p$到 $w$的路径，因此弧 $(w,p)$会被加入 $D$。首先，注意到由于 $\alpha$满足 $\varphi$，对于每个子句 $C_i \in \mathrm{C}$，都存在某个索引 $j$，使得 $C_i^j$ 在 $\mathrm{E}'$中被提名，因此 $C_i^{\neg j}$不会出现在 $\mathrm{E}'$中。因此，在 $D$中不存在一条从 $p$到 $w$且经过候选人集合 $\mathrm{C}^-$中的候选人的路径。其次，注意到 $D$在 $L \times \mathrm{C}^+$中不含任何弧，因为如果某个候选人 $C_i^j$是 $\mathrm{E}'$中的提名人，那么 $\alpha$必须将文字 $\ell$（其中 $C_i^j \in A(\ell)$）赋值为假，因此该文字不会在 $\mathrm{E}'$中被提名。这表明 $(w,p)$是 $D$中的一条弧，因此 $p$不是 $\mathrm{E}'$中的获胜者。□

我们以如下结果作结，该结果证明：即使将政党数量作为参数且选民数量为 20，排序对中的必要主席仍然具有难解性。

在证明中，我们给出一个从 W[1]-难问题多色团 [28], 出发的参数化归约，其定义如下。

**问题**多色团**实例：** 一个图 $G = (U,\ E)$，其顶点集 $U$ 被划分为 $k$ 个独立集 $U_1,\ U_2,\ \ldots\ldots,\ U_k$，以及一个整数 $k$。**询问：** 是否存在一个多色团，即大小为 $k$ 、且从每个集合 $U_i$、$i \in [k]$中各含一个顶点的团?

定理 4.4. 即使选民数量为 $|V| = 20$，排序对中的必要主席关于表示政党数量的参数 $t$仍是 W[1]-难的。

证明。设给定的 Multicolored Clique 实例是一个图 $G = (U,\ E)$，其顶点集 $U$划分为 $k$ 个集合 $U_1,\ U_2,\ \ldots\ldots,\ U_k$，以及一个整数 $k$。不失一般性，假设每个集合 $U_i$ 都包含相同数量的顶点，设为 $r$，因此对于每个 $i \in [k]$，我们可以将 $U_i$中的顶点记为 $u_i^1,\ u_i^2,\ \ldots\ldots,\ u_i^r$。

对于 $[k]$中两个不同的整数 $i$ 和 $j$，令 $E_{\{i,j\}}$ 表示 $G$ 中一端点位于 $U_i$、另一端点位于 $U_j$ 的边的集合；我们还将使用记号 $E_{i,\ >i} = \bigcup_h:\ i<h\le k\ E_{\{i,h\}}$ 和 $E_{i,\ <i} = \bigcup_h:\ 1\le h<i\ E_{\{i,h\}}$。此外，对于 $G$中的边集 $F \subseteq E$ 以及顶点 $u \in U$，我们令 $F\ (u)$ 表示 $F$ 中与 $u$ 相关联的那些边的集合。

为定义一个必要总统实例，我们先将其候选人集合定义为 $U \cup E \cup \{p,\ w\}$，其中 $p$ 和 $w$ 是新创建的候选人，$p$ 为指定候选人，且属于政党 $P = \{p\}$；我们还将 $P_w = \{w\}$ 设为一个单人政党。此外，对于每个 $i \in [k]$，我们将 $V_i$ 设为一个政党；类似地，对于每个满足 $i \neq j$ 的 $i,\ j \in [k]$，我们将 $E_{\{i,j\}}$ 设为一个政党。由此得到一个包含 $t = \binom{k}{2} + k + 2$ 个政党的集合 P。若 $i \in \{j,\ \ell\}$，我们就称政党 $U_i$ 和 $E_{\{j,\ell\}}$ 是相连的 。

我们将选民集合定义为 $V = X \cup Y \cup Z$，其中 $X = \{x_h,\ x'_h:\ h \in [4]\}$、$Y = \{y_h,\ y'_h:\ h \in [4]\}$ 和 $Z = \{z_h,\ z'_h:\ h \in [2]\}$；因此，选民数量为 $n = 20$。同样，如定理 4.3 的证明所示，我们固定所有已定义候选人构成的集合 $C$ 上的任意排序，并令 $\overrightarrow{C}$ 和 $\overleftarrow{C}$ 分别表示由该排序及其逆序所确定的候选人集合 $C' \subseteq C$ 上的偏好列表。

