

ARROW AND GIBBARD SATTERTHWAITE THEOREMS IN LEAN

ARTHUR FREITAS RAMOS, DAVID BARROS HULAK, AND RUY J. G. B. DE QUEIROZ

ABSTRACT. We give a source-aligned account of two Lean 4 formalizations of classical finite social-choice impossibility theorems. For a finite nonempty electorate and a finite alternative set containing three distinct alternatives, Arrow’s theorem derives dictatorship from unanimity and independence of irrelevant alternatives. Its proof expands a weakly decisive coalition from one ordered pair to all pairs, then contracts decisive coalitions to a singleton. The Gibbard–Satterthwaite development proves that every onto strategy-proof choice function on the unrestricted strict-ranking domain is dictatorial by constructing a social welfare function and applying that Arrow implementation. We explain the essential construction in full: top-two profiles define pairwise comparisons; strategy-proof monotonicity and three-top profiles establish a strict total order; predecessor counts provide injective natural-number ranks. Exact hypotheses, vendored dependencies, pinned source identities, prior formalizations, and the distinction between historical registry verification and the present source audit are recorded. The contribution is an exposition and audit of reusable formalization artifacts, not a new social-choice theorem or a claim of first formalization.

1. INTRODUCTION

Arrow’s theorem and the Gibbard–Satterthwaite theorem expose two different limits of collective decision making. Arrow studies the aggregation of individual rankings into a social ranking. Gibbard–Satterthwaite studies the selection of one alternative when each voter may misreport a ranking. Their classical sources are Arrow’s 1950 paper [1], Gibbard’s 1973 paper [2], and Satterthwaite’s 1975 paper [10].

The artifacts examined here are Arthur Freitas Ramos’s finite Arrow development [7] and its Gibbard–Satterthwaite extension [8], registered separately in Palomar [4, 5]. Their mathematical dependence makes a combined exposition natural. The second repository contains a byte-identical copy of the first repository’s four substantive Arrow modules, and its main theorem invokes the copied Arrow theorem. The manuscript’s subject is therefore one proof chain: decisive coalitions establish Arrow; strategy-proof choice is converted into a valid strict social ranking; Arrow’s dictator is transferred back to the original choice rule.

The source pins used throughout are

Arrow	4f405d27d0b244574bb5e5d3d79bae0c66bda16e
Gibbard–Satterthwaite	8398e65a99cd3d5e973c64963c4bab8c0491947c

These identify repository snapshots, not merely branch names. Both use Lean v4.35.0-rc2 and the same Mathlib manifest revision, recorded in Section 7.

Our main expository task is to make the intermediate social welfare function explicit. Choosing winners on two-alternative tests does not by itself produce a ballot: the resulting pairwise relation might have cycles or tied numerical scores. The formalization proves the order laws before using

Date: October 1, 2026.

2020 Mathematics Subject Classification. 91B14, 68V20.

Key words and phrases. social choice, Arrow theorem, Gibbard–Satterthwaite theorem, decisive coalition, strategy-proofness, Lean 4, formal verification.

Author ORCID identifiers: Arthur Freitas Ramos, [0009-0003-3568-0325](#); David Barros Hulak, [0009-0002-8056-1774](#); Ruy J. G. B. de Queiroz, [0000-0003-1482-0977](#).

Copyright 2026 the authors. This manuscript is licensed under [Creative Commons Attribution 4.0 International](#). The cited Lean repositories have their separate BSD-3-Clause licenses.

predecessor counts to construct an injective ranking. This step is essential to the reduction, and was specifically absent from the optional proof sketch noted in the Gibbard–Satterthwaite registry review [5].

The mathematical route is established. Satterthwaite describes Gibbard’s construction of a social welfare function from strategy-proof voting, the verification of Pareto and independence conditions, the use of Arrow, and the transfer of dictatorship [10, pp. 205–207]. Formal proofs of both theorems also predate these Lean repositories; Nipkow formalized them in Isabelle/HOL [3]. Peters’s earlier public Lean development proves a resolute unanimous strategy-proof choice correspondence dictatorial using voter cloning and strong induction [6]; we consulted its pinned source, without rebuilding it. The Arrow metadata separately acknowledges Joris Roos’s independent three-candidate Fourier-analytic Lean formalization [9]. We claim neither a new reduction nor priority among formalizations. The present contribution is a readable explanation of the precise finite strict-ranking implementations, their interfaces, and their verification boundary.

