Journal of Integrated Knowledge
Vol. 2 No. 7 (2026)

Task Counts and Rotation-Aware Workload Give Different Credit Distributions in Five Robot-Planning Missions

Ian Scott[1] · Jonathan Wells[2] · Paul Lewis[3]

[1] Portugal · [2] Spain · [3] Austria

Published: July 08, 2026 | Article 866

Abstract

A team completion total gives every finished errand equal weight, although errands require different amounts of movement. We reconstruct five original robot-planning missions with fixed task queues, 2,100 agents and 24,188 completions. For each completed task, an exact orientation-aware graph search supplies an individual action lower bound. Replacing unit completion credit with this workload reduces the finite Gini index in every mission. The equal-mission difference is -0.03510; deleting one mission gives means from -0.03873 to -0.03117. These are finite composition values. A complete action ledger also retains 2,100 unfinished tasks, whose observed exposure reaches 27.44% of one mission's budget. The findings distinguish observed credit units without identifying fair allocation, causal contribution or a superior policy.

Task Counts and Rotation-Aware Workload Give Different Credit Distributions in Five Robot-Planning Missions

Abstract

A team completion total gives every finished errand equal weight, although errands require different amounts of movement. We reconstruct five original robot-planning missions with fixed task queues, 2,100 agents and 24,188 completions. For each completed task, an exact orientation-aware graph search supplies an individual action lower bound. Replacing unit completion credit with this workload reduces the finite Gini index in every mission. The equal-mission difference is -0.03510; deleting one mission gives means from -0.03873 to -0.03117. These are finite composition values. A complete action ledger also retains 2,100 unfinished tasks, whose observed exposure reaches 27.44% of one mission's budget. The findings distinguish observed credit units without identifying fair allocation, causal contribution or a superior policy.

1. Counting a team's work leaves an individual measurement choice

Robot-planning benchmarks often summarize collective performance by the number of errands finished before a simulation ends. The League of Robot Runners makes this objective concrete through repeated goal assignments and coordinated robot movements (Chan et al., 2024). An individual agent can contribute several short errands or fewer long ones. Giving every completion one credit unit preserves the team count, but it does not express the action work represented by each completion. The distribution of individual credits therefore depends on what the evaluator chooses to count.

This issue is separate from finding a valid joint route. Multi-agent pathfinding has multiple conflict conventions and objective functions, which must be specified when comparing results (Stern et al., 2019). Once a joint trace is fixed, a second question remains: which attributes of its individual participants should enter a contribution summary? Measurement theory connects numerical assignments to the attribute being represented (Hand, 2004). Construct-validity analysis likewise cautions against giving an operational quantity a broader meaning than its assumptions support (Jacobs and Wallach, 2021).

Our research question is how the observed distribution changes when individual completion credits are weighted by rotation-aware shortest-action workload, and how that completed workload fits within the entire team action horizon. The behavior, task queues and source solutions remain fixed. We calculate a signed difference between two concentration indices and reconcile every agent's elapsed action time with an exhaustive ledger. A negative difference would mean workload credit is less concentrated than task-count credit; a positive difference would mean the reverse. Neither direction is assumed to indicate better coordination or fairer treatment.

2. Five original missions provide executed actions and predetermined task queues

We use every Random-domain instance in the official 2023 Main Round best-known-solution category: MR23-I-03, I-04, I-05, I-07 and I-08 (League of Robot Runners, n.d.-a). The pinned Benchmark-Archive revision is 1adae003763d53eb4100a59ce4bcbc0c7212eae0. Original compressed solutions, problem definitions, agent starts, task queues and the shared map were downloaded and hashed. The acquired cohort and historical source specification comprise twenty-eight files and 3,827,437 downloaded bytes. Repository Git blob hashes, ZIP integrity checks and extracted-member SHA256 values all agree.