我们首先定义以下部分偏好序，作为构造所创建偏好剖面的基本单元：

$$A_i := \overrightarrow{E_{i,>i}(u_i^1)} \succ u_i^1 \succ \overrightarrow{E_{i,>i}(u_i^2)} \succ u_i^2 \succ \cdots \succ \overrightarrow{E_{i,>i}(u_i^r)} \succ u_i^r;$$

$$A'_i := \overleftarrow{E_{i,>i}(u_i^r)} \succ u_i^r \succ \overleftarrow{E_{i,>i}(u_i^{r-1})} \succ u_i^{r-1} \succ \cdots \succ \overleftarrow{E_{i,>i}(u_i^1)} \succ u_i^1;$$

$$B_i := \overrightarrow{E_{i,<i}(u_i^1)} \succ u_i^1 \succ \overrightarrow{E_{i,<i}(u_i^2)} \succ u_i^2 \succ \cdots \succ \overrightarrow{E_{i,<i}(u_i^r)} \succ u_i^r;$$

$$B'_i := \overleftarrow{E_{i,<i}(u_i^r)} \succ u_i^r \succ \overleftarrow{E_{i,<i}(u_i^{r-1})} \succ u_i^{r-1} \succ \cdots \succ \overleftarrow{E_{i,<i}(u_i^1)} \succ u_i^1.$$

现在，我们可以如下定义选民的偏好：

$x_1$：$p \succ w \succ A_1 \succ A_2 \succ \cdots \succ A_k$;

$x_2$：$w \succ p \succ A_1 \succ A_2 \succ \cdots \succ A_k$;

$x'_h$对于 $h \in \{1,\ 2\}$：$A'_k \succ A'_{k-1} \succ \cdots \succ A'_1 \succ w \succ p$

$x_h$对于 $h \in \{3,\ 4\}$：$p \succ w \succ B_1 \succ B_2 \succ \cdots \succ B_k$

$x'_h$适用于 $h \in \{3、4\}$：$B'_k \succ B'_{k-1} \succ \cdots \succ B'_1 \succ w \succ p$

$y_h$适用于 $h \in \{1、2\}$:

$$p \succ w \succ \overrightarrow{V_1} \succ \overrightarrow{E_1}\ 、_{>1} \succ \overrightarrow{V_2}、\ \overrightarrow{E_2}\ 、_{>2} \succ \cdots \succ \overrightarrow{V_{k-1}} \succ \overrightarrow{E_{k-1}}\ 、_{>k-1} \succ \overrightarrow{V_k} y'_h$$ 适用于 $h \in \{1、2\}$:

$$\overleftarrow{V_k} \succ \overleftarrow{V_{k-1}} \succ \overleftarrow{E_{k-1}}\ 、_{>k-1} \succ \cdots \succ \overleftarrow{V_2} \succ \overleftarrow{E_2}\ 、_{>2} \succ \overleftarrow{V_1} \succ \overleftarrow{E_1}\ 、_{>1} \succ w \succ p y_h$$ 适用于 $h \in \{3、4\}$:

$$p \succ w \succ \overrightarrow{V_1} \succ \overrightarrow{V_2}, \overrightarrow{E_{2,<2}} \succ \cdots \succ \overrightarrow{V_k} \succ \overrightarrow{E_{k,<k}}$$

$y'_h$对于 $h \in \{3,4\}$:

$$\overleftarrow{V_k} \succ \overrightarrow{E_{k,<k}} \succ \overleftarrow{V_{k-1}} \succ \overleftarrow{E_{k-1,<k-1}} \succ \cdots \succ \overleftarrow{V_2} \succ \overleftarrow{E_{2,<2}} \succ \overleftarrow{V_1} \succ w \succ p$$

$z_h$ for $h \in \{1,2\} : p \succ \overrightarrow{V} \succ \overrightarrow{E} \succ w$

$z'_h$ for $h \in \{1,2\} : \overleftarrow{E} \succ w \succ p \succ \overleftarrow{V}.$

至此，我们完成了构造。所给出的归约是多项式时间归约，也是参数化归约，因为

政党数量 $t = \binom{k}{2} + k + 2$ 仅取决于原始参数 $k$。

我们声称，对于所构造的规则 R 的必要主席实例，存在一个缩减选举，其中 $p$ 不是 获胜者，当且仅当 $G$ 包含一个大小为 $k$ 的多色团。这将证明该缩减的正确性，从而证明在以 $t$ 为参数时，该问题是 W[1]- 难的，即使在 $n = 20$ 的条件下也是如此。