2. THE STRICT RANKING DOMAIN AND EXACT STATEMENTS

Let V be a finite nonempty electorate and A a finite set of alternatives. The main results take explicit witnesses $x, y, z \in A$ satisfying

$$(1) \quad x \neq y, \quad x \neq z, \quad y \neq z.$$

Thus the alternative-set hypothesis is at least three alternatives, without fixing a particular cardinality. The Lean interfaces use `Fintype V`, `Fintype A`, `DecidableEq V`, `DecidableEq A`, and `Nonempty v`. The Gibbard–Satterthwaite interface also supplies `Nonempty A`; condition (1) already entails it mathematically. No assumption of two or more voters is made. A one-voter electorate is included.

2.1. Ballots as injective natural ranks. The common representation is

$$(2) \quad \mathcal{B}(A) = \{s : A \rightarrow \mathbb{N} \mid s \text{ is injective}\}, \quad a \succ_s b \iff s(a) < s(b).$$

Lower numbers mean higher preference. Injectivity makes $\succ_s$ irreflexive, asymmetric, transitive, and total on distinct alternatives. Rankings need not be consecutive integers or bounded by $|A| - 1$. Conversely, every strict linear order on a finite set can be represented by counting predecessors in that order. This elementary mathematical observation explains the domain; the pinned Arrow library does not separately state a general representation theorem. Thus (2) supplies the full strict-ranking domain. It excludes indifference and partial orders.

A profile is $P \in \mathcal{P}(V, A) = V \rightarrow \mathcal{B}(A)$. We write $a \succ_{P_v} b$ for voter v ’s comparison. Both a social welfare function and a social choice function are defined on every profile:

$$F : \mathcal{P}(V, A) \rightarrow \mathcal{B}(A), \quad f : \mathcal{P}(V, A) \rightarrow A.$$

Unrestricted domain is built into these function types. The implementations do not begin with a quotient that identifies different numerical encodings of the same ranking. Their axioms and conclusions concern the induced comparisons; equality of complete natural-number score functions is not Arrow’s dictatorship conclusion.

2.2. Arrow’s axioms. A social welfare function F is *unanimous* if

$$(3) \quad \forall P, a, b, \quad (\forall v \in V, a \succ_{P_v} b) \implies a \succ_{F(P)} b.$$

It satisfies *independence of irrelevant alternatives* (IIA) if

$$(4) \quad [\forall v, a \succ_{P_v} b \iff a \succ_{Q_v} b] \implies [a \succ_{F(P)} b \iff a \succ_{F(Q)} b]$$

for all profiles and alternatives. A voter d is a dictator of F precisely when

$$(5) \quad \forall P, a, b, \quad a \succ_{P_d} b \iff a \succ_{F(P)} b.$$

There is no anonymity or neutrality hypothesis. Social transitivity and completeness come from the codomain $\mathcal{B}(A)$.

Theorem 2.1 (Finite strict-ranking Arrow theorem). *Under the finite-domain hypotheses above and (1), every unanimous social welfare function satisfying IIA has a dictator in the sense of (5).*

The selected declaration is `Arrow.Palomar.arrowImpossibility`; the substantive theorem is `Arrow.arrowImpossibility` in `Arrow/ArrowTheorem.lean`.

2.3. Strategy proof choice. For a profile P , voter v , and ballot b , let $P[v \leftarrow b]$ replace only v 's report. Strategy-proofness means

$$(6) \quad \forall P, v, b, \quad \neg(f(P[v \leftarrow b]) \succ_{P_v} f(P)).$$

The comparison uses the true ballot P_v on both outcomes. Onto means f is surjective: for every $a \in A$, some profile selects a . Dictatorship of a choice function is the existence of $d \in V$ such that

$$(7) \quad \forall P, a, \quad a \neq f(P) \implies f(P) \succ_{P_d} a.$$

Equivalently, the chosen alternative is always d 's unique top choice. A finite nonempty ballot has such a unique top alternative.

