Robust Multidimensional Mean-Payoff Games are Undecidable

Yaron Velner*

The Blavatnik School of Computer Science, Tel Aviv University, Israel

Abstract. Mean-payoff games play a central role in quantitative synthesis and verification. In a single-dimensional game a weight is assigned to every transition and the objective of the protagonist is to assure a non-negative limit-average weight. In the multidimensional setting, a weight vector is assigned to every transition and the objective of the protagonist is to satisfy a boolean condition over the limit-average weight of each dimension, e.g., $LimAvg(x_1) \leq 0 \vee LimAvg(x_2) \geq 0 \wedge LimAvg(x_3) \geq 0$. We recently proved that when one of the players is restricted to finite-memory strategies then the decidability of determining the winner is inter-reducible with Hilbert's Tenth problem over rationals (a fundamental long-standing open problem). In this work we consider arbitrary (infinite-memory) strategies for both players and show that the problem is undecidable.

1 Introduction

Two-player games on graphs provide the mathematical foundation for the study of reactive systems. In these games, the set of vertices is partitioned into player-1 and player-2 vertices; initially, a pebble is placed on an initial vertex, and in every round, the player who owns the vertex that the pebble resides in, advances the pebble to an adjacent vertex. This process is repeated forever and give rise to a *play* that induces an infinite sequence of edges. In the quantitative framework, an objective assigns a value to every play, and the goal of player 1 is to assure a value of at least ν to the objective. In order to have robust quantitative specifications, it is necessary to investigate games on graphs with multiple (and possibly conflicting) objectives. Typically, multiple objectives are modeled by multidimensional weight functions (e.g., [4,5,7,1]), and the outcome of a *play* is a vector of values $(r_1, r_2, \ldots, r_k)$. A *robust specification* is a boolean formula over the atoms $r_i \sim \nu_i$, for $\sim \in \{\leq, <, \geq, >\}$, $i \in \{1, \ldots, k\}$ and $\nu_i \in \mathbb{Q}$. For example, $\varphi = ((r_1 \geq 9 \vee r_2 \leq 9) \wedge r_3 < 0 \wedge r_4 > 9)$. The most well studied quantitative metric is the mean-payoff objective, which assigns the limit-average (long-run average) weight to an infinite sequence of weights (and if the limit does not exist, then we consider the limit infimum of the sequence). In this setting, r_i is the limit-average of dimension i of the weight function, and the goal of player 1 is to satisfy the boolean condition. In this work we prove that determining whether player 1 can satisfy such a condition is undecidable.

* The author was funded by the European Research Council under the European Unions Seventh Framework Program (FP7/20072013) / ERC grant agreement no. [321174-VSSC].

A. Pitts (Ed.): FOSSACS 2015, LNCS 9034, pp. 312–327, 2015.
DOI: 10.1007/978-3-662-46678-0_20

Related Work. The model checking problem (one-player game) for such objectives (with some extensions) was considered in [1,6,3,12,13] and decidability was established. Two-player games for restricted subclasses that contain only conjunction of atoms were studied in [15,7,2,9] and tight complexity bounds were obtained (and in particular, the problem was proved to be decidable). In [16] a subclass that contains disjunction and conjunction of atoms of the form $r_i \sim \nu_i$ for $\sim \in \{\geq, >\}$ was studied and decidability was shown. In [14] we considered a similar objective but restricted player-1 to play only with finite-memory strategies. We showed that the problem is provably hard to solve and its decidability is inter-reducible with Hilbert's tenth problem over rationals — a fundamental long standing open problem. In this work we consider for the first time games with robust quantitative class of specifications that is closed under boolean union, intersection and complement with arbitrary (infinite-memory) strategies.

Undecidability for (single-dimensional) mean-payoff games was proved for partial information mean-payoff games [10] and for mean-payoff games that are played over infinite-state pushdown automata [8]. These works did not exploit the different properties of the $\geq$ and $\leq$ operators (which correspond to the different properties of limit-infimum-average and limit-supremum-average). To the best of our knowledge, the undecidability proof in the paper is the first to exploit these properties. (As we mentioned before, when we consider only the $\geq$ and $>$ operators, the problem is decidable.)

Robust multidimensional mean-payoff games were independently suggested as a subject to future research by Alur et al [1], by us [16], and by Doyen [11].

Structure of this Paper. In the next section we give the formal definitions for robust multidimensional mean-payoff games. We prove undecidability by a reduction from the halting problem of a two-counter machine. For this purpose we first present a reduction from the halting problem of a one-counter machine and then we extend it to two-counter machine. In Section 3 we present the reduction and give an intuition about its correctness. In Section 4 we give a formal proof for the correctness of the reduction and extend the reduction to two-counter machine. Due to lack of space, some of the proof are omitted. Full proofs are available in the technical report [17].

2 Robust Multidimensional Mean-Payoff Games

Game Graphs. A *game graph* $G = ((V, E), (V_1, V_2))$ consists of a *finite* directed graph (V, E) with a set of vertices V a set of edges E, and a partition (V_1, V_2) of V into two sets. The vertices in V_1 are *player-1 vertices*, where player 1 chooses the outgoing edges, and the vertices in V_2 are *player 2 vertices*, where player 2 (the adversary to player 1) chooses the outgoing edges. We assume that every vertex has at least one out-going edge.

Plays. A game is played by two players: player 1 and player 2, who form an infinite path in the game graph by moving a token along edges. They start by placing the token on an initial vertex, and then they take moves indefinitely in the following way. If the token is on a vertex in V_1, then player 1 moves the token along one of the edges going out of the vertex. If the token is on a vertex in V_2, then player 2 does likewise. The result is an infinite path in the game graph, called *plays*. Formally, a *play* is an infinite sequence of vertices such that $(v_k, v_{k+1}) \in E$ for all $k \geq 0$.

Strategies. A strategy for a player is a rule that specifies how to extend plays. Formally, a *strategy* τ for player 1 is a function τ: $V^* \cdot V_1 \to V$ that, given a finite sequence of vertices (representing the history of the play so far) which ends in a player 1 vertex, chooses the next vertex. The strategy must choose only available successors. The strategies for player 2 are defined analogously. A winning objective is a subset of V^ω and a strategy is a winning strategy if it assures that every formed play is in the winning objective.

Multidimensional Mean-Payoff Objectives. For multidimensional mean-payoff objectives we will consider game graphs along with a weight function $w : E \to \mathbb{Q}^k$ that maps each edge to a vector of rational weights. For a finite path π, we denote by $w(\pi)$ the sum of the weight vectors of the edges in π and $avg(\pi) = \frac{w(\pi)}{|\pi|}$, where $|\pi|$ is the length of π, denote the average vector of the weights. We denote by $avg_i(\pi)$ the projection of $avg(\pi)$ to the i-th dimension. For an infinite path π, let π_i denote the finite prefix of length i of π; and we define $LimInfAvg_i(\pi) = \liminf_{i\to\infty} avg(\rho_i)$ and analogously $LimSupAvg_i(\pi)$ with $\liminf$ replaced by $\limsup$. For an infinite path π, we denote by $LimInfAvg(\pi) = (LimInfAvg_1(\pi), \dots, LimInfAvg_k(\pi))$ (resp. $LimSupAvg(\pi) = (LimSupAvg_1(\pi), \dots, LimSupAvg_k(\pi)))$ the limit-inf (resp. limit-sup) vector of the averages (long-run average or mean-payoff objectives). A multidimensional mean-payoff condition is a boolean formula over the atoms $LimInfAvg_i \sim \nu_i$ for $\sim \in \{\geq, >, \leq, >\}$. For example, the formula $LimInfAvg_1 > 8 \vee LimInfAvg_2 \leq -10 \wedge LimInfAvg < 9$ is a possible condition and a path π satisfies the formula if $LimInfAvg_1(\pi) > 8 \vee LimInfAvg_2(\pi) \leq -10 \wedge LimInfAvg(\pi) < 9$. We note that we may always assume that the boolean formula is positive (i.e., without negation), as, for example, we can always replace $\neg(r \geq \nu)$ with $r < \nu$.