考虑任意一个缩减选举。我们可以直接从选民偏好中观察到以下事实：

- $N(p, u) = 12$ 对每个 $u \in U$；
- $N(e, w) = 12$ 对每个 $e \in E$；
- $N(w, p) = 11$；
- $N(u, w) = 10$ 对每一对不同的 $u$ 和 $w$，其中二者均属于 $U$；
- $N(e, e') = 10$ 对于 $E$中每一对不同的 $e$ 和 $e'$；
- $N(w, u) = 10$ 对于每个 $u \in U$；
- $N(p, e) = 10$ 对于每个 $e \in E$。

现在来考察一些候选人 $u \in U$ 和 $e \in E$之间的两两比较。首先假设 $u$ 和 $e$ 属于未关联的政党。在这种情况下，它们不会同时出现在 $A_i$、 $A'_i$、 $B_i$或 $B'_i$这些区块中（对于 $i \in [k]$而言），因此 $X$中恰有一半的选民更偏好 $u$而非 $e$。此外， $Y$中也恰有一半的选民更偏好 $u$而非 $e$；显然， $Z$中的选民也同样如此。因此，除非 $u$和 $e$属于关联政党，否则二者在选民偏好中平票。

现在假设 $u$和 $e$属于关联政党，也就是说， $u \in U_i$和$e \in E_{\{i,j\}}$对应两个不同的索引 $i$、 $j \in [k]$。首先考察 $X$中的选民。显然， $u$会出现在 $A_i$、 $A'_i$、 $B_i$和 $B'_i$各项中。然而， $e$出现在 $A_i$和 $A'_i$中的充要条件是 $i < j$，而出现在 $B_i$和 $B'_i$中的充要条件是 $i > j$。此外，若 $i < j$，则当且仅当$e \in E_{i,\succ_i}(u)$时，$\{x_h, x'_h : h \in [2]\}$中的所有选民都更偏好 $e$而非 $u$；否则， $e$和 $u$在这四位选民的偏好中平票。类似地，若 $i > j$，则当且仅当$e \in E_{i,\prec_i}(u)$时，$\{x_h, x'_h : h \in \{3, 4\}\}$中的所有选民都更偏好 $e$而非 $u$；否则， $e$和 $u$在这四位选民的偏好中平票。因此，无论是 $i < j$还是 $i > j$，我们都可得出：若 $e$与 $u$关联，则 $X$中恰有六位选民更偏好 $e$而非 $u$；否则， $X$中的选民偏好为平票。

现在考虑 $Y$ 中的选民。一方面，如果 $i < j$，那么 $u$ 相较于 $e$ 更受 $Y$ 中所有选民的偏好，但选民 $y'_3$ 和 $y'_4$ 除外。另一方面，如果 $i > j$，那么 $u$ 相较于 $e$ 更受 $Y$ 中所有选民的偏好，但选民 $y_1$ 和 $y_2$ 除外。因此，无论哪种情况， $Y$ 中都有六名选民偏好 $u$ 而非 $e$，此外 $Z$ 中还有两名这样的选民。

综上所述，我们得到

$$N(u, e) = \begin{cases} 12, & \text{如果 } e \text{ 与 } u \text{ 不相邻，但它们所属的政党相连，则为 } 10, \text{ 否则。} \end{cases}$$

令 $F$ 表示添加到无环子图中的边集 $D$在缩减选举 $\mathrm{E}_{C'}$ 的排序对获胜者判定过程中构造的多数图中，候选人集合为 $C'$；见图 2。根据我们对候选人之间两两比较的观察，我们知道 $F$ 包含弧集 $(\{p\} \times U) \cup (E \times \{w\})$ 中端点均为 $\mathrm{E}_{C'}$中提名的候选人的所有弧，即候选人集合 $C'$中的候选人。也就是说， $F$包含弧集

$$F_0 = ((\{p\} \times U) \cup (E \times \{w\})) \cap (C' \times C').$$

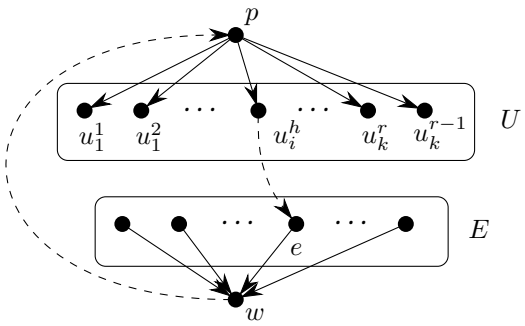


**图 2：在根据定理 4.4 证明中构造的实例，使用排序对进行获胜者判定时形成的无环子图 $D$ 的示意图。实线弧始终包含在 $D$ 中，而以虚线箭头表示的 $U \times E$ 中的弧以及弧 $(w, p)$，可能包含在 $D$中，也可能不包含。**