The five missions contain 2,100 agent records and 2,750,000 executed agent-action cells. They share one 32 by 32 map with 819 traversable cells. Team sizes and horizons differ, as Table 1 records. These

quantities describe the released cohort; action cells and agent identities do not supply independent experiments. The archive selects best-known outputs and names the producing teams, implementation revisions and submission identifiers. It is not a randomized comparison of algorithms, loads or planning horizons, and selection of successful source submissions remains part of its provenance.

Every selected problem sets one revealed task per agent and uses round-robin assignment. An agent's next goal follows its predetermined queue when the current task finishes. The executed planner coordinates movements; the inputs do not give it unrestricted choice of which tasks to assign. Every recorded assignment was checked against the corresponding original queue. Global task identifiers reflect reveal order, so they are not substituted for input-file row numbers. These joins distinguish an actual individual assignment history from an aggregate team score.

Table 1. *The complete five-mission cohort. H is the number of executed action steps. AllValid describes submitted planner actions, rather than a physical-safety measurement. The flagged mission remains included.*

Mission	Agents	H	Executed agent-action cells	Team completions	Source AllValid	Source errors
MR23-I-03	200	500	100000	3042	No	1
MR23-I-04	100	500	50000	1741	Yes	0
MR23-I-05	400	1000	400000	7432	Yes	0
MR23-I-07	600	1000	600000	5914	Yes	0
MR23-I-08	800	2000	1600000	6059	Yes	0

The historical Start-Kit revision cdd0356af57a394d4d242a5cc4896e4086b3df31 defines the discrete action and event clock (League of Robot Runners, n.d.-b). Forward movement, clockwise rotation, counterclockwise rotation and waiting each occupy one action step. We use actual executed paths, without replacing them by proposed planner paths or treating steps as measured seconds. The source exporter stores task events as task identifier, timestep and description. Its Markdown documentation reverses the first two fields in one description. We resolved this discrepancy through the exporter and by matching every finish event to the replayed goal position.

MR23-I-03 reports AllValid as No and records an incorrect-vector-size error at step one. The historical move routine replaces an invalid submitted joint action with an all-agent wait. The executed first-step waits are present in this trace and remain in its elapsed budgets. Our separate audit finds no executed vertex conflict, edge swap, obstacle entry or boundary violation in any of the five missions under the declared grid rules. That computational result does not remove the source error, authenticate an exact historical executable or establish physical robot safety.

The Benchmark-Archive repository has no named license file, and its recorded license metadata is null. Its official public release invites research comparison, but we do not infer unrestricted raw redistribution rights. Originals stay local; this manuscript reports attributed aggregate calculations and retains source locators. The historical Start-Kit code has an MIT notice, which does not automatically license the benchmark data. Documenting these distinctions, along with source selection and missing metadata, follows the purpose of dataset documentation described by Gebru et al. (2021).

3. A completed-task graph and a terminal ledger measure different things

The protocol was frozen before the first shortest-path or workload calculation. Published totals, source flags and raw event examples were already visible during source validation, so no outcome-blinding claim is made. For each finished task, we reconstruct the assignment cell and heading, the goal and its recorded finish step. The static directed graph contains each traversable cell crossed with four headings. Forward movement enters a free adjacent cell; either turn changes heading in place. Every edge costs one action, and any heading at the goal is accepted.

Grid path-search methods require explicit rules for admissible transitions and costs (Lee, 1961). A valid observed route also differs conceptually from a shortest path in a specified graph (Dijkstra, 1959). Our reverse breadth-first search begins at all four goal headings and returns an exact individual lower bound, ignoring other agents. It therefore includes required rotation but excludes joint occupancy constraints. It is not an optimal joint-plan estimate. The actual departure heading is itself a result of preceding execution, and completed-task selection is observed rather than randomized; the lower bound is not a wholly external measure of assigned difficulty.

For task k, d is this shortest action distance and elapsed time is finish minus assignment. Completed-task excess is elapsed minus d. We retain zero distances and every agent with no completed task. A negative excess would trigger a material reconstruction failure; it is never clipped. At the mission horizon, each agent still has an active task. Its observed terminal exposure B is the horizon minus that task's assignment step. A newly assigned terminal task can have zero exposure, and its unobserved eventual completion is not invented.