Theorem 2.2 (Finite strict-ranking Gibbard–Satterthwaite theorem). *Under the finite-domain hypotheses above and (1), every onto strategy-proof social choice function is dictatorial in the sense of (7).*

This is `GibbardSatterthwaite.gibbardSatterthwaite`, proved in `GS/Main.lean`. It concerns deterministic single-valued choice without transfers on the full strict-ranking domain. It does not assert an impossibility for restricted domains, randomized rules, weak preferences, or infinite electorates.

3. ARROW THROUGH DECISIVE COALITIONS

For a coalition $G \subseteq V$ and distinct a, b , the implementation distinguishes two properties. *Weak decisiveness* for (a, b) requires $F(P)$ to rank a above b whenever every member of G ranks a above b and every outsider ranks b above a . *Decisiveness for the pair* requires the same social comparison whenever the members agree on $a \succ b$, with no condition on outsiders. A coalition is *decisive* if it is decisive for every ordered pair of distinct alternatives. These are `WeakDecisive`, `DecisivePair`, and `Decisive` in `Arrow/Basic.lean`.

The whole electorate is decisive by unanimity. The selected theorem `Arrow.Palomar.decisiveUniv` exposes this first step. The remaining argument uses field expansion and group contraction.

3.1. Realizing auxiliary profiles. The proof must produce full ballots, not just diagrams involving three alternatives. Given any injective baseline $s_0 : A \rightarrow \mathbb{N}$ and three distinct alternatives a, b, c , `chainFun` assigns scores 0, 1, 2 to a, b, c respectively, and $3s_0(w) + 3$ to every other w . The resulting function is injective: the designated scores are distinct and less than every residual score, while injectivity on the residual alternatives follows from that of s_0 . A baseline is obtained from an equivalence with a finite initial segment of $\mathbb{N}$. The theorem `exists_chain` consequently realizes $a \succ b \succ c$ as a ballot over all of A .

3.2. Field expansion.

Lemma 3.1 (Field expansion). *If F is unanimous and satisfies IIA, and G is weakly decisive for one ordered pair (a, b) , then, given a third distinct alternative, G is decisive for every ordered pair.*

Proof. We describe the first expansion in the source, which is representative of the construction. To prove decisiveness for (a, c) , take an arbitrary profile Q at which every member of G ranks a above c . Construct P with these three-alternative restrictions:

Voters	Restriction of P
$v \in G$	$a \succ b \succ c$
$v \notin G, a \succ_{Q_v} c$	$b \succ a \succ c$
$v \notin G, c \succ_{Q_v} a$	$b \succ c \succ a$

Weak decisiveness gives $a \succ_{F(P)} b$. Unanimity gives $b \succ_{F(P)} c$, so transitivity gives $a \succ_{F(P)} c$. Every voter has the same a, c comparison at P and Q ; IIA transfers the result to Q . Crucially, the outsiders' rankings were selected according to Q . The conclusion is strong pair decisiveness, not merely decisiveness on another polarized profile.

The dual right-expansion construction gives decisiveness for (c, b) , keeping the second alternative fixed. Once these two pairs are decisive, replace the coalition members' ballots by $a \succ c \succ b$ while retaining outsiders' ballots. Transitivity and IIA give strong decisiveness for the original pair (a, b) as well. The source then repeatedly applies these left and right expansion lemmas, with case distinctions when a target alternative coincides with one of the designated alternatives, to reach every ordered pair. No neutrality assumption is introduced. $\square$

The implementation separates `weak_expand_left`, `weak_expand_right`, and `weak_expand_same`, then packages the result as `Arrow.Palomar.fieldExpansion`.

