

A LEAN FORMALIZATION OF THE SHAPLEY VALUE CHARACTERIZATION FOR FINITE COOPERATIVE GAMES

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

ABSTRACT. We describe a Lean 4 formalization of the standard four-axiom characterization of the Shapley value for finite transferable-utility cooperative games. Games are arbitrary real-valued characteristic functions normalized at the empty coalition; the player type may itself be empty. The factorial coalition formula is proved efficient, symmetric for interchangeable players, null-player respecting, and additive. Uniqueness follows from the values of arbitrarily scaled unanimity games and an explicit Boolean-lattice Möbius decomposition. The proof requires neither a real-homogeneity axiom nor a continuity assumption on competing allocation rules. We explain the finite-sum arguments, the precise Lean statements, the separation between the registry’s challenge and implemented solution, and the historical verification evidence for the pinned source revision. The contribution is an auditable exposition of an existing formalization of classical mathematics, with no claim of new mathematics or formalization priority.

1. CONTEXT AND CONTRIBUTION

The Shapley value assigns a payoff to each player of a cooperative game by weighting that player’s marginal contributions to coalitions. Shapley’s classical work introduced a value characterized axiomatically [7]; subsequent developments and applications are collected in Roth’s edited volume [6]. This paper concerns the standard modern formulation using efficiency, symmetry, the null-player property, and additivity. It does not present that quartet as a literal transcription of the original paper’s axiom list.

The subject here is the Lean 4 and Mathlib development [1, 8, 4] registered as PALOMAR-2026-09-25-000008, version 1 [3]. Every source claim refers to commit `ee15891fecdf294363791c61a5ec018109fe423a`. The registered source credits Arthur Freitas Ramos as its author and responsible maintainer. The three-author byline above is the byline of this explanatory manuscript; it does not change the registered source attribution.

The development has two mathematically substantive parts. First, direct finite-sum manipulations show that the factorial formula satisfies all four axioms. In particular, efficiency is established by cancellation of coalition coefficients, rather than by a formal permutation-average representation. Second, unanimity games and a Boolean-lattice inversion formula show that any rule satisfying the axioms has a prescribed value on every game. The uniqueness proof is careful about the difference between additivity and real linearity: it determines arbitrary real multiples of unanimity games directly from the other three axioms.

Our contribution is to make the formal artifact’s exact scope, proof architecture, and evidence boundary accessible without requiring the reader to infer them from Lean syntax. The mathematical characterization and inversion method are classical. Neither registration

Date: October 1, 2026.

2020 Mathematics Subject Classification. Primary 91A12; Secondary 68V20.

Key words and phrases. Shapley value, transferable utility, cooperative games, axiomatic characterization, unanimity games, Möbius inversion, Lean 4, Mathlib.

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. Licensed under Creative Commons Attribution 4.0 International (CC BY 4.0).

nor the present exposition establishes a priority claim, independent human review of every proof term, or an endorsement by the authors of the mathematical sources.

2. GAMES AND THE EXACT AXIOMS

Let N be a finite player type with decidable equality, and write $n = |N|$. Lean expresses these assumptions as `[FintypeN]` and `[DecidableEqN]`. Coalitions are finite sets of players, represented by `FinsetN`. The grand coalition is `Finset.univ`.

Definition 2.1. A game is a function $v : \mathcal{P}(N) \rightarrow \mathbb{R}$ satisfying $v(\emptyset) = 0$. Write $\mathcal{G}(N)$ for the set of these games. Addition and real scalar multiplication are pointwise.

The source definition `ShapleyValue.Game` bundles the function with its normalization proof as a subtype. It does not impose positivity, monotonicity, superadditivity, convexity, or an assumption that N is nonempty. Although the game type itself can be defined without a finite player assumption, the Shapley formula and the characterization use finite players throughout.

An allocation rule is an arbitrary function $\psi : \mathcal{G}(N) \rightarrow (N \rightarrow \mathbb{R})$. It need not initially be continuous, measurable, computable, or linear. Its four properties are as follows.