Let c denote an agent's completed-task count, D the sum of its completed-task distances and E its completed excess. Immediate successive assignments produce the exact per-agent identity H = D + E + B. Summing gives the full team identity N times H = sum(D) + sum(E) + sum(B). This ledger reserves a distinct place for unfinished work. Right-censoring is not a statement that an event never occurs (Kaplan and Meier, 1958); we do not estimate a completion-time survival distribution or assume that the terminal tasks resemble completed ones.

The graph computations also receive an optimality check based on the routing recursion associated with Bellman (1958). Every non-goal state must have distance one plus the smallest successor distance. All outgoing edges must satisfy the lower-bound inequality, while an attaining transition provides a route witness. Together with zero goal distances and positive integer decrements, these conditions establish both the bound and its attainment. They provide a numerical validation of our graph definition, without validating that definition as an economic value or a physical travel-time model.

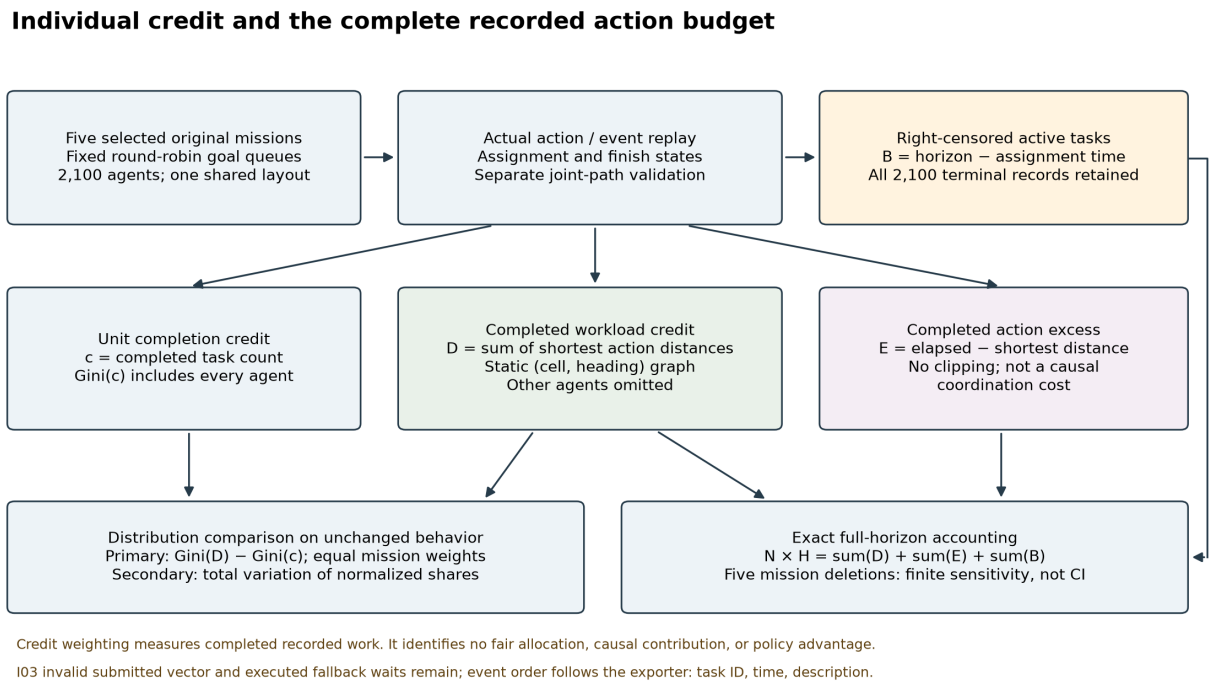

Figure 1. *Original accounting model. Arrows show reconstruction and measurement operations on unchanged recorded behavior. Completed work, completed excess and terminal exposure exhaust the observed action budget.*