3.3. Group contraction.

Lemma 3.2 (Group contraction). *A decisive coalition with at least two members contains a strictly smaller nonempty decisive coalition.*

Proof. Choose $v_1 \in G$ and put $G_1 = \{v_1\}$ and $G_2 = G \setminus \{v_1\}$. Both are nonempty proper subsets of G . Using three distinct alternatives, construct the profile

Voters	Restriction of P
G_1	$a \succ b \succ c$
G_2	$c \succ a \succ b$
$V \setminus G$	$b \succ c \succ a$

All members of G rank a above b , hence $a \succ_{F(P)} b$. If $b \succ_{F(P)} c$, transitivity yields $a \succ_{F(P)} c$. At P , precisely the members of G_1 favor a over c , while all outsiders favor c over a . IIA extends this social comparison to every profile with that polarized comparison pattern. Thus G_1 is weakly decisive for (a, c) and is fully decisive by Lemma 3.1.

In the other case, $c \succ_{F(P)} b$. At P , the members of G_2 favor c over b and all outsiders favor b over c . IIA makes G_2 weakly decisive for (c, b) ; field expansion again makes it decisive. Completeness of the social ballot supplies the exhaustive case split. $\square$

The selected declaration `Arrow.Palomar.groupContraction` returns $G' \subsetneq G$, a proof that G' is nonempty, and a proof that G' is decisive. Strict containment is used to obtain the strict cardinality decrease.

3.4. From contraction to a dictator. Theorem 2.1 uses strong induction on coalition cardinality, starting with V . Nonemptiness excludes the zero case. A coalition of cardinality one already gives a decisive singleton; a larger one contracts and invokes the induction hypothesis on a smaller cardinality.

If $\{d\}$ is decisive, $a \succ_{P_d} b$ implies $a \succ_{F(P)} b$. Conversely, if the social ballot ranks a above b but d does not, strict completeness of d 's ballot gives $b \succ_{P_d} a$. Decisiveness would then give $b \succ_{F(P)} a$, contradicting asymmetry. Equal alternatives have neither comparison. This proves the full equivalence (5), not just a one-way implication.

4. STRATEGY PROOFNESS AND MONOTONICITY

The reduction begins with a consequence of strategy-proofness. For an alternative a , its lower contour in ballot s is the set of alternatives ranked below a .

Lemma 4.1 (Outcome monotonicity). *Let f satisfy (6). Suppose $f(P) = a$ and*

$$(8) \quad \forall v, b, \quad a \succ_{P_v} b \implies a \succ_{Q_v} b.$$

Then $f(Q) = a$.

Proof. First suppose only voter v changes ballot. If the new outcome is $b \neq a$, strategy-proofness at P excludes $b \succ_{P_v} a$. Therefore $a \succ_{P_v} b$, and (8) gives $a \succ_{Q_v} b$. At the true profile Q , reporting the old ballot would obtain a instead of b , contradicting strategy-proofness at Q .

For the general case, change voters' ballots one at a time. The electorate is finite, so this sequence reaches Q . At each intermediate profile, a voter's ballot is either its original or target ballot, and the same lower-contour preservation holds for the next update. The one-voter argument preserves the outcome throughout. $\square$

These steps are `monotone_one_step` and `monotone` in `GS/Reduction.lean`. The finite update construction uses a list containing all voters.

Lemma 4.2 (Unanimous top choice). *If f is strategy-proof and onto, and a is every voter's top alternative at P , then $f(P) = a$.*

Proof. Onto supplies a profile Q with $f(Q) = a$. Since a is top at P , moving from Q to P preserves every comparison in which a is above another alternative. Apply Lemma 4.1. $\square$

This is `unanimous_of_sp_onto`. Onto is used to obtain the starting profile; strategy-proofness transports its choice. It is not an additional unanimity hypothesis in Theorem 2.2.

5. FROM PAIR TESTS TO A STRICT SOCIAL ORDER

5.1. The top two transformation. For distinct a, b , let $T_{ab}(P)$ put a and b in the first two positions of every ballot, preserving that voter's original comparison between them. Its numerical construction assigns the preferred member score 0, the other score 1, and every other alternative c score $P_v(c) + 2$. Thus

$$(9) \quad a \succ_{T_{ab}(P)_v} b \iff a \succ_{P_v} b,$$

$$(10) \quad a \succ_{T_{ab}(P)_v} c \text{ and } b \succ_{T_{ab}(P)_v} c \quad (c \notin \{a, b\}).$$

The score assignment is injective. The source defines `topTwoProfile` and proves these properties as `topTwo_preserves` and `topTwo_above_rest`. It also proves $T_{ab}(P) = T_{ba}(P)$; writing the pair in the opposite order does not change the ballot transformation.