Definition 2.2. The rule ψ is:

- (1) *efficient* if $\sum_{i \in N} \psi_i(v) = v(N)$ for every game v ;
- (2) *symmetric* if, for every v and $i, j \in N$,

$$[\forall S \subseteq N, i, j \notin S \implies v(S \cup \{i\}) = v(S \cup \{j\})] \implies \psi_i(v) = \psi_j(v);$$
- (3) *null-player respecting* if, for every v and i ,

$$[\forall S \subseteq N, i \notin S \implies v(S \cup \{i\}) = v(S)] \implies \psi_i(v) = 0;$$
- (4) *additive* if $\psi_i(v + w) = \psi_i(v) + \psi_i(w)$ for all v, w, i .

These are exactly the predicates `Efficient`, `Symmetric`, `NullPlayer`, and `Additive` in the `ShapleyValue` namespace. Symmetry is a within-game condition on interchangeable players; the interface does not posit a separate cross-game relabeling axiom. The null-player property is the zero-marginal condition stated above, not a separately assumed general dummy-player payment formula.

For $1 \leq k \leq n$, define

$$(1) \quad w_n(k) = \frac{(k-1)!(n-k)!}{n!}.$$

The Shapley formula implemented in `shapleyValue` is

$$(2) \quad \phi_i(v) = \sum_{\substack{S \subseteq N \\ i \in S}} w_n(|S|)(v(S) - v(S \setminus \{i\})).$$

Factorials are natural numbers cast to $\mathbb{R}$. The implementation defines `weight` for every natural k using truncated natural subtraction, but only $1 \leq k \leq n$ occurs in (2). The denominator $n!$ is nonzero even when $n = 0$. In the coefficient identities below, w_n denotes this total extension, including $k = 0$ and $k = n + 1$.

Theorem 2.3 (Registered characterization). *For every finite player type N and allocation rule ψ ,*

$$\psi \text{ satisfies the four properties in Definition 2.2} \iff \psi = \phi.$$

The public declaration is `ShapleyValue.Palomar.shapley_characterization`. Equality is equality of functions on every game and every player, not agreement only on a restricted class of games. If $N = \emptyset$, every payoff vector is the unique empty function, efficiency is $0 = v(\emptyset)$, and the remaining player-indexed conditions are vacuous. The source retains this case instead of imposing a nonempty-player hypothesis.

3. THE FACTORIAL FORMULA SATISFIES THE AXIOMS

3.1. Null players and additivity. If i is null and $i \in S$, apply the null hypothesis to $S \setminus \{i\}$ to obtain $v(S) = v(S \setminus \{i\})$. Every summand in (2) is therefore zero. For additivity, pointwise game addition gives

$$(v + w)(S) - (v + w)(S \setminus \{i\}) = (v(S) - v(S \setminus \{i\})) + (w(S) - w(S \setminus \{i\})).$$

Distributivity and finite-sum additivity finish the argument. These are `shapleyValue_nullP` layer and `shapleyValue_additive` in `ShapleyValue/Existence.lean`.

3.2. Symmetry by coalition reindexing. The insertion–erasure bijection rewrites the formula as

$$(3) \quad \phi_i(v) = \sum_{R \subseteq N \setminus \{i\}} w_n(|R| + 1)(v(R \cup \{i\}) - v(R)).$$

Suppose $i \neq j$ are interchangeable. Split each sum according to whether the other player belongs to R . Coalitions excluding both players give equal contributions by the symmetry hypothesis. In the other parts, write the coalitions as $R \cup \{j\}$ and $R \cup \{i\}$, respectively, where R excludes both. The common upper coalition is $R \cup \{i, j\}$, the lower worths are equal, and the weights agree because the cardinalities agree. Thus $\phi_i(v) = \phi_j(v)$. The case $i = j$ is immediate.

The Lean proof `shapleyValue_symmetric` follows precisely this split. Its helper lemmas establish the insertion–erasure bijections as finite-sum reindexings, including their membership and inverse conditions.

3.3. Efficiency by weight cancellation. For $1 \leq k < n$, the factorial identities give

$$(4) \quad k w_n(k) = (n - k) w_n(k + 1).$$

When $n > 0$, they also give $n w_n(n) = 1$. These are the source lemmas `weight_balance` and `weight_top`.