For a given multidimensional weighted graph and a multidimensional mean-payoff condition, we say that player 1 is the winner of the game if he has a winning strategy that satisfy the condition against any player-2 strategy.

For an infinite sequence or reals $x_1, x_2, x_3, \dots$ we have $LimInfAvg(x_1, x_2, \dots) = -LimSupAvg(-x_1, -x_2, \dots)$. Hence, an equivalent formulation for multidimensional mean-payoff condition is a positive boolean formula over the atoms $LimInfAvg_i \sim \nu_i$ and $LimSupAvg_i \sim \nu_i$ for $\sim \in \{\geq, >\}$. For positive formulas in which only the $LimInfAvg_i \sim \nu_i$ occur, determining the winner is decidable by [16]. In the sequel we abbreviate $LimInfAvg_i$ with $\underline{i}$ and $LimSupAvg_i$ with $\overline{i}$. In this work we prove undecidability for the general case and for this purpose it is enough to consider only the $\geq$ operator and thresholds 0. Hence, in the sequel, whenever it is clear that the threshold is 0, we abbreviate the condition $\underline{i} \geq 0$ with $\underline{i}$ and $\overline{i} \geq 0$ with $\overline{i}$. For example, $\underline{i} \vee \overline{j} \wedge \underline{\ell}$ stands for $LimInfAvg_i \geq 0 \vee LimSupAvg_j \geq 0 \wedge LimInfAvg_\ell \geq 0$. By further abuse of notation we abbreviate the current total weight in dimension i by i (and make sure that the meaning of i is always clear from the context) and the absolute value of the total weight by $|i|$.

3 Reduction from the Halting Problem and Informal Proof of Correctness

In this chapter we prove the undecidability of determining the winner in games over general multidimensional mean-payoff condition by a reduction from the halting problem of two-counter machine. For this purpose we will first show a reduction from the halting problem of a one-counter machine to multidimensional mean-payoff games, and the reduction from two-counter machines relies on similar techniques. We first give a formal definition for a one-counter machine, and in order to simplify the proofs we give a non-standard definition that is tailored for our needs. A *two-sided one-counter machine* M consists of two finite set of control states, namely Q (*left states*) and P (*right states*), an initial state $q_0 \in Q$, a final state $q_f \in Q$, a finite set of *left to right* instructions $\delta_{\ell \to r}$ and a finite set of *right to left* instructions $\delta_{r \to \ell}$. An instruction determines the next state and manipulates the value of the counter c (and initially the value of c is 0). A left to right instruction is of the form of either:

- $q : \textit{if}\, c = 0 \textit{ goto } p \textit{ else } c := c - 1 \textit{ goto } p'$, for $q \in Q$ and $p, p' \in P$; or
- $q : \textit{goto } p$, for $q \in Q$ and $p \in P$ (the value of c does not change).

A right to left instruction is of the form of either

- $p : c := c + 1 \textit{ goto } q$, for $p \in P$ and $q \in Q$; or
- $p : \textit{goto } q$, for a state $p \in P$ and a state $q \in Q$ (the value of c does not change).

We observe that in our model, decrement operations are allowed only in left to right instructions and increment operations are allowed only in right to left instructions. However, since the model allows state transitions that do not change the value of the counter (*nop transitions*), it is trivial to simulate a standard one-counter machine by a two-sided counter machine.

For the reduction we use the states of the game graph to simulate the states of the counter machine and we use two dimensions to simulate the value of the one counter. In the most high level view our reduction consists of three main gadgets, namely, reset, sim and blame (see Figure 1), and a state q_f that represents the final state of the counter machine. Intuitively, in the sim gadget player 1 simulates the counter machine, and if the final state q_f is reached then player 1 loses. If player 2 detects that player 1 does not simulate the machine correctly, then the play goes to the blame gadget. From the blame gadget the play will eventually arrive to the reset gadget. This gadget assigns proper values for all the dimensions of the game that are suited for an honest simulation in the sim gadget. When a play leaves the reset gadget, it goes to the first state of the simulation gadget which represent the first state of the counter machine.

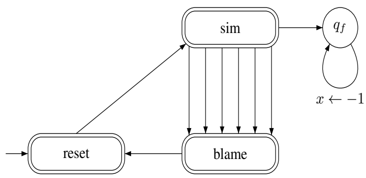

Fig. 1. Overview

We now describe the construction with more details. We first present the winning objective and then we describe each of the three gadgets. For a two-sided counter machine M we construct a game graph with 8 dimensions denoted by $\ell, r, g_s, c_+, c_-, g_c, x$ and y and the objective $[(\underline{\ell} \wedge \underline{r} \vee \overline{g_s}) \wedge (\underline{c_+} \wedge \underline{c_-} \vee \overline{g_c}) \wedge \overline{x} \wedge \overline{y}$.

The Sim Gadget. In the sim gadget player 1 suppose to simulate the run of M, and if the simulation is not honest, then player 2 activates a blame gadget. The simulation of the states is straight forward (since the game graph has states), and the difficulty is to simulate the counter value, more specifically, to simulate the zero testing of the counter. For this purpose we use the dimensions r, ℓ, g_s and c_+, c_-, g_c.

We first describe the role of r, ℓ and g_s. The reset gadget makes sure that in every invocation of the sim gadget, we have $avg(g_s) \approx -1, avg(r) \approx 1$ and $avg(\ell) \approx 0$. (The reader should read $a \approx b$ as "the value of a is *very close* to the value of b". Precise definitions are given in Section 4.) Then, during the simulation the value of g_s is always negative, and the blame gadget makes sure that player 1 must play in such a way that whenever the machine M is in a right state, $r \approx |g_s|$ and $\ell \approx 0$, and whenever the machine is in a left state, then $r \approx 0$ and $\ell \approx |g_s|$. Intuitively, the role of ℓ and r is to make sure that every left to right or right to left transition is simulated by a *significant* number of rounds in the sim gadget, and g_s is a *guard* dimension that makes sure that the above assumptions on r and ℓ are satisfied.

We now describe the role of c_+, c_- and g_c. In the beginning of each simulation (i.e., every time that the sim gadget is invoked), we have $avg(c_+) \approx avg(c_-) \approx 1$ and $avg(g_c) \approx -1$. During the entire simulation we have $avg(g_c) \approx -1$ and if c is the value of the counter in the current simulation (i.e., since the sim gadget was invoked), then $c_+ \approx |g_c| + |g_s|c$ and $c_- \approx |g_c| - |g_s|c$. Intuitively, whenever $c > 0$, then $c_- \ll |g_c|$, and if $c < 0$ (this can happen only if player 1 is dishonest), then $c_+ \ll |g_c|$ (the reader should read $a \ll b$ as "a is much smaller than b").