Lemma 5.1 (Range of a top two test). *For a strategy-proof onto rule, $f(T_{ab}(P)) \in \{a, b\}$.*

Proof. Suppose instead it selects $c \notin \{a, b\}$. Promote a alone to the top of every ballot while keeping the order among other alternatives. Every voter already ranks a above c , so this promotion cannot remove any alternative from c 's lower contour. Monotonicity therefore preserves outcome c . But the new profile has unanimous top choice a , forcing outcome a by Lemma 4.2, a contradiction. $\square$

The analogous construction with three designated alternatives gives a three-top profile whose outcome lies among those three. The proof has the same lower-contour argument. Both range claims use onto through unanimous-top choice.

5.2. Pairwise comparison and independence. Define

$$(11) \quad a \triangleright_P b \iff a \neq b \text{ and } f(T_{ab}(P)) = a.$$

This is `beats f P a b`. It is irreflexive by definition. Symmetry of the top-two transformation and single-valued choice imply asymmetry. Lemma 5.1 supplies totality: for distinct a, b , exactly one of $a \triangleright_P b$ and $b \triangleright_P a$ holds.

Lemma 5.2 (Pairwise independence). *If every voter has the same comparison of a, b at P and Q , then $a \triangleright_P b \iff a \triangleright_Q b$.*

Proof. Suppose $f(T_{ab}(P)) = a$. In the move from $T_{ab}(P)$ to $T_{ab}(Q)$, a remains above all alternatives outside the pair. Its comparison with b is also unchanged by hypothesis and (9). Thus its lower contour is preserved, and monotonicity gives $f(T_{ab}(Q)) = a$. Interchange P and Q for the reverse implication. $\square$

This is `pairwise_IIA`. The residual ballots need not be identical: monotonicity makes their differences irrelevant to this pair test. Onto is not needed for this lemma itself.

5.3. Why there are no cycles. Totality and asymmetry alone allow a tournament with cycles. The source establishes transitivity by using one profile that puts three alternatives jointly above the rest.

For distinct a, b, c , the three-top profile $U_{abc}(P)$ preserves all original comparisons among them. The score of a designated alternative is the number of the other two ranked above it; these scores are 0, 1, 2 in some order. Every other alternative w receives $P_v(w) + 3$. This is an injective ballot, and it preserves each designated pair comparison. The auxiliary lemma `predCount_1t` verifies the elementary three-score comparison used in the construction.

Lemma 5.3 (Transitivity). *If $a \triangleright_P b$ and $b \triangleright_P c$, then $a \triangleright_P c$.*

Proof. The hypotheses imply $a \neq b$ and $b \neq c$; asymmetry rules out $a = c$. Put $Q = U_{abc}(P)$. Pairwise independence transfers the two hypotheses to Q . The three-top range lemma gives $f(Q) \in \{a, b, c\}$.

If $f(Q) = b$, move to $T_{ab}(Q)$. This move preserves b 's lower contour: its relation to a is unchanged and all alternatives outside the pair are now below it. Monotonicity would give $f(T_{ab}(Q)) = b$, contrary to $a \triangleright_Q b$. Similarly $f(Q) = c$ would force $f(T_{bc}(Q)) = c$, contrary to $b \triangleright_Q c$. Therefore $f(Q) = a$.

Moving from Q to $T_{ac}(Q)$ preserves a 's lower contour, so the latter profile selects a . Hence $a \triangleright_Q c$. Pairwise independence, using the preserved a, c comparisons, transfers this conclusion back to P . $\square$

This is `beats_trans` in `GS/SWF.lean`. Its explicit distinctness binders are sufficient because repeated alternatives are excluded by irreflexivity and asymmetry. Consequently $\triangleright_P$ is a strict total order for every profile. This order property is proved before numerical social ranks are introduced.