4. Concentration and sensitivity are defined before examining their signs

We use the uncorrected finite Gini index: the sum of all ordered pairwise absolute differences divided by twice the agent count times the sum of values. Concentration measures have a long connection with mean differences and cumulative shares (Gini, 1921; Lorenz, 1905). Our convention includes every agent and has finite upper bound one minus one divided by N. The two inputs are nonnegative ratio-scale quantities with meaningful zeros. Choosing a concentration measure also chooses properties of the comparison, as Allison (1978) explains; no alternative index was selected after inspecting the result.

For each mission, the primary contrast is Gini(D) minus Gini(c). The aggregate is the arithmetic mean of these five signed contrasts, giving every mission equal weight. Weighting defines the descriptive target rather than automatically representing a population (Solon et al., 2015). The secondary total-variation quantity is half the sum of absolute differences between each agent's normalized count share and normalized workload share. It reports how much the two share vectors differ, alongside the scalar concentration comparison. Retaining the component values avoids treating one index as a complete distributional account (Cowell, 2011).

We delete each mission once and recompute the mean of the four remaining signed contrasts. All five deletion values are reported. Their range describes finite mission composition, without a sampling coverage interpretation. Within-run records are not interchangeable with replicated simulation experiments (Law, 2015). These missions also share a layout and differ in selected implementation, load and horizon. No independent-agent or independent-task bootstrap, standard error, significance test or new-layout guarantee is supplied.

The budget identity and the inequality contrast answer different questions. Adding components of observed time does not automatically attribute inequality to those components. Factor-based inequality decomposition requires its own allocation principles (Shorrocks, 1982), which we do not implement. Similarly, the welfare interpretation of an inequality index depends on normative assumptions (Atkinson,

1970). Our workload weights are a stated measurement convention, rather than a verified definition of fairness, merit or socially desirable credit.

5. Unit completion credit is more concentrated in every released mission

All five distribution contrasts are defined. Their equal-mission mean is -0.035098771, and every mission has a negative difference. In MR23-I-04, count Gini is 0.071241, whereas workload Gini is 0.020434. MR23-I-08 has the largest two component indices in this cohort, 0.319062 and 0.298492, while still retaining a negative contrast. Table 2 gives the complete comparison. The result concerns a change in measurement weights applied to the same actions; no agent changed its behavior.

Table 2. *Actual per-mission credit distributions. ΔG is workload Gini minus count Gini. V is total variation between the two normalized credit vectors. The final columns retain zero-completion agents and all terminal task records.*

Mission	Gini(c)	Gini(D)	ΔG	V	Zero-completion agents	Terminal tasks
MR23-I-03	0.070434	0.027341	-0.043093	0.050621	0	200
MR23-I-04	0.071241	0.020434	-0.050807	0.048988	0	100
MR23-I-05	0.103190	0.075797	-0.027392	0.047889	0	400
MR23-I-07	0.197845	0.164213	-0.033633	0.063656	6	600
MR23-I-08	0.319062	0.298492	-0.020569	0.061912	32	800

Total variation ranges from 0.047889 to 0.063656. These differences show that the credit vectors change even though their normalized totals are both one. Thirty-eight agents have no completed task, all in I-07 or I-08; they remain in both Gini denominators. Zero completed credit does not imply no attempted movement or no time spent. Their active terminal task histories remain observable in the ledger.

The five mission-deletion means range from -0.038731170 to -0.031171803. Removing the flagged I-03 trace gives -0.033100229; it does not reverse the direction. This is a transparent finite sensitivity calculation, without implying that the source error had no behavioral effect or that all unseen missions would share the same sign.