此外， $F$也可能包含来自 $U \times E$的一些弧，或者可能包含弧（$w, p$），因此我们得到

$$F_0 \subseteq F \subseteq F_0 \cup (U \times E) \cup \{(w, p)\}.$$

现在请注意，任意弧 $(u, e) \in F$都会在 $D$中形成一条从 $p$到 $w$的路径。事实上，当且仅当 $F \cap (U \times E) \neq \emptyset$时， $D$中才存在一条从 $p$到 $w$的路径；而这又当且仅当 $(w, p) \notin F$时成立。此外，请注意， $p$唯一可能的入弧是 $(w, p)$，这意味着，在缩减选举 $\mathrm{E}_{C'}$中， $p$不是 获胜者，当且仅当弧 $(w, p)$ 存在于 $F$ 中；等价地，当 $F \cap (V \times E) = \emptyset$时成立。

仍需证明， $F \cap (V \times E) = \emptyset$ 成立，当且仅当除 $P$和 $P_w$ 以外所有政党的提名人构成 $G$中的多色团的顶点和边；这等价于如下性质：对于属于关联政党的任意两位提名人 $e \in E$和 $u \in U$， $e$在 $G$中与 $u$相邻。

令 $u_i$ 表示每个 $i \in [k]$对应的 $U_i$ 提名人。注意，若某个政党 $E_{\{i,j\}}$ 的提名人 $e_{\{i,j\}}$ 与 $u_i$不相邻，那么根据 $N(u_i, e_{\{i,j\}}) = 12$，构造有向图 $D$时加入的弧集合 $F$将包含弧 $(u_i, e_{\{i,j\}})$；这里我们依据这样一个事实：弧集合 $(\{p\} \times U) \cup (E \times \{w\}) \cup (U \times E)$ 是无环的，其中包含所有满足 $N(c, c') \geq 12$ 的候选人对 $(c, c')$。反过来，若将 $(u, e) \in U \times E$加入 $F$，则必有 $N(u, e) = 12$；这只有在 $u$和 $e$属于关联政党且 $e$与 $u$不相邻时才会发生，正如所需；注意，这里我们依据这样一个事实：加入 $(w, p)$ 后，不会再有任何弧从 $U \times E$加入 $F$，因为加入 $(w, p)$ 会形成一条从 $E$中的每位候选人到 $U$中每位候选人的路径，从而阻止之后再将 $(U \times E)$ 中的某条弧加入 $F$。

这证明，当且仅当 $G$ 包含一个多色团时，才可能提名候选人，使得在由此产生的选举中 $p$不是获胜者。□

## 5 结论与展望

我们探讨了若干常见投票规则下必要总统问题的计算复杂性；结合此前在 [3], [31], 和 [30], 中的研究，我们的工作全面呈现了

选举前候选人提名过程中政党所面临问题的计算可处理性。

未来研究的一个可能方向，是将单峰 [11] 或单交叉 [23] 偏好下简单多数制下的必要总统问题已有的可解性结果，推广到不同的投票规则或其他结构化领域。

一个有趣的新课题是研究多赢家选举中的候选人提名问题。假设选举的目标是选出一个由 $k$ 名成员组成的委员会。在这种情况下，自然可以假设一个政党可以提名多名候选人。如果该政党希望至少有一名提名人进入委员会，或者希望尽可能增加进入委员会的提名人数，那么它的最优策略是什么？

## 致谢

IldikóSchlotter 获匈牙利科学院动量计划（LP2021-2）及亚诺什·博尧伊研究奖学金资助。Katarína Cechlárová 获 VEGA 1/0585/24 和 APVV-21-0369 资助。