6. PREDECESSOR COUNTS AND THE APPLICATION OF ARROW

6.1. Encoding the order as a ballot. For each alternative define

$$(12) \quad \text{Pred}_P(a) = \{c \in A : c \triangleright_P a\}, \quad r_P(a) = |\text{Pred}_P(a)|.$$

The implementation calls `r_P copelandScore`, but here it counts predecessors in the already proved strict total order. It does not aggregate voters by a Copeland voting rule, and it does not repair a cyclic relation by scoring it.

Proposition 6.1 (Order representation). *For distinct a, b ,*

$$(13) \quad a \triangleright_P b \iff r_P(a) < r_P(b).$$

In particular, $r_P : A \rightarrow \mathbb{N}$ is injective.

Proof. If $a \triangleright_P b$, transitivity implies $\text{Pred}_P(a) \subseteq \text{Pred}_P(b)$. The containment is strict because $a \in \text{Pred}_P(b)$ but $a \notin \text{Pred}_P(a)$. Finiteness gives $r_P(a) < r_P(b)$. Conversely, if this inequality holds, totality excludes $b \triangleright_P a$, which would give the opposite strict inequality. Hence $a \triangleright_P b$. For injectivity, any two distinct alternatives are ordered one way or the other and so have distinct predecessor counts. $\square$

The strict inequality is proved as `copeland_lt_of_beats`. The injectivity theorem is `copelandScore_injective`. The source then defines

$$(14) \quad F_f(P) = \langle r_P, \text{proof of injectivity} \rangle \in \mathcal{B}(A).$$

This is `swf0fSCF f hSP h0`. The characterization `swf0fSCF_spec` states exactly that the social ballot ranks a above b if and only if $a \triangleright_P b$. At this point, and only at this point, the construction has the required social-welfare codomain.

6.2. The two Arrow axioms. Unanimity of F_f follows directly from a unanimous pair comparison. If every voter ranks a above b at P , then a is top in every ballot of $T_{ab}(P)$. Lemma 4.2 gives $f(T_{ab}(P)) = a$, hence $a \triangleright_P b$ and therefore $a \succ_{F_f(P)} b$ by (13). The code supplies this through `beats_of_unanimous` and `swf0fSCF_unanimous`. Its proof uses an intermediate unanimous-top profile and monotonicity to reach the top-two profile, with the same conclusion.

IIA of F_f is Lemma 5.2 translated through (13), and is proved as `swf0fSCF_IIA`. Equal alternatives are handled separately using irreflexivity. Therefore Theorem 2.1 applies to the actual injective ballot-valued function (14) and yields a voter d whose pairwise ranking agrees with F_f at every profile.

6.3. Transferring the dictator back to choice. Fix P and put $a = f(P)$. Suppose some $b \neq a$ is ranked above a by d . Arrow dictatorship and the social-ranking characterization imply $b \triangleright_P a$, so

$$f(T_{ba}(P)) = b.$$

On the other hand, moving from P to $T_{ba}(P)$ preserves a 's lower contour. Its comparison with b is retained, and all other alternatives are below a in the new profile. Monotonicity, starting from $f(P) = a$, therefore gives

$$f(T_{ba}(P)) = a,$$

a contradiction. Thus no alternative is above $f(P)$ in d 's ballot. Strict completeness gives (7) for every other alternative. This proves Theorem 2.2. No extra onto assumption is imposed on the constructed welfare function, and no restriction to top-two profiles is imposed on the final dictatorship conclusion.

7. SOURCE STRUCTURE AND VERIFICATION BOUNDARY

7.1. Modules and dependency identity. The Arrow proof is organized into `Arrow/Basic.lean`, `Arrow/FieldExpansion.lean`, `Arrow/GroupContraction.lean`, and `Arrow/ArrowTheorem.lean`. The Gibbard–Satterthwaite repository vendors those same four files. A byte comparison at the two stated pins finds them identical. Its Lake file declares a local `Arrow` library alongside the `GibbardSatterthwaite` library; `Arrow` is not a separate fetched Lake package.