Table 3. *Every prespecified mission-deletion result. Each value averages four unchanged mission contrasts. The range is not a confidence interval.*

Mission omitted	Missions retained	Mean ΔG
MR23-I-03	4	-0.033100229
MR23-I-04	4	-0.031171803
MR23-I-05	4	-0.037025345
MR23-I-07	4	-0.035465309
MR23-I-08	4	-0.038731170

6. Unfinished exposure changes the interpretation of a complete ledger

The three budget components add exactly to every agent's horizon and every team's full action budget. Table 4 reports the integer values rather than only a ratio computed from completed tasks. In I-04, completed shortest-action workload occupies 89.074% of the full budget, completed excess 7.738% and terminal exposure 3.188%. In I-08, the corresponding values are 9.191%, 63.369% and 27.440%. These contrasts describe different selected missions; their differences cannot identify a causal effect of increasing the team size or of changing the planner.

Table 4. *Exhaustive action-budget accounting. D and E concern completed tasks only. B is observed exposure in the active terminal tasks. Fractions use the full N×H denominator, without removing unfinished observations.*

Mission	Full budget N×H	Completed D	Completed E	Terminal B	B / full budget
MR23-I-03	100000	78766	17891	3343	0.033430
MR23-I-04	50000	44537	3869	1594	0.031880
MR23-I-05	400000	189839	192213	17948	0.044870
MR23-I-07	600000	148295	373431	78274	0.130457
MR23-I-08	1600000	147058	1013898	439044	0.274402

Completed excess includes all time beyond the independent lower bound within finished tasks. It can include turns, detours, necessary yielding, fallback waits and other consequences of the joint execution. A smaller value is not automatically available to a different joint planner, because shortest individual routes may be mutually incompatible. We therefore do not call the excess a recoverable coordination loss. Terminal exposure has an additional boundary: it records time already spent, without completing or valuing the remaining task.

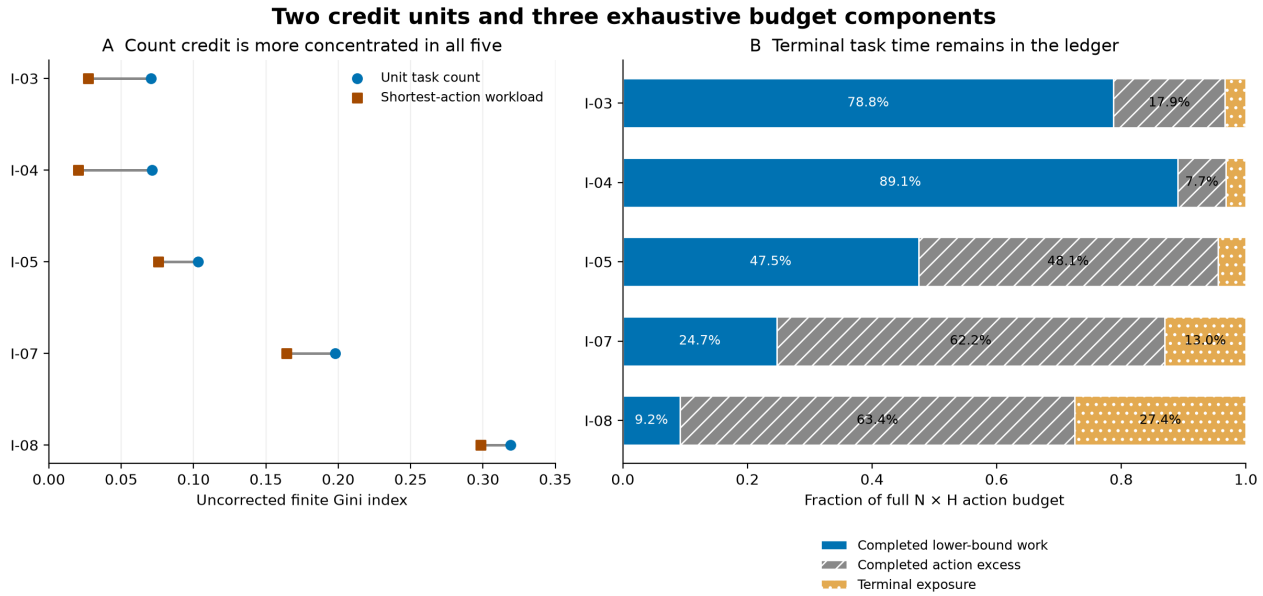

Figure 2. *Full finite results. Circles and squares distinguish the two credit units. The stacked bars exhaust each mission's action budget and retain terminal exposure. All five missions, including I-03, remain visible.*