We now describe the gadgets that simulate the operations of inc, dec and nop. The gadgets are illustrated in Figures 2-5 and the following conventions are used: (i) Player 1 owns the ○ vertices, player 2 owns the □ vertices, and the ⊡ vertex stands for a gadget; (ii) A transition is labeled either with $a \leftarrow b$ symbol or with a text (e.g., blame). For a transition e the label $a \leftarrow b$ stands for $w_a(e) = b$. Whenever the weight of a dimension is not explicitly presented, then the weight is 0. We use text labels only to give intuition on the role of the transition. In such transitions the weights of all dimensions are 0.

In order to satisfy the invariants, in the first state of every inc, dec or nop gadget, in a left to right transition, player 1 always moves to the state below (namely, to $\ell \ll 0?$) until $\ell \approx 0$ and $r \approx |g_s|$, and in a right to left transition he always moves to the state below (namely, to $r \ll 0?$) loops until $r \approx 0$ and $\ell \approx |g_s|$. If in a left to right gadget the loop is followed too many times, then ℓ is decremented too many times and player 2 has an incentive invokes the $\ell \ll 0$ gadget. If the loop was not followed enough times, then r was not incremented enough times and player 2 invokes the $r \ll |g_s|$ blame gadget. Hence, the blame gadgets allows player 2 to blame player 1 for violating the assumptions about the values of ℓ, r and g_s.

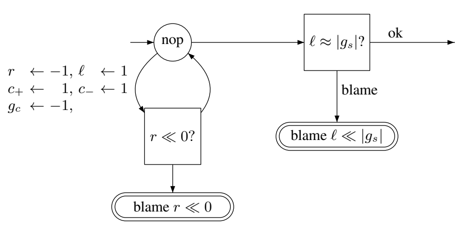

Fig. 2. nop $r \to \ell$ gadget

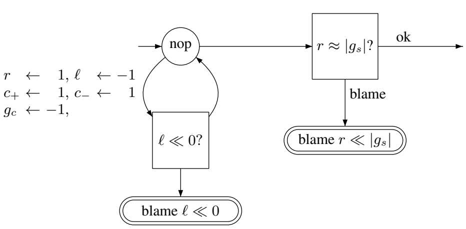

Fig. 3. nop $\ell \to r$ gadget

A transition $q : \textit{if } c = 0 \textit{ goto } p \textit{ else } c := c - 1 \textit{ goto } p'$, for $q \in Q$ and $p, p' \in P$ is described in Figure 6.

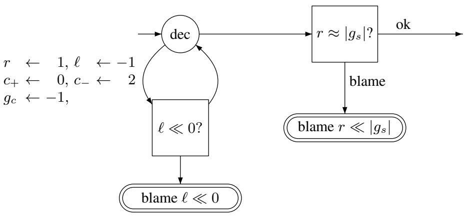

Fig. 4. dec $\ell \to r$ gadget

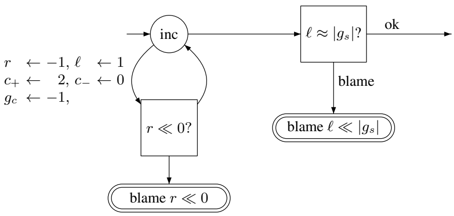

Fig. 5. inc $r \to \ell$ gadget

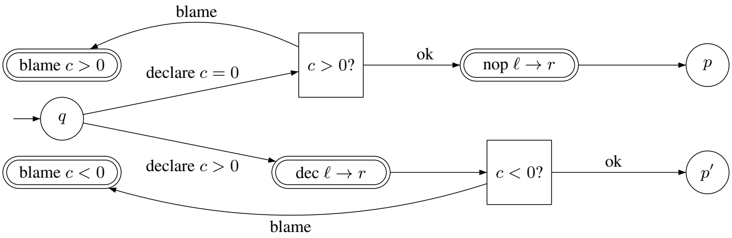

Fig. 6. q : if $c = 0$ then goto p else $c := c - 1$ goto p'

The Blame Gadgets. The role of the blame gadgets is to make sure that the assumptions on ℓ, r and g_s are kept in the simulation and to make sure that the zero testing is honestly simulated. There are six blame gadgets. Four for the honest simulation of r, ℓ and g_s, and two for the zero testing (one for $c > 0$ and one for $c < 0$). The gadgets are described in Figures 7-12. In the blame $r \ll 0$ and blame $\ell \ll 0$ gadgets the play immediately continues to the reset gadget. The concept of the other four gadgets is similar and hence we describe only the blame $r \ll |g_s|$ gadget. We note that in an honest simulation we have $avg(r), avg(\ell), avg(c_+), avg(c_-) \gtrapprox 0$ in every round. Hence, if player 1 honestly simulates M and M does not halt, then the winning condition is satisfied. The $r \ll |g_s|$ blame gadget is described in Figure 12. If the gadget is invoked and $r \ll |g_s|$, then player 2 can loop on the first state until $r \ll 0$ and still have $g_s \ll 0$. If $r \approx g_s$, then whenever we have $r \ll 0$ we will also have $g_s \gtrapprox 0$, and thus the winning objective is still satisfied. We note that player 2 should eventually exit the blame gadget, since otherwise he will lose the game.

The Reset Gadget. The role of the reset gadget is to assign the following values for the dimensions: $avg(\ell) \approx 0, avg(r) \approx 1, avg(g_s) \approx -1, avg(c_-) \approx avg(c_+) \approx 1, avg(g_c) \approx -1$. The gadget is described in Figure 13. We construct the gadget is such way that each of the players can enforce the above values (player 2 by looping enough times on the first state, player 1 by looping enough time on his two states). But the construction only gives this option to the players and it does not *punish* a player if he acts differently. However, the game graph is constructed in such way that if:

- M does not halt and in the reset gadget, at least one of the players, correctly resets the values, then player 1 wins.

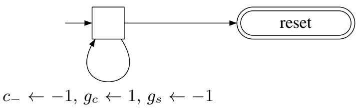

Fig. 7. blame $c > 0$ gadget

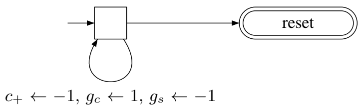

Fig. 8. blame $c < 0$ gadget

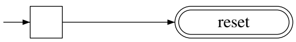

Fig. 9. blame $\ell \ll 0$ gadget

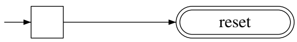

Fig. 10. blame $r \ll 0$ gadget

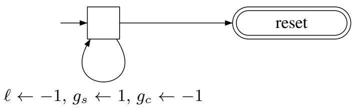

Fig. 11. blame $\ell \ll |g_s|$ gadget

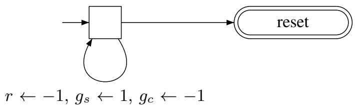

Fig. 12. blame $r \ll |g_s|$ gadget

- M halts and in the reset gadget (at least one of the players) correctly reset the values, then player 2 wins.