Sum (2) over players and interchange the finite sums. Each $v(S)$ occurs $|S|$ times as an upper-coalition worth. For lower-coalition occurrences, set $T = S \setminus \{i\}$, so that $i \notin T$ and $S = T \cup \{i\}$. There are $n - |T|$ choices of such i . Consequently,

$$(5) \quad \sum_{i \in N} \phi_i(v) = \sum_{T \subseteq N} [|T| w_n(|T|) - (n - |T|) w_n(|T| + 1)] v(T).$$

Every nonempty proper-coalition coefficient vanishes by (4). The empty-coalition term vanishes by normalization, regardless of the value of its coefficient. If $n > 0$, the grand-coalition coefficient equals one, giving $v(N)$. If $n = 0$, the only coalition is empty and both sides are zero.

The implementation `shapleyValue_efficient` makes all these boundary cases explicit. It uses `sum_over_members`, `sum_over_nonmembers`, and `sum_filter_insert` to obtain (5), rather than asserting an unproved interchange or a counting identity. This proves existence of an allocation rule satisfying the four axioms before uniqueness is invoked.

4. UNANIMITY GAMES AND UNIQUENESS

4.1. Scaled unanimity games without a homogeneity axiom. For nonempty $T \subseteq N$, the unanimity game u_T is

$$u_T(S) = \begin{cases} 1, & T \subseteq S, \\ 0, & T \not\subseteq S. \end{cases}$$

The Lean definition `unanimityGame` is total on all coalitions: $u_\emptyset$ is defined to be the zero game, so that normalization is never violated.

Lemma 4.1. *If ψ is efficient, symmetric, and null-player respecting, then for every nonempty T , every real c , and every player i ,*

$$(6) \quad \psi_i(cu_T) = \begin{cases} c/|T|, & i \in T, \\ 0, & i \notin T. \end{cases}$$

Proof. Every player outside T is null in cu_T , hence receives zero. Members of T are interchangeable: for two distinct members and a coalition excluding both, inserting only one cannot complete T . Their payoffs are therefore equal. Efficiency and $(cu_T)(N) = c$ give $|T|$ times the common payoff equal to c . Since T is nonempty, division by $|T|$ is valid. The argument also covers $c = 0$ and $c < 0$. $\square$

This is `valueOn_unanimity`. Notably, its assumptions do not include additivity. It does not infer $\psi(cu_T) = c\psi(u_T)$ from additivity: such an inference for arbitrary real c would require justification. Instead, (6) is established directly for each scaled game. The later uniqueness proof needs only additivity over a finite sum.

4.2. Boolean-lattice inversion. Define the coefficient

$$(7) \quad m_v(T) = \sum_{S \subseteq T} (-1)^{|T|-|S|} v(S),$$

implemented as `mobiusCoeff`. This is the Boolean-lattice Möbius transform; incidence-algebra inversion is treated generally by Rota [5].

Lemma 4.2. *For every normalized game v and coalition Q ,*

$$(8) \quad v(Q) = \sum_{\emptyset \neq T \subseteq Q} m_v(T).$$

Consequently, as games,

$$(9) \quad v = \sum_{\emptyset \neq T \subseteq N} m_v(T) u_T.$$

Proof. Expand the right side of (8) using (7) and interchange the two finite sums. For a nonempty $S \subseteq Q$, its worth $v(S)$ has coefficient

$$\sum_{S \subseteq T \subseteq Q} (-1)^{|T|-|S|} = \sum_{U \subseteq Q \setminus S} (-1)^{|U|},$$

using the bijection $T \mapsto T \setminus S$, with inverse $U \mapsto S \cup U$. This alternating sum is one if $S = Q$ and zero otherwise. In the latter case $Q \setminus S$ is nonempty, and the sum vanishes by the binomial identity $(1 - 1)^{|Q \setminus S|} = 0$. The $S = \emptyset$ term vanishes because $v(\emptyset) = 0$. If Q is empty, the sum in (8) is empty and the identity again follows from normalization. Evaluating the right side of (9) at Q selects exactly its nonempty subsets, so (8) proves equality of games. $\square$

The source isolates the alternating cancellation in `alt_sum_powerset` and `inner_mobius`. The latter assumes a nonempty S , $S \subseteq Q$, and $S \neq Q$; empty S is handled separately by normalization. The resulting pointwise and game-level theorems are `mobius_pointwise` and `mobius_decomposition`. Thus no coefficient is silently divided by a zero coalition size.