The independent reconstruction checks all 26,288 task records and 2,752,100 agent states, including initial states. A separate PowerShell JSON decoder agrees with Python on canonical hashes of every used start, task, event and actual action string. Exhaustive optimality checks cover 2,683,044 goal-state combinations and 7,446,348 directed edges; 819 separate forward searches agree with the reverse-search

distances. Direct pairwise integer sums reproduce all ten Gini indices and the exact rational primary, secondary and deletion results. No material reconstruction failure or undefined mission contrast occurred.

7. Individual fitness, observed workload and causal contribution remain distinct

Guo et al. (2023) provide a specific motivation for examining individual measurements within a team objective. Their EMARL method constructs an additive flocking reward, uses accumulated individual rewards as fitness and applies senior/junior evolutionary selection with policy inheritance. The actual method discussion warns that a poor position can lower an agent's measured performance without demonstrating a poorer policy. Our completion/workload comparison examines a related measurement ambiguity with observed fixed task queues. It does not execute EMARL, transplant its reward equation, test its convergence argument or establish an advantage of evolutionary selection.

Neither credit vector supplies a counterfactual marginal contribution. Coalition-value approaches require a model of what different coalitions can achieve (Shapley, 1953). Counterfactual multi-agent policy gradients instead use a critic and a baseline that varies one agent's action while holding others fixed (Foerster et al., 2018). We observe one selected joint execution per mission and provide neither construction. An agent with modest completion credit may have faced a long queue segment, congestion or a consequential yielding role; the present arithmetic cannot allocate those causes separately.

The distinction also matters if a reported score is later proposed as a learning reward. Reinforcement learning optimizes accumulated reward through interaction (Sutton and Barto, 2018). Changing a retrospective credit unit does not show what a learner would do under that unit. Policy invariance under reward transformations requires specific conditions (Ng et al., 1999); none is established by the present reweighting. A lower measured Gini therefore does not authorize a claim that workload-weighted training would improve teamwork or produce equitable behavior.

The completed-task set is itself selected by reaching goals before the horizon. Its workload distribution excludes the eventual work of active terminal tasks, which remains unknown. Shared layouts, differing implementation revisions and best-known-submission selection further limit interpretation. The full terminal ledger prevents their observed time from disappearing, but it does not solve this selection problem. A new-layout transfer study or causal coordination experiment would need additional evidence and a separately fixed design.

8. Reproducible finite comparisons and conclusion

The calculation uses integer action counts and exact rational arithmetic through Python's Fraction implementation (Python Software Foundation, n.d.). NumPy stores the reconstructed states and distance arrays (Harris et al., 2020), pandas exports agent, task and mission tables (McKinney, 2010), and Matplotlib produces the original model and quantitative figures (Hunter, 2007). Data, program versions, protocol and independent-check hashes accompany the draft. Such an inspectable computational chain supports reproducibility (Peng, 2011), with intermediate records retained as recommended by Sandve et al. (2013).

For these five released missions, unit completion credit is consistently more concentrated than the independently traversable completed workload measured here. The same missions also show that terminal task exposure can occupy a substantial portion of the full action horizon. These findings favor reporting both the credit unit and the unfinished-task ledger when interpreting individual summaries of a team objective. They establish a finite measurement difference, without identifying deserved credit, physical safety, causal coordination efficiency or a policy improvement.

Original benchmark files remain local under their unresolved redistribution terms; derived statistical summaries preserve attribution and provenance. No new training, physical experiment, human-participant study was performed. Accountable authorship, source-claim and rights review, funding/conflict declarations and final scientific approval remain pending. AI-assisted computational and writing checks are recorded separately from human approval and submission authorization.