## 参考文献

[1] Piotr Berman、Marek Karpinski 和 Alexander D. Scott。2003。《MAX-3SAT 短对称实例的近似困难性》。技术报告 TR03-049。计算复杂性电子讨论会。[2] Markus Brill 和 Vincent Conitzer。2015。策略性投票与策略性候选。载于第 *29* 届 *AAAI*人工智能大会论文集（*AAAI 2015*）。AAAI Press，819–826。https://dl.acm.org/doi/10.5555/2887007.2887121[3] Katarína Cechlárová、Julien Lesca、Diana Trellová、Martina Hančová 和 Jozef Hanč。2023。候选人提名的困难性。自主智能体与多智能体系统 37 2023 37 10 1007 10458 023 09622 9 ( )，。https://doi.org/[4]。/s - - - Jiehua Chen、Piotr Faliszewski、Rolf Niedermeier 和 Nimrod Talmon。2017。选民人数较少的选举：候选人控制可能很容易。人工智能研究杂志 60 （ 2017 ），937–1002。https://doi.org/10.1613/jair.5515[5] Marek Cygan 、 Fedor V Fomin、Łukasz Kowalik、Daniel Lokshtanov、Dániel Marx、Marcin Pilipczuk、Michał Pilipczuk 和 Saket Saurabh。2015。《参数化算法》。施普林格。[6] Argyrios Deligkas、Eduard Eiben 和 Tiger-Lily Goldsmith。2022。含政党提名候选人的霍特林–唐斯模型的参数化复杂性。载于第 *31* 届国际人工智能联合会议论文集（*IJCAI 2022*）。244–250。https://doi.org/ 10 24963 2022 35 。/ijcai。/ Reinhard Diestel。[7]2005《图论》。数学研究生教材，第 173 卷。Springer-Verlag，柏林、海德堡。[8] Rodney G. Downey 和 Michael R. Fellows。2013。《参数化复杂性的基础》。施普林格，伦敦。[9] Bhaskar Dutta、Matthew O. Jackson 和 Michel Le Breton。2001。策略性候选与投票程序。《计量经济学》69，4（2001），1013–1037。https: // doi.org/10.1111/1468-0262.00228[10] Gábor Erdélyi、Marc Neveling、Christian Reger、Jörg Rothe、Yongjie Yang 和 Roman Zorn。2021。迈向谜题的完整解答：通过替换、增加和删除候选人或选民实施控制的复杂性。自主智能体与多智能体系统 35 41 2021 10 1007 10458 021 09523 9 : ( )。[11]https://doi.org/ 。/s - - - Piotr Faliszewski、Laurent Gourvès、Jérôme Lang、Julien Lesca 和 Jérôme Mon- not。2016。政党提名选举获胜者有多难？载于第 *25* 届国际人工智能联合会议论文集（*IJCAI 2016*）。AAAI Press，257 263 10 5555 3060621 3060658 –。https://doi.org/ 。/ 。[12]Piotr Faliszewski、Edith Hemaspaandra、Lane A Hemaspaandra 和 Jörg Rothe。2009。Llull 和科普兰投票在计算上能够抵御贿赂和建设性控制。人工智能研究杂志 35（ 2009），275–341。https: //doi.org/ 10 1613 2697 。/jair。Piotr Faliszewski 和 Jörg Rothe。2016。[13]投票中的控制与贿赂。载于《计算社会选择手册》，Felix Brandt、Vincent Conitzer、Ulle Endriss、Jérôme Lang 和 Ariel D. Procaccia（编）。剑桥大学出版社，146–168。[14] Jiong Guo、Yash Raj Shrestha 和 Yongjie Yang。2015。政党选举的预测有多可信？载于第 *14* 届自主代理与多智能体系统国际会议论文集（ *AAMAS 2015*）。IFAAMAS，1431–1439 10 5555 2772879 2773335 。https://dl.acm.org/doi/abs/ 。/ 。[15]Paul Harrenstein 、 Grzegorz Lisowski 、 Ramanujan Sridharan 和 Paolo Turrini。2021。政党提名人的霍特林–唐斯框架。载于第 20 届自主代理与多智能体系统国际会议（AAMAS *2021* 593 601 10 5555 3463952 3464025 ）。–。[16]2022https://dl.acm.org/doi/ 。/ 。Paul Harrenstein 和 Paolo Turrini。计算基于选区的提名的纳什均衡。载于第 21 届自主代理与多智能体系统国际会议（AAMAS 2022）论文集。IFAAMAS，588–596。https://dl. acm.org/doi/abs/10.5555/3535850.3535917[17] John E. Hopcroft 和 Richard M. Karp。1973。一种 $n^{5/2}$ 二分图最大匹配算法。《SIAM 计算杂志》2，4（1973），225–231。https: //doi.org/ 10 1137 0202019 。/ John J. Bartholdi III、Craig A. Tovey 和 Michael A. Trick[18]。1992。控制一场选举有多难？