Counter Machines and the Border of Effective Computation

VeTSS Summer School 2026

Patrick Totzke

Counter Machines a la Minsky are one of the simplest computational models and a great starting point to explore the limits of effective computability and verification. We will recall some foundational results in the theory of computation, including the undecidability of the termination problem for two-counter machines and related decision questions. These provide natural starting points to show impossibility results and complexity lower bounds.

We will survey ways to modify the model to make it less expressive and allow to recover decidability for verification questions. This can be achieved by restricting the use of zero-tests, accepting imprecise updates to counters, allowing a single counter, or combinations thereof. Several such models form the theoretical underpinning of automated formal verification methods in use today and we will demonstrate applications and directions for future research.

1 Effective Computation and Minsky Machines

Exercises

1. Construct a Turing Machine that implements an increment-by-one operation on binary encoded input.
2. Turn the incrementing TM into a machine with two stacks.
3. Construct a 2CM with edge labels that prints the word $a^n bc^n$ if started with counters $0, n$.
4. Doubling: Construct a 2CM that moves $(s_0, 0, n)$ to $(s_f, 0, 2n)$. What changes if you want to multiply by 3,5,7?
5. Conditional division by 3: Construct a 3CM that, if started in $(s, 0, 0, n)$, either ends in $(s', 0, 0, n/3)$ if n is divisible by 3, or in $(s'', 0, 0, n)$ otherwise.

References

[Min67] M. L. Minsky. *Computation: finite and infinite machines*. Prentice-Hall, Inc., 1967.

[Sip12] M. Sipser. *Introduction to the Theory of Computation*. Cengage Learning, 2012. URL: `https://cs.brown.edu/courses/csci1810/fall-2023/resources/ch2_readings/Sipser_Introduction.to.the.Theory.of.Computation.3E.pdf`.

[Tur37] A. M. Turing. "On computable numbers, with an application to the Entscheidungsproblem". In: *Proc. London Math. Soc.* s2-42.1 (1937), pp. 230–265. DOI: `10.1112/PLMS/S2-42.1.230`. URL: `https://doi.org/10.1112/plms/s2-42.1.230`.

2 Petri Nets and other Well-structured Transition Systems

Exercises

1. Construct a 3-VASS from [HP76]; Compute the reachability set from $(s, 0, 1, 0)$.
2. Show that one can simulate any VASS using a VAS; How many extra dimensions do you need?
3. Show that the coverability problem reduces to the reachability problem for k-VASS.
4. Show that the coverability problem for 1-VASS can be solved in polynomial time.
5. Prove Dickson's Lemma: Assuming that $(N, \leq)$ is a wqo, show that $(N^2, \leq)$ is a wqo for the point-wise ordering.
6. Build the KM-tree for the VAS with transitions $(-1, 2), (1, -1)$ starting at $v_0 = (1, 0)$.
7. LCM: convince yourself that the upwards-closure of $Pre(s)$ is effectively computable from the up-closure of s.
8. GCS: define the partial ordering that is a WQO on GCS configurations. Is if effective?

References

[BP14] L. Bozzelli and S. Pinchinat. "Verification of gap-order constraint abstractions of counter systems". In: *Theoretical Computer Science* 523 (2014), pp. 1–36. DOI: 10.1016/j.tcs.2013.12.002.

[CO21] W. Czerwinski and L. Orlikowski. "Reachability in Vector Addition Systems is Ackermann-complete". In: *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*. 2021, pp. 1229–1240. DOI: 10.1109/FOCS52979.2021.00120. URL: https://doi.org/10.1109/FOCS52979.2021.00120.

[Cpn] *Colored Petri Nets – Industrial Use and Applications*. Webpage. URL: https://cs.au.dk/cpnets/industrial-use/.

[FS01] A. Finkel and P. Schnoebelen. "Well-structured transition systems everywhere!" In: *Theor. Comput. Sci.* 256.1-2 (2001), pp. 63–92. DOI: 10.1016/S0304-3975(00)00102-X.

[KM69] R. M. Karp and R. E. Miller. "Parallel program schemata". In: 3.2 (1969), pp. 147–195. DOI: 10.1016/S0022-0000(69)80011-5.

[Kos82] S. R. Kosaraju. "Decidability of reachability in VAS". In: *STOC'82*. 1982, pp. 267–281. DOI: 10.1145/800070.802201.

[Lip76] R. J. Lipton. *The reachability problem requires exponential space*. Tech. rep. 62. Yale University, 1976.