4.3. Determination of every allocation. Additivity first implies $\psi_i(0) = 0$, since $\psi_i(0) = \psi_i(0) + \psi_i(0)$. Induction then extends it to any finite sum; this is `additive_sum`. Apply it to (9) and use Lemma 4.1 for each summand to obtain

$$(10) \quad \psi_i(v) = \sum_{\substack{\emptyset \neq T \subseteq N \\ i \in T}} \frac{m_v(T)}{|T|}.$$

The theorem `value_eq_of_axioms` proves this formula for every rule satisfying the four properties.

The rule ϕ satisfies those properties by Section 3, so it has the same expression (10). Any other such ψ therefore agrees with ϕ at every game and player. Function extensionality gives `shapleyValue_unique`. Its converse, substituting $\psi = \phi$ and collecting the four existence results, proves Theorem 2.3. This proof uses no continuity axiom and does not assume a real-linear structure on the competing rule.

Remark 4.3 (An illustrative three-player calculation). Let $N = \{a, b, c\}$ and write $v_{ab} = v(\{a, b\})$, and similarly for other coalitions. For player a , formula (2) reads

$$\phi_a(v) = \frac{1}{3}v_a + \frac{1}{6}(v_{ab} - v_b) + \frac{1}{6}(v_{ac} - v_c) + \frac{1}{3}(v_{abc} - v_{bc}).$$

Meanwhile (7) gives $m_v(\{a\}) = v_a$, $m_v(\{a, b\}) = v_{ab} - v_a - v_b$, and $m_v(N) = v_{abc} - v_{ab} - v_{ac} - v_{bc} + v_a + v_b + v_c$. Substitution into (10) gives the same expression. This calculation illustrates the two representations; it is explanatory algebra, not an additional registered theorem or source benchmark.

5. THE FORMAL INTERFACE AND ITS EVIDENCE BOUNDARY

5.1. Implementation and challenge. The substantive source modules have distinct roles:

- `ShapleyValue/Basic.lean`: games, operations, factorial weight, the Shapley formula, and the four predicates;
- `ShapleyValue/Existence.lean`: finite-sum reindexing, weight cancellation, and the four properties of the formula;
- `ShapleyValue/Uniqueness.lean`: the additive game structure, unanimity games, inversion, determination of values, and characterization;
- `Solution.lean`: the six public wrapper theorems in the `ShapleyValue.Palomar` namespace.

Besides the characterization and uniqueness theorems, the exported wrappers are `shapleyValue_efficient`, `shapleyValue_symmetric`, `shapleyValue_nullPlayer`, and `shapleyValue_additive`, all under that namespace.

`Challenge.lean` is a separate specification module. It repeats the seven compared definitions and six theorem statements, each theorem ending in an intentional `sorry`. The solution imports the substantive uniqueness module, not the challenge. Compiling the challenge alone cannot establish the six results, and its six statement holes must not be mistaken for holes in the solution proof dependency chain. The comparator configuration fixes the compared definitions and theorems and permits only `propext`, `Quot.sound`, and `Classical.choice` as axioms.

5.2. Pinned environment and historical verification. The authoritative `lean-toolchain` selects `leanprover/lean4:v4.35.0-rc2`, and `lake-manifest.json` pins `Mathlib` to `065356127b1dc0016f66b7283ce0ce2c4055aa55`, together with eight transitive packages. Reproduction should preserve the complete manifest. The README’s older reference to Lean 4.33.0 is inconsistent with this pin and is not the environment used for the registered verification. The separate local comparator script also has older-toolchain comments and pins; its compatibility should be checked before using it for a new replay.

The registry records successful verification on September 24, 2026, followed by registration on September 25. The official mechanical report from workflow run `36005001365` was retrieved during manuscript preparation [2]. It identifies the exact source commit, challenge and solution hashes, toolchain, dependencies, and permitted axioms. It reports a successful solution build and acceptance by Lean’s default kernel and by the `con-ron` and `NanoDa` kernels. Its status is `pass`, with no listed errors or warnings. The comparator’s kernel-acceptance statement concerns the solution checked against that configured challenge surface; it is historical machine-checking evidence, not a fresh run performed for this manuscript.