Hence, if M does halts, then player 2 winning strategy will make sure that the reset assigns correct values, and if M does not halt, then we can rely on player 1 to reset the values. We note that player 2 will not stay forever in his state (otherwise he will lose). In order to make sure that player 1 will not stay forever in one of his states we introduce two *liveness dimensions*, namely x and y. In the simulation and blame gadgets they get 0 values. But if player 1 remains forever in one of his two states in the reset gadget, then either x or y will have negative lim-sup value and player 1 will lose. Hence, in the reset gadget, player 1 should not only reset the values, but also assign a positive value for y and then a positive value for x.

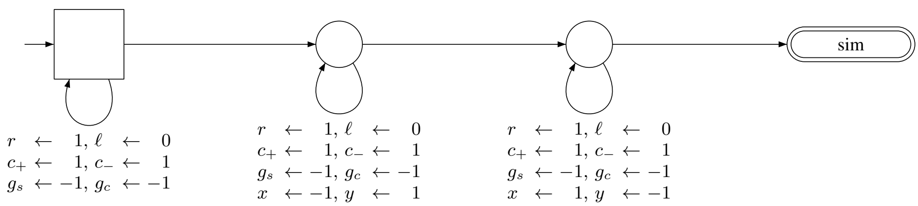

Fig. 13. Reset gadget

Correctness of the Reduction. We claim that player 1 has a winning strategy if and only if the machine M does not halt. We first summarize the (informal) invariants that we described in the construction of the reduction. Then, we prove that if M halts, then player 2 has a winning strategy, and then we prove the converse direction (the proofs are informal, and formal proofs are given in Section 4).

Summary of Invariants. We first describe the *reset invariants* that hold each time the play leaves the reset gadget (or equivalently, each time the sim gadget is invoked). The reset invariants for the side dimensions are: $avg(g_s) \approx -1$, $avg(r) \approx 1$, $avg(\ell) \approx 0$, and for the counter dimensions the invariants are: $avg(c_+) \approx 1, avg(c_-) \approx 1, avg(g_c) \approx -1$. We now describe the *sim invariants* that hold whenever the play is in the sim gadget (in all rounds that are played in the sim gadget and also before the first round that is played in the sim gadget). The sim invariants for the side dimensions are: When the play is in a right state (i.e., in a state of Q) then $r \approx |g_s|$ and $\ell \approx 0$. When the play is in a left state (i.e., in a state of P) then $\ell \approx |g_s|$ and $r \approx 0$.

The next claim follows from the sim invariants: whenever the play is in a state from Q or P (i.e., after the machine step was simulated), then $c_+ \approx |g_c| + |g_s|c$ and $c_- \approx |g_c| - |g_s|c$, where c is the current value of the counter according to the simulation steps (i.e., c is the value of the number of times the increment gadget was invoked minus the number of times that the decrement gadget was invoked from the beginning of the current invocation of the sim gadget). Informally, the proof of the claim follows by the fact that according to the sim invariants every step of the machine is simulated by a sub-play of length $|g_s|$ and by the fact that in the increment gadget the dimension c_+ is incremented by 2 while $|g_c|$ is incremented by 1 (and similar arguments can be applied for the decrement gadget and for dimension c_-). We formally prove the claim in Section 4.

Another simple consequence of the sim invariants is that $r + \ell \approx |g_s|$ in every round in the sim gadget. Indeed, whenever in a right or left state the equality holds directly from the invariants, and in every transition of the sim gadget the sum of weights of dimension ℓ and r is zero.

If M Halts, Then Player 2 Wins. The winning strategy for player 2 is as follows: In the reset gadget make sure that the *reset invariants* are satisfied. This is done by looping the first state of the reset gadget for enough rounds. In the sim gadget, whenever the *sim invariants* are not fulfilled or whenever player 1 cheats a zero-test, then player 2 invokes a blame gadget. If the sim invariants are fulfilled and player 1 does not cheat a zero test, then it must be the case that the game reaches state q_f, and in that case player 2 wins. Otherwise, we claim the player 2 wins.

We first prove that if player 1 violates the sim invariants infinitely often, then the winning condition is violated. W.l.o.g we assume that the first sim invariant is violated infinitely often and the proof for the second invariant is similar. By the assumption infinitely often the play is in a right state and either $r \gg |g_s|$ and $\ell \ll 0$ or $r \ll |g_s|$ and $\ell \gg 0$. If $r \gg |g_s|$, then $\ell \ll 0$ and it follows that in the last round that the state $\ell \ll 0?$ was visited, the value of ℓ was much smaller than 0. Hence, player 2 invoked the $\ell \ll 0$ blame gadget and the play immediately continued to the reset gadget. If this happens infinitely often then $\underline{\ell} < 0$ while $\overline{g_s} < 0$ (as g_s remains negative in the blame $\ell \ll 0$ gadget and never increases in the sim and reset gadgets) and the winning condition is violated. If $r \ll |g_s|$, then player 2 invokes the blame gadget and loop the first state until $r \ll 0$. As $r \ll |g_s|$ we still have $g_s \ll 0$, and thus $\underline{r} < 0$ while $\overline{g_s} < 0$ and the condition is violated.

We now assume that the sim invariants are violated only finitely often (for simplicity we assume that they are never violated) and we assume that infinitely often player 1 cheats the zero-test before the play reaches q_f. W.l.o.g we assume that player 1 infinitely often declares $c = 0$ while the actual value of c is positive (and the proof for the second cheat is similar). In this case, as $c_- \approx |g_c| - |g_s|c$, we have $c_- \ll |g_c|$. Hence, in the blame $c > 0$ gadget player 2 loops the first state until $c_- \ll 0$. As $c_- \ll |g_c|$ it still holds that $g_c \ll 0$. Hence, $\underline{c_-} < 0$ while $\overline{g_c} < 0$ and the condition is violated.

To conclude, if the invariants are not maintained or player 1 does not honestly simulate the zero-tests, then in each simulation, the guard dimensions have negative average weights, while at least one of the dimensions ℓ, r, c_+ or c_- has a negative average weight in the blame gadget. Hence, we get that $\overline{g_s}, \overline{g_c} < 0$ and $\underline{\ell} < 0$ or $\underline{r} < 0$ or $\underline{c_-} < 0$ or $\underline{c_+} < 0$. Hence, the winning condition is not satisfied and player 2 is the winner.

If M does not Halt, Then Player 1 Wins. The winning strategy is to honestly simulate M while maintaining the *sim invariants* and the *reset invariants*. If player 2 never invokes the blame gadget, namely, the play stays forever in the sim gadget, then the winning condition is satisfied. Indeed, in the sim gadget g_s, c_+, c_-, x and y are never decremented, thus their mean-payoff value is at least zero and the winning condition is satisfied. Otherwise, after every invocation of the blame gadget, if a *side blame gadget* was invoked, then either the average value of r and ℓ is non-negative or the value of the guard dimension g_s is non-negative. Indeed, if the sim invariants are maintained, then before a blame $\ell \ll |g_s|$ gadget is invoked we have $\ell \approx |g_s|$. Hence, if in the gadget we have $\ell < 0$, then it must be the case that $g_s \geq 0$. Thus, eventually, we get that $\underline{r}, \underline{\ell} \geq 0$ or $\overline{g_s} \geq 0$. Similarly, when a $c > 0$ gadget is invokes, we have $c = 0$ and thus $c_- \approx |g_c|$, and thus in the gadget either c_- is non-negative or g_c is non-negative (and similar arguments hold for the $c < 0$ gadget and for c_+). Hence, we get that $\underline{c_+}, \underline{c_-} \geq 0$ or $\overline{g_c} \geq 0$. Thus, the winning condition is satisfied, and player 1 is the winner.