The extension's dependency chain is `GS/Basic.lean`, `GS/Reduction.lean`, `GS/SWF.lean`, and `GS/Main.lean`. The first reuses `Arrow.Basic`; the last imports both `GS.SWF` and `Arrow.ArrowTheorem`. Although the README broadly locates the reduction in `GS/Reduction.lean`, that file proves monotonicity and unanimous-top choice. The social order and rank construction are in `GS/SWF.lean`; the Arrow axioms, invocation, and dictatorship transfer are in `GS/Main.lean`. This division follows the source files, not the abbreviated README description.

Both manifests pin Mathlib to

065356127b1dc0016f66b7283ce0ce2c4055aa55.

The remaining manifest packages are Mathlib’s supporting Lean packages. The vendored Arrow copy and the Mathlib dependency have different roles: the former is substantive source developed by the same maintainer; the latter supplies the general finite-set, order, equivalence, and tactic infrastructure.

7.2. Challenge and solution separation. Each `Challenge.lean` is a self-contained statement module importing only `Mathlib.Data.Finset.Basic` and `Mathlib.Data.Fintype.EquivFin`. Arrow’s Challenge contains four deliberate statement holes, one per selected theorem; Gibbard–Satterthwaite’s contains one. These are the problems to be compared with the implementations, not proofs of the results.

The GS Challenge defines its own support vocabulary for ballots and profiles, while the implementation reuses `Arrow.Basic`. Its comparator selects the four choice concepts and the final theorem; it does not separately select every support definition. Arrow’s comparator selects nine definitions and four theorems; `WeakDecisive` occurs in its statement but is not a separately selected definition.

Each `Solution.lean` imports the final implementation module. Neither solution imports its Challenge. The selected proved declarations are reached through that implementation import closure. At the inspected pins, a lexical scan of the substantive Arrow and GS modules and both solution wrappers finds no proof placeholders, custom axiom declarations, or `unsafe` declarations. This scan is a source observation, not a substitute for elaboration or a transitive axiom audit.

The registry configurations permit `propext`, `Quot.sound`, and `Classical.choice`. The source uses classical selection, including choosing a finite enumeration and top alternatives. The paper therefore makes no claim of a constructive or executable decision algorithm extracted from these proofs. The repositories contain axiom-audit scripts, but their presence alone does not establish that a new audit has been run.

7.3. Historical checks and present checks. Palomar records historical verification on September 25, 2026, under the stated Lean toolchain. The Arrow record reports verification at 18:26:09 UTC and registration at 19:02:37 UTC, with workflow run [36171797049](#). The Gibbard–Satterthwaite record reports verification at 20:43:00 UTC and registration at 21:11:21 UTC, with workflow run [36187074973](#). Both records include comparator configuration, permitted axioms, dependency pins, file hashes, preservation receipts, and independent-kernel tool identities [4, 5].

For this manuscript we retrieved both exact repository snapshots, read the substantive proofs and theorem interfaces, compared the vendored Arrow files, checked manifest and toolchain identities, and matched both Challenge and both Solution SHA-256 hashes against the immutable registry records. We also reviewed the mathematical exposition and bibliographic references independently of the initial drafting pass. We did not execute a fresh Lean build or a fresh kernel/axiom audit in this preparation environment, where Lean and Lake were unavailable. The September 25 verification is historical evidence associated with the pinned artifacts. Compilation of the present L^AT_EX manuscript, source comparison, and mathematical source review are distinct checks and are not described as fresh formal verification or human peer review.

8. ATTRIBUTION AND LIMITATIONS

The source mathematics belongs to Arrow, Gibbard, and Satterthwaite [1, 2, 10]. Earlier machine-checked work includes Nipkow’s Isabelle/HOL development of both theorems [3]. The independent Roos record concerns a three-candidate Fourier-analytic Arrow proof, whereas the inspected Arrow artifact follows decisive coalitions for any finite alternative set with three distinct members [9, 4]. These are differences in proof route and formal statement, not a claim that the current development originated the general theorem.

The pinned repository and registry metadata credit Arthur Freitas Ramos as the author and responsible maintainer of the Lean developments. That source attribution is separate from the three-author byline of this manuscript. Both repositories are BSD-3-Clause; Mathlib is a dependency

with its own Apache-2.0 license. The manuscript’s CC BY 4.0 license does not relicense those source artifacts. The bibliography cites and links the existing work; the prose and displayed proofs here are a newly prepared exposition rather than a republication of a supplied journal manuscript.