References

1. Bellman, Richard. (1958). On a routing problem. Quarterly of Applied Mathematics 16(1), 87-90. https://doi.org/10.1090/qam/102435.
2. Foerster, Jakob; Farquhar, Gregory; Afouras, Triantafyllos; Nardelli, Nantas; Whiteson, Shimon. (2018). Counterfactual Multi-Agent Policy Gradients. Proceedings of the AAAI Conference on Artificial Intelligence 32(1). https://doi.org/10.1609/aaai.v32i1.11794.
3. Hunter, John D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering 9(3), 90-95. https://doi.org/10.1109/mcse.2007.55.
4. Peng, Roger D. (2011). Reproducible Research in Computational Science. Science 334(6060), 1226-1227. https://doi.org/10.1126/science.1213847.
5. League of Robot Runners. (n.d.-a). 2023 Main Round benchmark instances and best-known solutions, revision 1adae003763d53eb4100a59ce4bcbc0c7212eae0. MAPF-Competition [Data]. https://github.com/MAPF-Competition/Benchmark-Archive/tree/1adae003763d53eb4100a59ce4bcbc0c7212eae0/2023%20Competition.
6. Law, Averill M. (2015). Simulation Modeling and Analysis (5th edition). McGraw-Hill Education. https://www.mheducation.com/highered/product/simulation-modeling-and-analysis-law.html.
7. Gini, Corrado. (1921). Measurement of Inequality of Incomes. The Economic Journal 31(121), 124-126. https://doi.org/10.2307/2223319.
8. Sutton, Richard S., and Andrew G. Barto. *Reinforcement Learning: An Introduction*. 2nd ed., MIT Press, 2018. http://incompleteideas.net/book/the-book-2nd.html.
9. Atkinson, Anthony B. (1970). On the measurement of inequality. Journal of Economic Theory 2(3), 244-263. https://doi.org/10.1016/0022-0531(70)90039-6.
10. League of Robot Runners. (n.d.-b). Historical Start-Kit action model, task assignment and output specification, revision cdd0356af57a394d4d242a5cc4896e4086b3df31. MAPF-Competition [Source code]. https://github.com/MAPF-Competition/Start-Kit/tree/cdd0356af57a394d4d242a5cc4896e4086b3df31.
11. Hand, David J. (2004). Measurement Theory and Practice: The World Through Quantification. Wiley. https://www.wiley-vch.de/de/fachgebiete/mathematik-und-statistik/measurement-theory-and-practice-978-0-470-68567-9
12. Guo, Yunxiao; Xie, Xinjia; Zhao, Runhao; Zhu, Chenglan; Yin, Jiangting; Long, Han. (2023). Cooperation and Competition: Flocking with Evolutionary Multi-Agent Reinforcement Learning. Lecture Notes in Computer Science, 271-283. https://doi.org/10.1007/978-3-031-30105-6_23.
13. Stern, Roni; Sturtevant, Nathan R.; Felner, Ariel; Koenig, Sven; Ma, Hang; Walker, Thayne T.; Li, Jiaoyang; Atzmon, Dor; Cohen, Liron; Kumar, T. K. Satish; Boyarski, Eli; Barták, Roman. (2019). Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks. Proceedings of the International Symposium on Combinatorial Search 10(1), 151-158. https://doi.org/10.1609/socs.v10i1.18510.
14. Cowell, Frank. (2011). Measuring Inequality. Oxford University Press. https://doi.org/10.1093/acprof:osobl/9780199594030.001.0001.
15. Lee, C. Y. (1961). An Algorithm for Path Connections and Its Applications. IEEE Transactions on Electronic Computers EC-10(3), 346-365. https://doi.org/10.1109/tec.1961.5219222.
16. Chan, Shao-Hung; Chen, Zhe; Guo, Teng; Zhang, Han; Zhang, Yue; Harabor, Daniel; Koenig, Sven; Wu, Cathy; Yu, Jingjin. (2024). The League of Robot Runners: Competition Goals, Designs, and Implementation. ICAPS 2024 System's Demonstration Track. https://harabor.net/data/papers/lorr-demo-icaps24.pdf.
17. Jacobs, Abigail Z.; Wallach, Hanna. (2021). Measurement and Fairness. Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 375-385. https://doi.org/10.1145/3442188.3445901.
18. Allison, Paul D. (1978). Measures of Inequality. American Sociological Review 43(6), 865-880. https://doi.org/10.2307/2094626.