4 Detailed Proof

In the previous section we accurately described the reduction, and only the proof of the correctness was informal. In this section we give a precise proof for the correctness of the reduction, namely, we formally describe player-2 winning strategy in the case that M halts (Subsection 4.1), and player-1 winning strategy in the case that M does not halt (Subsection 4.2). In Subsection 4.3 we extend the reduction to two-counter machine.

Terminology. In the next subsections we use the next terminology and definitions:

A *round* is a round in the game graph (i.e., either player-1 or player-2 move).

A *simulation step* denotes all the rounds that are played in a transition gadget (i.e., in a nop,inc or dec gadget). Formally, a simulation step is a sub-play that begins and ends in a node from $P \cup Q$ (i.e., a left or a right state) and visits exactly one time in a left state and exactly one time in a right state.

A *simulation session* is a sub-play that begins in an invocation of the sim gadget and ends before (or exactly when) the play leaves the sim gadget. The *first i simulation steps* of a simulation session is a sub-play that begins in an invocation of the sim gadget and ends after i simulation steps were played.

A *loop* in a transition gadget is a two round sub-play in the gadget that consists of the loop that is formed by the first state and the state beneath it.

The *total number of rounds* is the total number of rounds (moves) from the beginning of the play. We say that the average weight of dimension d in round i is a, and we denote $avg(d) = a$, if the value of dimension d in round i is $a \cdot i$ (i.e., the average weight of d from the beginning of the play up to round i is a). Given a play prefix of length i, we say player-2 can achieve $avg(d_1) \leq a_1$ while maintaining $avg(d_2) \leq a_2$, for dimensions d_1, d_2 and thresholds a_1, a_2, if player 2 has a strategy to extend the play prefix in such way that in some round $j \geq i$ it holds that $avg(d_1) \leq a_1$ and in every round k such that $i \leq k \leq j$ it holds that $avg(d_2) \leq a_2$.

4.1 If M Halts, Then Player 2 is the Winner

In this subsection we assume that M halts. We denote by N the number of steps after which M halts (for initial counter value 0) and we denote $\epsilon = \frac{1}{(N+1)^2}$. WLOG we assume that $N > 10$. The strategy of player 2 in the reset gadget is to achieve the following *reset invariants* (after the play leaves the gadget):

- $avg(g_s), avg(g_c) \leq -\frac{1}{2}$
- $(1 - \frac{\epsilon}{4})|g_s| \leq r \leq (1 + \frac{\epsilon}{4})|g_s|$
- $-\frac{\epsilon}{4}|g_s| \leq \ell \leq \frac{\epsilon}{4}|g_s|$
- $(1 - \frac{\epsilon}{4})|g_c| \leq c_+, c_- \leq (1 + \frac{\epsilon}{4})|g_c|$

We note that player 2 can maintain the above by looping sufficiently long time in the first state, and once the invariants are reached, player 1 cannot violate them in his states in the reset gadget (since the average value of g_s and g_c can only get closer to -1, the value of $\frac{\ell}{|g_s|}$ only gets closer to 0 and $\frac{r}{|g_s|}$, $\frac{c_-}{|g_c|}$ and $\frac{c_+}{|g_c|}$ only gets closer to 1).

The strategy of player 2 in the sim gadget is to maintain, in every step of the simulation session, the next three invariants, which we denote by the *left right invariants*:

- (Left state invariant) If the machine is in a left state, then $(1 - \epsilon)|g_s| \leq \ell \leq (1 + \epsilon)|g_s|$ and $-\epsilon|g_s| \leq r \leq \epsilon|g_s|$.
- (Right state invariant) If the machine is in a right state, then $(1 - \epsilon)|g_s| \leq r \leq (1 + \epsilon)|g_s|$ and $-\epsilon|g_s| \leq \ell \leq +\epsilon|g_s|$.
- (Minimal value invariant) In every round of a simulation session $r, \ell \geq -\epsilon|g_s|$.

We denote $\delta = \frac{1}{\frac{1}{2}+2N(1+2\epsilon)}$. We first prove that under these invariants $avg(g_s) \leq -\delta$ in every round of the play. Then we use this fact to show that if player 1 violates these invariants, then player 2 can violate $(\underline{\ell} \wedge \underline{r} \vee \overline{g_s})$, and therefore he wins.

Lemma 1. *Assume that for a given simulation session: (i) in the beginning of the session $avg(g_s) \leq -\frac{1}{2}$; (ii) no more that N steps are played in the simulation session; and (iii) the left-right invariants are maintained in the session. Then for every round in the session $avg(g_s) \leq -\delta$.*

Proof. We denote by R the number of rounds that were played before the current invocation of the simulation gadget. We claim that after simulating i steps of the machine (in the current invocation of the sim gadget), the **total** number of rounds in the play (i.e., number of rounds from the beginning of the play, not from the beginning of the current invocation) is at most $R + 2i \cdot |g_s|(1 + 2\epsilon)$. The proof is by a simple induction, and for the base case $i = 0$ the proof is trivial. For $i > 0$, we assume WLOG that the i-th transition is a left-to-right transition. Hence, before the last simulation step we had $r \geq -\epsilon|g_s|$ and after the i-th step was completed we had $r \leq (1 + \epsilon)|g_s|$. Since in every odd round of a step gadget the value of r is incremented by 1, we get that at most $2(1 + 2\epsilon)|g_s|$ rounds were played and the proof of the claim follows (and the proof for a right-to-left transition is symmetric).

Hence, after N simulation steps we have $avg(g_s) \leq \frac{g_s}{R+2N|g_s|(1+2\epsilon)}$. Since in the beginning of the sim gadget we had $avg(g_s) \leq -\frac{1}{2}$, then $R \leq \frac{|g_s|}{2}$. Hence, and since $g_s < 0$ we get $avg(g_s) \leq \frac{g_s}{\frac{|g_s|}{2}+2N|g_s|(1+2\epsilon)} = -\frac{1}{\frac{1}{2}+2N(1+2\epsilon)} = -\delta$.

We note that in every transition of a simulation session the value of g_s is not changed. Hence, $avg(g_s)$ gets the maximal value after the N-th step and the proof is complete. □

Lemma 2. *Let $\gamma = \min(\frac{\epsilon\delta}{4}, \frac{\frac{\epsilon}{4}}{1+\frac{1}{2\delta}-\frac{\epsilon}{4}})$. If player 1 violates the left-right invariants in the first N steps of a session, then player 2 can achieve in the blame gadget either $avg(r) \leq -\gamma$ or $avg(\ell) \leq -\gamma$ (or both) while maintaining $avg(g_s), avg(g_c) \leq -\gamma$.*

Proof. We first prove the assertion over the value of g_c. It is an easy observation that if at the invocation of the sim gadget $avg(g_c) \leq -\frac{1}{2}$, then it remains at most $-\frac{1}{2}$ as it gets a value of -1 in every round in a blame gadget and -1 in every odd round in a step gadget.