One source-metadata correction is recorded explicitly. The pinned Gibbard–Satterthwaite metadata cites Satterthwaite’s intended 1975 title with DOI 10.1016/0022-0531(75)90002-6. The publisher identifies that title as 10.1016/0022-0531(75)90050-2, which is the DOI used in our bibliography [10]. This bibliographic correction changes neither the pinned code nor its theorem statements. Likewise, a dated search that found no earlier Gibbard–Satterthwaite entry in Palomar would not establish absence of earlier formalizations elsewhere; we make no such inference.

AI assistance. This manuscript was prepared with GPT-6.1 assistance for exposition, source alignment, and review. The formalization metadata separately records proof-engineering assistance by `gpt-6-luna` via the Codex CLI. These are distinct provenance claims: the manuscript-generation model is not substituted for the model recorded by the source repositories. AI-assisted review and mechanical proof checking do not establish comprehensive human understanding, source-author endorsement, or independent human peer review.

The verified scope is finite strict-ranking social choice. Broader variants with weak preferences or infinite electorates require different statements and are not supplied by these artifacts. The value of the present development is its explicit proof interface: auxiliary ballots satisfy the domain, coalition contraction terminates by finite cardinality, the strategy-proof reduction really constructs an injective social ballot, and the final conclusions match the declared pairwise and top-choice notions of dictatorship.

REFERENCES

1. Kenneth J. Arrow, *A difficulty in the concept of social welfare*, Journal of Political Economy **58** (1950), no. 4, 328–346, <https://doi.org/10.1086/256963>.
2. Allan Gibbard, *Manipulation of voting schemes: A general result*, Econometrica **41** (1973), no. 4, 587–601, <https://doi.org/10.2307/1914083>.
3. Tobias Nipkow, *Social choice theory in HOL: Arrow and Gibbard–Satterthwaite*, Journal of Automated Reasoning **43** (2009), no. 3, 289–304, <https://doi.org/10.1007/s10817-009-9147-4>.
4. Palomar Registry, *Arrow’s impossibility theorem (general finite version)*, 2026, PALOMAR-2026-09-25-000024, version 1. Registered September 25, 2026 <https://palomar-registry.org/entry?id=PALOMAR-2026-09-25-000024&version=1>.
5. ———, *Gibbard–Satterthwaite theorem (strategy-proof social choice)*, 2026, PALOMAR-2026-09-25-000028, version 1. Registered September 25, 2026 <https://palomar-registry.org/entry?id=PALOMAR-2026-09-25-000028&version=1>.
6. Dominik Peters, *SocialChoiceLean: Gibbard–Satterthwaite in Lean*, 2026, Public repository snapshot, July 21, 2026. Consulted as source; not rebuilt for this manuscript <https://github.com/DominikPeters/SocialChoiceLean/tree/94a4c650b6a3ef14df801a613c3b46169dbd754d>.
7. Arthur Freitas Ramos, *Arrow’s impossibility theorem in Lean 4*, 2026, Pinned repository snapshot. BSD-3-Clause <https://github.com/Arthur742Ramos/arrow-impossibility-lean/tree/4f405d27d0b244574bbee5d3d79bae0c66bda16e>.
8. ———, *Gibbard–Satterthwaite theorem in Lean 4*, 2026, Pinned repository snapshot. BSD-3-Clause <https://github.com/Arthur742Ramos/gibbard-satterthwaite-lean/tree/8398e65a99cd3d5e973c64963c4bab8c0491947c>.
9. Joris Roos, *Arrow’s theorem via Fourier analysis*, 2026, Palomar record PALOMAR-2026-09-01-000011, version 1 <https://palomar-registry.org/entry?id=PALOMAR-2026-09-01-000011&version=1>.
10. Mark A. Satterthwaite, *Strategy-proofness and Arrow’s conditions: Existence and correspondence theorems for voting procedures and social welfare functions*, Journal of Economic Theory **10** (1975), no. 2, 187–217, [https://doi.org/10.1016/0022-0531\(75\)90050-2](https://doi.org/10.1016/0022-0531(75)90050-2).