19. Harris, Charles R.; Millman, K. Jarrod; van der Walt, Stéfan J.; Gommers, Ralf; Virtanen, Pauli; Cournapeau, David; Wieser, Eric; Taylor, Julian; Berg, Sebastian; Smith, Nathaniel J.; Kern, Robert; Picus, Matti; Hoyer, Stephan; van Kerkwijk, Marten H.; Brett, Matthew; Haldane, Allan; del Río, Jaime Fernández; Wiebe, Mark; Peterson, Pearu; Gérard-Marchant, Pierre; Sheppard, Kevin; Reddy, Tyler; Weckesser, Warren; Abbasi, Hameer; Gohlke, Christoph; Oliphant, Travis E. (2020). Array programming with NumPy. Nature 585(7825), 357-362. https://doi.org/10.1038/s41586-020-2649-2.

20. Python Software Foundation. (n.d.). Python 3.12 documentation: fractions, collections, array, json and csv. Python Software Foundation [Software documentation]. https://docs.python.org/3.12/library/fractions.html.

21. Shapley, L. S. (1953). 17. A Value for n-Person Games. Contributions to the Theory of Games (AM-28), Volume II, 307-318. https://doi.org/10.1515/9781400881970-018.

22. Shorrocks, A. F. (1982). Inequality Decomposition by Factor Components. Econometrica 50(1), 193-211. https://doi.org/10.2307/1912537.

23. Sandve, Geir Kjetil; Nekrutenko, Anton; Taylor, James; Hovig, Eivind. (2013). Ten Simple Rules for Reproducible Computational Research. PLoS Computational Biology 9(10), e1003285. https://doi.org/10.1371/journal.pcbi.1003285.

24. Gebru, Timnit; Morgenstern, Jamie; Vecchione, Briana; Vaughan, Jennifer Wortman; Wallach, Hanna; III, Hal Daumé; Crawford, Kate. (2021). Datasheets for datasets. Communications of the ACM 64(12), 86-92. https://doi.org/10.1145/3458723.

25. Solon, Gary; Haider, Steven J.; Wooldridge, Jeffrey M. (2015). What Are We Weighting For? Journal of Human Resources 50(2), 301-316. https://doi.org/10.3368/jhr.50.2.301.

26. Ng, Andrew Y.; Harada, Daishi; Russell, Stuart. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. Proceedings of the Sixteenth International Conference on Machine Learning, 278-287. https://people.eecs.berkeley.edu/~pabbeel/cs287-fa09/readings/NgHaradaRussell-shaping-ICML1999.pdf.

27. Lorenz, M. O. (1905). Methods of Measuring the Concentration of Wealth. Publications of the American Statistical Association 9(70), 209-219. https://doi.org/10.1080/15225437.1905.10503443.

28. Kaplan, E. L.; Meier, Paul. (1958). Nonparametric Estimation from Incomplete Observations. Journal of the American Statistical Association 53(282), 457-481. https://doi.org/10.1080/01621459.1958.10501452.

29. McKinney, Wes. (2010). Data Structures for Statistical Computing in Python. Proceedings of the Python in Science Conference, 56-61. https://doi.org/10.25080/majora-92bf1922-00a.

30. Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. Numerische Mathematik 1(1), 269-271. https://doi.org/10.1007/bf01386390.