Next, we prove the assertion for the left-state and minimal value invariants and the proof for the right-state invariant is symmetric. Recall that the invariant consistences of four assumptions, namely, (i) $(1-\epsilon)|g_s| \leq \ell$ after a right to left transition; (ii) $\ell \leq (1+\epsilon)|g_s|$ after a right to left transition; (iii) $-\epsilon|g_s| \leq r$ in every round; and (iv) $r \leq \epsilon|g_s|$ after a right to left transition. We first prove the assertion when the first condition is violated, i.e., we assume that $\ell < (1-\epsilon)|g_s|$. If this is the case after a right-to-left transition, then player 2 will invoke the blame $\ell \ll |g_s|$ gadget after the transition ends. In the blame gadget he will traverse the self-loop for $X \cdot (1-\frac{\epsilon}{2})$ times, where X is the value of $|g_s|$ before the invocation of the blame gadget, and then he will go to the reset gadget. As a result (since in every loop ℓ is decremented by 1 and g_s is incremented by 1) we get that the value of ℓ and g_s is at most $-X \cdot \frac{\epsilon}{2}$. Before the last simulation step the left-right invariants were maintained. Hence, before the last step we had $\ell \geq -\epsilon|g_s|$ (by the left-right invariants) and thus the last step had at most $|g_s|$ rounds (as we assume that after the last step $\ell < (1-\epsilon)|g_s|$). In addition, as the invariants were maintained, by Lemma 1 we get that before the last step we had $avg(g_s) \leq -\delta$ and thus after the last step we have $avg(g_s) \leq -\frac{\delta}{2}$ (as the value of g_s is not changed in simulation steps). Hence, if R is the number of rounds before the invocation of the blame gadget, then $R \leq \frac{X}{2\delta}$. Hence, after the blame gadget ends, we have $avg(\ell), avg(g_s) \leq -\frac{X\cdot\frac{\epsilon}{2}}{R+X\cdot(1-\frac{\epsilon}{2})} \leq -\frac{X\cdot\frac{\epsilon}{2}}{\frac{X}{2\delta}+X\cdot(1-\frac{\epsilon}{2})} = -\frac{\frac{\epsilon}{2}}{1+\frac{1}{2\delta}-\frac{\epsilon}{2}}$. In addition, the value of g_s is incremented in every round of the blame gadget. Thus, if after the gadget ends we have $avg(g_s) \leq -\gamma$, then in every round in the blame gadget we also have $avg(g_s) \leq -\gamma$.

If the second condition is violated, namely, if $\ell > (1+\epsilon)|g_s|$, then we claim that it must be the case that $r < -\frac{\epsilon|g_s|}{2}$. Indeed, when the sim gadget is invoked we have $r \leq |g_s|(1+\frac{\epsilon}{4})$ and $\ell \leq |g_s|\frac{\epsilon}{4}$. In the sim gadget the value of the sum $r+\ell$ is not changed (since r is incremented only when ℓ is decremented and vice versa). Hence, the sum never exceeds $|g_s|(1+\frac{\epsilon}{2})$. Thus, if $\ell > (1+\epsilon)|g_s|$, then it must be the case that $r < -\frac{\epsilon|g_s|}{2}$. Hence, in the first round that $avg(r) \leq -\frac{\epsilon}{2}|g_s|$ player 2 can invoke the blame $r \ll 0$ gadget which leads the play to the reset gadget after exactly one move. We note that in this scenario the left-right invariants are satisfied and thus, after leaving the blame gadget by Lemma 1 we have $avg(g_s) \leq -\delta$ and as $r \leq -\frac{\epsilon|g_s|}{2}$ we get that $avg(r) \leq -\frac{\epsilon\delta}{2}$.

If the third condition is violated, namely, if $r < -\epsilon|g_s|$, then it must be the case that the condition is first violated in a left to right transition (since in a right to left transition

r is incremented) and the proof follows by the same arguments as in the proof of the second case.

Finally, if the fourth condition is violated, namely, if $r > \epsilon|g_s|$, then by analyzing the sum $r + \ell$ we get that $\ell \le (1 - \frac{\epsilon}{2})|g_s|$. We repeat the same analysis as in the case where the first invariant is violated (i.e., when $\ell \le (1-\epsilon)|g_s|$) and get that $avg(g_s), avg(r) \le -\frac{\frac{\epsilon}{4}}{1+\frac{1}{2\delta}-\frac{\epsilon}{4}}$. The proof is complete. □

By Lemma 2, if player 2 maintains the reset invariant in the reset gadget, then other than finitely many simulation sessions, in every simulation session player 1 must satisfy the left-right invariants. Otherwise, we get that infinitely often the average value of either r or ℓ is at most $-\gamma$ while the average value of g_s is always at most $-\gamma$. Hence $\overline{g_s} < 0$ and either $\underline{r} < 0$ or $\underline{\ell} < 0$ and thus the condition $(\underline{\ell} \wedge \underline{r} \vee \overline{g_s})$ is violated and therefore player 1 is losing.

In the next three lemmas we prove that player 1 must honestly simulates the zero-testing. The first lemma is a simple corollary of the left-right invariants.

Lemma 3. *Under the left-right invariants, in the dec, inc and nop gadgets, player 1 follows the loop of the first state at most $|g_s|(1+2\epsilon)$ times and at least $|g_s|(1-2\epsilon)$ times.*

The next lemma shows the correlation between g_c and c_+ and c_-.

Lemma 4. *Let $\#inc$ (resp., $\#dec$) be the number of times that the inc (dec) gadget was visited (in the current simulation session), and we denote $c = \#inc - \#dec$ (namely, c is the actual value of the counter in the counter machine M). Then under the left-right invariants, in the first N steps of the simulation session we always have $c_+ \le |g_c|(1+\epsilon) + c|g_s| + \frac{|g_s|}{2}$ and $c_- \le |g_c|(1+\epsilon) - c|g_s| + \frac{|g_s|}{2}$.*

Proof. We prove the claim of the lemma for c_+ and the proof for c_- is symmetric. Let X be the value of $|g_c|$ when the sim gadget is invoked. By the reset invariants we get that $c_+ \le X(1+\frac{\epsilon}{4})$. By Lemma 3 we get that every visit in the inc gadget contributes at most $|g_s|(1+2\epsilon)$ more to c_+ than its contribution to $|g_c|$ and every visit in the dec contributes at least $|g_s|(1-2\epsilon)$ more to $|g_c|$ than its contribution to c_+. Hence,

$$c_+ \le X(1+\tfrac{\epsilon}{4}) + (|g_c| - X) + \#inc \cdot |g_s|(1+2\epsilon) - \#dec \cdot |g_s|(1-2\epsilon) =$$
$$|g_c| + \epsilon X + (\#inc - \#dec)|g_s|(1+2\epsilon) + 4\epsilon|g_s| \cdot \#dec$$

We recall that $c = (\#inc - \#dec)$, and observe that $X \le |g_c|$, and that $\#dec \le N$ and thus $\epsilon \cdot \#dec < \frac{1}{10}$. Hence, we get that $c_+ \le |g_c|(1+\epsilon) + c|g_s| + \frac{|g_s|}{2}$. □

The next lemma suggests that player 1 must honestly simulate the zero-tests.