[May03] R. Mayr. "Undecidable problems in unreliable computations". In: *Theor. Comput. Sci.* 297.1-3 (2003), pp. 337–354. DOI: 10.1016/S0304-3975(02)00646-1. URL: https://doi.org/10.1016/S0304-3975(02)00646-1.

[May81] E. W. Mayr. "An algorithm for the general Petri net reachability problem". In: *STOC'81*. 1981, pp. 238–246. DOI: 10.1145/800076.802477.

[MT16] R. Mayr and P. Totzke. "Branching-Time Model Checking Gap-Order Constraint Systems". In: *Fundamenta Informaticae* 143.3-4 (2016), pp. 339–353. DOI: 10.3233/FI-2016-1317.

[Rac78] C. Rackoff. "The covering and boundedness problems for vector addition systems". In: 6.2 (1978), pp. 223–231. DOI: 10.1016/0304-3975(78)90036-1.

[Sch10a] P. Schnoebelen. "Lossy Counter Machines Decidability Cheat Sheet". In: *Reachability Problems, 4th International Workshop, RP 2010, Brno, Czech Republic, August 28-29, 2010. Proceedings.* Vol. 6227. 2010, pp. 51–75. DOI: 10.1007/978-3-642-15349-5_4. URL: https://doi.org/10.1007/978-3-642-15349-5_4.

[Sch10b] P. Schnoebelen. "Revisiting Ackermann-Hardness for Lossy Counter Machines and Reset Petri Nets". In: *Mathematical Foundations of Computer Science 2010, 35th International Symposium, MFCS 2010, Brno, Czech Republic, August 23-27, 2010. Proceedings*. Vol. 6281. 2010, pp. 616–628. DOI: 10.1007/978-3-642-15155-2_54. URL: https://doi.org/10.1007/978-3-642-15155-2_54.

[Sch16] S. Schmitz. "Complexity Hierarchies beyond Elementary". In: *ACM Trans. Comput. Theory* 8.1 (2016), 3:1–3:36. DOI: 10.1145/2858784. URL: https://doi.org/10.1145/2858784.

3 One-counter Machines

Exercises

1. Construct a 1CA that recognises the language $\{a^n b^n \mid n > 0\}$.

References

[HT18] P. Hofman and P. Totzke. "Trace inclusion for one-counter nets revisited". In: *Theor. Comput. Sci.* 735 (2018), pp. 50–63. DOI: 10.1016/J.TCS.2017.05.009.

[VP75] L. G. Valiant and M. Paterson. "Deterministic One-Counter Automata". In: *J. Comput. Syst. Sci.* 10.3 (1975), pp. 340–350. DOI: 10.1016/S0022-0000(75)80005-5.

4 Games

References

[ABT24] P. Austin, S. Bose, and P. Totzke. "Parity Games on Temporal Graphs". In: *International Conference on Foundations of Software Science and Computational Structures (FoSSaCS)*. Vol. 14574. 2024, pp. 79–98. DOI: 10.1007/978-3-031-57228-9_5. URL: https://doi.org/10.1007/978-3-031-57228-9_5.

[Aus23] P. Austin. *temporis: temporal graph tools (GitHub)*. https://github.com/petejaustin/temporis. GitHub repository. 2023.

[con24] G. contributors. *GameGraphGym (ggg) - graph/game benchmarks and tools (GitHub)*. https://github.com/gamegraphgym/ggg. GitHub repository. 2024.

[Fij+25] N. Fijalkow, C. Aiswarya, G. Avni, N. Bertrand, P. Bouyer, R. Brenguier, A. Carayol, A. Casares, J. Fearnley, P. Gastin, H. Gimbert, T. A. Henzinger, F. Horn, R. Ibsen-Jensen, N. Markey, B. Monmege, P. Novotný, P. Ohlmann, M. Randour, O. Sankur, S. Schmitz, O. Serre, M. Skomra, N. Sznajder, and P. Vandenhove. *Games on Graphs: From Logic and Automata to Algorithms*. 2025. URL: https://arxiv.org/abs/2305.10546.

[JLS07] M. Jurdzinski, F. Laroussinie, and J. Sproston. "Model Checking Probabilistic Timed Automata with One or Two Clocks". In: *Tools and Algorithms for the Construction and Analysis of Systems, 13th International Conference, TACAS 2007, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2007 Braga, Portugal, March 24 - April 1, 2007, Proceedings*. Vol. 4424. 2007, pp. 170–184. DOI: 10.1007/978-3-540-71209-1_15. URL: https://doi.org/10.1007/978-3-540-71209-1_15.

[paz24] pazz. *ontime: scheduling utilities (GitHub)*. https://github.com/pazz/ontime. GitHub repository. 2024.