The three permitted foundations are propositional extensionality, quotient soundness, and classical choice. These are standard classical Lean foundations, not the four economic axioms, which are explicit hypotheses on ψ . A permitted-axiom list bounds what may occur in the checked proof dependency closure; by itself it does not say that each theorem uses all three. The pinned `scripts/AxiomAudit.lean` prints axiom dependencies for twelve declarations, consisting of the six library theorems and their six wrappers. Its checker requires exact coverage and rejects axioms outside the configured set. We inspected this audit’s source, but do not claim a new per-declaration axiom execution.

5.3. Checks performed for this exposition. Fresh source retrieval on October 1, 2026 checked all 22 repository blobs against the Git tree at the registered commit. The implementation modules and solution contain no visible `sorry`, `admit`, or new axiom declaration in this static inspection. The challenge contains exactly the six deliberate statement holes described above. Challenge, solution, manifest, and configuration hashes agree with the official mechanical report. A separate AI-based review checked the mathematical exposition against the pinned declarations and proof organization, and the bibliography against primary-source publication records.

No fresh Lean build, kernel replay, or executed axiom audit was performed while preparing this manuscript. Static scans cannot replace elaboration and transitive proof-term checking. The proofs in Sections 3 and 4 are mathematical explanations of the source architecture, not an alternative verification system. Likewise, checking a theorem proves a proposition in the specified definitions; it does not validate an economic application whose assumptions differ from those definitions.

6. SCOPE AND PREPARATION PROVENANCE

The formal artifact covers finite normalized real-valued characteristic functions and the specified four-axiom characterization. It does not formalize infinite-player games, nontransferable utility, a general permutation-average theorem, computational complexity, numerical approximation, or application-specific guarantees for attribution methods. The Shapley formula and associated constructions are marked `noncomputable`; finite mathematical sums are not presented here as an extracted executable implementation.

The pinned source metadata reports AI-assisted proof engineering with `gpt-6-luna` (Code xCLI). That historical disclosure concerns the formal development. This manuscript was prepared with GPT-6.1 assistance and its text is primarily AI-generated. The submitting contributor reports some human understanding of its content. These statements do not assert that every listed manuscript author personally reviewed every proof term. The independent source and mathematical review and the document checks have the limited scope stated above.

The Lean repository remains licensed BSD-3-Clause. The CC BY 4.0 license of this newly written manuscript is separate and does not relicense the repository. No source-repository change or new registry verification is implied by preparation of this exposition.

REFERENCES

1. Leonardo de Moura and Sebastian Ullrich, *The Lean 4 theorem prover and programming language*, Automated Deduction—CADE 28, Lecture Notes in Computer Science, vol. 12699, Springer, 2021, https://doi.org/10.1007/978-3-030-79876-5_37, pp. 625–635.
2. Palomar Registry, *Mechanical verification of the Shapley characterization*, GitHub Actions workflow run 36005001365, 2026, Verified September 24, 2026. <https://github.com/PalomarRegistry/PalomarSubmission/actions/runs/36005001365>.
3. ———, *Shapley value axiomatic characterization*, Registered formalization, PALOMAR-2026-09-25-000008, version 1, 2026, Registered September 25, 2026. <https://palomar-registry.org/entry?id=PALOMAR-2026-09-25-000008&version=1>.

4. Arthur Freitas Ramos, *Shapley value axiomatic characterization in Lean 4*, Source repository, registered revision, 2026, Commit `ee15891fecdf294363791c61a5ec018109fe423a`. <https://github.com/Arthur742Ramos/shapley-value-lean/tree/ee15891fecdf294363791c61a5ec018109fe423a>.
5. Gian-Carlo Rota, *On the foundations of combinatorial theory I. theory of Möbius functions*, Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete **2** (1964), 340–368, <https://doi.org/10.1007/BF00531932>.
6. Alvin E. Roth (ed.), *The Shapley value: Essays in honor of Lloyd S. Shapley*, Cambridge University Press, 1988, <https://doi.org/10.1017/CB09780511528446>.
7. Lloyd S. Shapley, *A value for n -person games*, Contributions to the Theory of Games II (Harold W. Kuhn and Albert W. Tucker, eds.), Annals of Mathematics Studies, vol. 28, Princeton University Press, 1953, <https://doi.org/10.1515/9781400881970-018>.
8. The mathlib Community, *The Lean mathematical library*, Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, ACM, 2020, <https://doi.org/10.1145/3372885.3373824>, pp. 367–381.