Lemma 5. *If the reset and left-right invariants hold, then for $\gamma = \min(\frac{1}{20N}, \frac{\delta}{8})$ the following hold: (i) if the blame $c < 0$ gadget is invoked and $c < 0$ then player 2 can achieve $avg(c_+) \le -\gamma$ while maintaining $avg(g_s), avg(g_c) \le -\gamma$; and (ii) if the blame $c > 0$ gadget is invoked and $c > 0$ then player 2 can achieve $avg(c_-) \le -\gamma$ while maintaining $avg(g_s), avg(g_c) \le -\gamma$.*

Proof. We prove the first item of the lemma and the proof for the second item is symmetric. Suppose that $c < 0$ (i.e., $c \le -1$) when blame $c < 0$ gadget is invoked. Let X and Y be the values of $|g_c|$ and $|g_s|$ before the invocation of the blame gadget. Then by

Lemma 4, before the invocation we have $c_+ \leq X(1+\epsilon) - \frac{Y}{2}$. Hence, by traversing the loop of the first state of the blame $c < 0$ gadget for $X(1+\epsilon) - \frac{Y}{4}$ times we get $c_+ \leq -\frac{Y}{4}$ and $g_c \leq \epsilon X - \frac{Y}{4}$. Let R be the number of rounds that were played from the beginning of the play (and not just from the beginning of the current invocation of the sim gadget). Since g_c is decremented by at most 1 in every round we get that $X(1+\epsilon) - \frac{Y}{4} \leq 2X \leq 2R$. By lemma 1 we have $\frac{Y}{R} \leq -\delta$. Hence, $avg(c_+) \leq \frac{c_+}{2R} \leq -\frac{Y}{8R} \leq -\frac{\delta}{8}$. Similarly, since $\frac{X}{R}$ is bounded by 1, we have $avg(g_c) \leq \frac{\epsilon X}{2R} - \frac{\delta}{8} \leq \frac{\epsilon}{2} - \frac{\delta}{8}$. Recall that $\delta = \frac{1}{\frac{1}{2}+2N(1+2\epsilon)}$. Hence, $avg(g_c) \leq \frac{2\epsilon+4N\epsilon+8\epsilon^2-1}{8(\frac{1}{2}+N(1+2\epsilon))}$ and since $\epsilon = \frac{1}{(N+1)^2}$ and $N > 10$ we get that $avg(g_c) \leq -\frac{1}{20N}$. Note that g_c is only incremented in the blame gadget. Thus, as $avg(g_c) \leq -\gamma$ after the last round of the blame gadget we get that $avg(g_c) \leq -\gamma$ in all the rounds that are played in the blame gadget. The value of g_s was at most $-\delta R$ before the blame gadget, and in the blame gadget g_s is decreased by 1 in every round. Hence $avg(g_s) \leq -\delta$ in every round of the blame gadget and the proof follows by taking $\gamma = \min(\frac{1}{20N}, \frac{\delta}{8})$. □

We are now ready to prove one side of the reduction.

Proposition 1. *If the counter machine M halts, then player 2 has a winning strategy for violating $(\underline{\ell} \wedge \underline{r} \vee \overline{g_s}) \wedge (\underline{c_+} \wedge \underline{c_-} \vee \overline{g_c}) \wedge \overline{x} \wedge \overline{y}$. Moreover, if M halts then there exists a constant $\zeta > 0$ that depends only on M such that player 2 has a winning strategy for violating $(\underline{\ell} \geq -\zeta \wedge \underline{r} \geq -\zeta \vee \overline{g_s} \geq -\zeta) \wedge (\underline{c_+} \geq -\zeta \wedge \underline{c_-} \geq -\zeta \vee \overline{g_c} \geq -\zeta) \wedge \overline{x} \geq -\zeta \wedge \overline{y} \geq -\zeta$.*

Proof. Suppose that M halts and let N be the number of steps that M runs before it halts (for an initial counter value 0). Player-2 strategy is to (i) maintain the reset-invariants; (ii) whenever the left-right invariants are violated, he invokes a side blame gadget; (iii) whenever the zero-testing is dishonest, he activates the corresponding blame gadget (either $c > 0$ or $c < 0$); and (iv) if q_f is reached, he stays there forever. The correctness of the construction is immediate by the lemmas above. We first observe that it is possible for player 2 to satisfy the reset-invariants and that if player 1 stays in the reset gadget forever, then he loses.

Whenever the left-right invariant is violated, then the average weight of r and/or ℓ is negative, while the average weight of g_s and g_c remains negative. Hence, if in every simulation session player 1 violates the left-right invariants in the first N steps we get that the condition is violated since $\overline{g_s} \leq -\gamma$ and either $\underline{r} \leq -\gamma$ or $\underline{\ell} \leq -\gamma$. Hence, we may assume that these invariants are kept in every simulation session.

Whenever the zero-testing is dishonest (while the left-right invariants are satisfied), then by Lemma 5, player 2 can invoke a counter blame gadget and achieve negative average for either c_+ or c_- while maintaining g_c and g_s negative. If in every simulation session player 1 is dishonest in zero-testing, then we get that either $\underline{c_-} \leq -\gamma$ or $\underline{c_+} \leq -\gamma$ while $\overline{g_c} \leq -\gamma$ and the condition is violated. Hence, we may assume that player 1 honestly simulates the zero-tests. Finally, if the transitions of M are properly simulated, then it must be the case the state q_f is reached and when looping this state forever player 1 loses (since $\overline{x} \leq -1 < 0$). □

4.2 If M does not Halt, Then Player 1 is the Winner

Suppose that M does not halt. A winning strategy of player 1 in the reset gadget is as following: Let i be the number of times that the reset gadget was visited, and we denote $\epsilon_i = \frac{1}{i+10}$. Similarly to player-2 strategy in Subsection 4.1, player-1 strategy in the reset gadget is to achieve the following invariants (after the play leaves the gadget): (i) $avg(g_s), avg(g_c) \leq -\frac{1}{2}$; (ii) $(1-\frac{\epsilon_i}{4})|g_s| \leq r \leq (1+\frac{\epsilon_i}{4})|g_s|$; (iii) $-\frac{\epsilon_i|g_s|}{4} \leq \ell \leq \frac{\epsilon_i|g_s|}{4}$; and (iv) $(1-\frac{\epsilon_i}{4})|g_c| \leq c_+, c_- \leq (1+\frac{\epsilon_i}{4})|g_c|$. To satisfy these invariants, he follows the self-loop of his first state until $avg(y) \geq 0$ and then follows the self-loop of the second state until the invariants are fulfilled and $avg(x) \geq 0$. In the sim gadget, player-1 strategy is to simulate every nop,inc and dec step by following the self-loop in the corresponding gadget for $|g_s|$ rounds, and to honestly simulate the zero-tests..

We denote the above player-1 strategy by τ. The next two lemmas show the basic properties of a play according to τ, and that player 2 loses if he invokes the blame gadgets infinitely often.