《数学与计算机建模》16，8-9（1992），27–40。https://doi.org/10.1016/0895-7177(92)90085-Y [19] TVN24 英文新闻。2020。公民联盟候选人 Małgorzata Kidawa-Błońska 退出总统竞选。https://tvn24.pl/tvn24-news-in-english/civic-coalition-candidate-malgorzata-kidawa-blonska-quits-presidential-run- st4584889。访问日期：2025-09-30。[20] Jérôme Lang、Vangelis Markakis、Nicolas Maudet、Svetlana Obraztsova、Maria Polukarov 和 Zinovi Rabinovich。2019。积极候选人参与的策略性候选。技术报告 hal- 02294985 02294985 1 。HAL。[21]https://hal.science/hal- v Jérôme Lang、Nicolas Maudet、Maria Polukarov 和 Alice Cohen-Hadria。2025。常见投票规则下的策略性参选均衡。《计算机系统理论》69 2025 23 10 1007 00224 025 10220 3 ( )，。https://doi.org/ 。/ s - - - Grzegorz Lisowski[22]。2022。策略性被提名人选择与竞赛解。载于第 19 届欧洲多智能体系统会议（EUMAS 2022）论文集。 239 256 10 1007 978 3 031 20614 6_14 –。https://doi.org/ 。/ - - - - Neeldhara Misra[23]。2019。政党提名的参数化复杂性。载于第 6 届算法决策理论国际会议（ADT 2019）论文集（《计算机科学讲义》，第 11834 卷）。施普林格，112–125。https: //doi.org/ 10 1007 978 3 030 31489 7_8 。/ - - - - 《世界报》。土耳其反对派推选 Kemal Kilicdaroglu，[24]2023在五月选举中挑战 Erdogan。https://www.lemonde.fr/en/elections/article/2023/03/07/turkey-s-opposition-picks-candidate-to-take-on-erdogan-in-may- election 6018420 84.html。访问日期：2025-09-30。[25] Svetlana Obraztsova、Maria Polukarov、Edith Elkind 和 Zinovi Rabinovich。2015。懒惰候选人的策略性候选人提名博弈。载于第 24 届国际人工智能联合会议（IJCAI 2015）论文集。610–616。https://doi.org/ 10 5555 2832249 2832334 。/ 。Michaela Kovacova（TASR：斯洛伐克共和国通讯社）。2024。Election24：Danko 退出总统候选资格。https://newsnow.tasr.sk/ election24-danko-withdraws-presidential-candidacy/。访问日期：2025-09-30。[27] Tomasz Perek、Piotr Faliszewski、Maria Silvia Pini 和 Francesca Rossi。2013。失去选民的复杂性。载于第 12 届自主代理与多智能体系统国际会议（AAMAS 2013）论文集。407–414。https: //dl.acm.org/doi/abs/ 10 5555 2484920 2484986 。/ 。Krzysztof Pietrzak[26]。固定字母表最短公共超序列和最长公共子序列问题的参数化复杂性。《计算机系统科学杂志》67，4（2003），757–771。https://doi.org/10.1016/S0022- 0000(03)00078-3[29] Maria Polukarov、Svetlana Obraztsova、Zinovi Rabinovich、Alexander Kruglyi 和 Nicholas R. J ennings。2015。策略性候选中的均衡收敛。载于第 24 届国际人工智能联合会议（IJCAI *2015* 624 630 10 5555 2832249 2832336 ）论文集。–。[28]2003https://dl.acm.org/doi/ 。/ 。Ildikó Schlotter 和 Katarína Cechlárová。孔多塞一致投票规则下的候选人提名。载于第 24 届自主代理与多智能体系统国际会议（AAMAS 2025）论文集。1858–1866。https://dl.acm.org/doi/ 10 5555 3709347 3743822 。/ 。[30]2025Ildikó Schlotter、Katarína Cechlárová[31] 和 Diana Trellová。2024。基于位置计分规则的选举中候选人提名的参数化复杂性。《自主智能体与多智能体系统》38（2024），28。https://doi.org/ 10.1007/s10458-024-09658-5[32]《斯洛伐克观察家》。2019。Mistrík 退出总统竞选。https://spectator.sme.sk/politics-and-society/c/mistrik-withdraws-from-the-presidential-race。访问日期：2025-09-30。[33] 维基百科。2024。2024年美国总统选举。https://en.wikipedia. org/wiki/2024_United_States_presidential_election。访问日期：2025-09-30。[34] 维基百科。2025。2022年匈牙利议会选举。https://en.wikipedia. org/wiki/2022_Hungarian_parliamentary_election。访问日期：2025-09-30。