Lemma 6. *In any play according to τ, after the reset gadget was visited for i times, in the sim gadget we always have: (i) in a right state: $r \geq -\epsilon_i|g_s|, \ell \geq (1-\epsilon_i)|g_s|$ and in a left state $\ell \geq -\epsilon_i|g_s|, r \geq (1-\epsilon_i)|g_s|$; (ii) in every round of the simulation session $r, \ell \geq -\epsilon_i|g_s|$; and (iii) $c_+ \geq (1-\epsilon_i)|g_c| + c|g_s|$ and $c_- \geq (1-\epsilon_i)|g_c| - c|g_s|$, where $c = \#inc - \#dec$ in the current invocation of the sim gadget.*

Lemma 7. *In a play prefix consistent with τ, in every round that is played in a blame gadget: (1) In the blame $\ell \ll 0$ and blame $r \ll 0$ gadgets: $avg(\ell), avg(r) \geq -\epsilon_i$. (2) In blame $\ell \ll \||g_s|$ gadget: if $avg(\ell) \leq -\epsilon_i$, then $avg(g_s) \geq -\epsilon_i$. (3) In blame $r \ll |g_s|$ gadget: if $avg(r) \leq -\epsilon_i$, then $avg(g_s) \geq -\epsilon_i$. (4) In the blame $c < 0$ gadget: if $avg(c_+) \leq -\epsilon_i$, then $avg(g_c) \geq -\epsilon_i$. (5) In the blame $c > 0$ gadget: if $avg(c_-) \leq -\epsilon_i$, then $avg(g_c) \geq -\epsilon_i$. Where i is the number of times that the reset gadget was visited.*

We are now ready to prove the τ is a winning strategy.

Proposition 2. *If M does not halt, then τ is a winning strategy.*

Proof. In order to prove that τ satisfies the condition $(\underline{\ell} \wedge \underline{r} \vee \overline{g_s}) \wedge (\underline{c_+} \wedge \underline{c_-} \vee \overline{g_c}) \wedge \overline{x} \wedge \overline{y}$ it is enough to prove that when playing according to τ, for any constant $\delta > 0$ the condition $(\underline{\ell} \geq -\delta \wedge \underline{r} \geq -\delta \vee \overline{g_s} \geq -\delta) \wedge (\underline{c_+} \geq -\delta \wedge \underline{c_-} \geq -\delta \vee \overline{g_c} \geq -\delta) \wedge \overline{x} \wedge \overline{y}$ is satisfied.

Let $\delta > 0$ be an arbitrary constant and in order to prove the claim we consider two distinct cases: In the first case, player 2 strategy will invoke the blame gadgets only finitely many times. Hence, there is an infinite suffix that is played only in either a blame gadget, the reset gadget or the sim gadget and in such suffix player 2 loses.

In the second case we consider, player 2 always eventually invokes a blame gadget. Since a blame gadget is invoked infinitely many times we get that the reset gadget is invoked infinitely often, and thus $\overline{x}, \overline{y} \geq 0$. In addition, the sim gadget is invoked infinitely often. Let i be the minimal index for which $\epsilon_i \leq \delta$. By Lemmas 6 and 7 we get that after the i-th invocation of the sim gadget, in every round (i) either $avg(\ell) \geq -\epsilon_i \wedge avg(r) \geq -\epsilon_i$ or $avg(g_s) \geq -\epsilon_i$; and (ii) either $avg(c_+) \geq -\epsilon_i \wedge avg(c_-) \geq -\epsilon_i$ or $avg(g_c) \geq -\epsilon_i$. (A detailed proof is given in the technical report.) Thus, as of certain round, either $avg(\ell)$ and $avg(r)$ are always at least $-\epsilon_i$, or infinitely often $avg(g_s) \geq -\epsilon_i$. Hence, $(\underline{\ell} \geq -\epsilon_i \wedge \underline{r} \geq -\epsilon_i \vee \overline{g_s} \geq -\epsilon_i)$ is satisfied and similarly $(\underline{c_+} \geq -\epsilon_i \wedge \underline{c_-} \geq -\epsilon_i \vee \overline{g_c} \geq -\epsilon_i)$ is satisfied. The proof is complete. □

4.3 Extending the Reduction to Two-counter Machine

When M is a two-counter machine, we use 4 dimensions for the counters, namely $c^1_+, c^1_-, c^2_+, c^2_-$ and one guard dimension g_c. The winning condition is $(\underline{\ell} \wedge \underline{r} \vee \overline{g_s}) \wedge (\underline{c^1_+} \wedge \underline{c^1_-} \wedge \underline{c^2_+} \wedge \underline{c^2_-} \vee \overline{g_c}) \wedge \overline{x} \wedge \overline{y}$. In a nop gadget all four dimensions $c^1_+, c^1_-, c^2_+, c^2_-$ get a value of 1 in the self-loop. When a counter c_i (for $i = 1, 2$) is incremented (resp., decremented), then counter c^i_+ and c^i_- are assigned with weights according to the weights of c_+ and c_- in the inc (dec) gadget that we described in the reduction for a one counter machine, and c^{3-i}_+, c^{3-i}_- are assigned with weights according to a nop gadget.

The proofs of Proposition 1 and Proposition 2 easily scale to a two-counter machine. Hence, the undecidability result is obtained.

Theorem 1. *The problem of deciding who is the winner in a multidimensional mean-payoff game with ten dimensions is undecidable.*

The winning condition that we use in the reduction can be encoded also by mean-payoff expressions [6]. Hence, games over mean-payoff expressions are also undecidable.

References

1. Alur, R., Degorre, A., Maler, O., Weiss, G.: On omega-languages defined by mean-payoff conditions. In: de Alfaro, L. (ed.) FOSSACS 2009. LNCS, vol. 5504, pp. 333–347. Springer, Heidelberg (2009)
2. Bohy, A., Bruyère, V., Filiot, E., Raskin, J.-F.: Synthesis from LTL specifications with mean-payoff objectives. In: Piterman, N., Smolka, S.A. (eds.) TACAS 2013 (ETAPS 2013). LNCS, vol. 7795, pp. 169–184. Springer, Heidelberg (2013)
3. Boker, U., Chatterjee, K., Henzinger, T.A., Kupferman, O.: Temporal specifications with accumulative values. In: LICS (2011)
4. Brázdil, T., Brozek, V., Chatterjee, K., Forejt, V., Kucera, A.: Two views on multiple mean-payoff objectives in markov decision processes. In: LICS (2011)
5. Brázdil, T., Chatterjee, K., Kučera, A., Novotný, P.: Efficient controller synthesis for consumption games with multiple resource types. In: Madhusudan, P., Seshia, S.A. (eds.) CAV 2012. LNCS, vol. 7358, pp. 23–38. Springer, Heidelberg (2012)
6. Chatterjee, K., Doyen, L., Edelsbrunner, H., Henzinger, T.A., Rannou, P.: Mean-payoff automaton expressions. In: Gastin, P., Laroussinie, F. (eds.) CONCUR 2010. LNCS, vol. 6269, pp. 269–283. Springer, Heidelberg (2010)
7. Chatterjee, K., Randour, M., Raskin, J.-F.: Strategy synthesis for multi-dimensional quantitative objectives. In: Koutny, M., Ulidowski, I. (eds.) CONCUR 2012. LNCS, vol. 7454, pp. 115–131. Springer, Heidelberg (2012)
8. Chatterjee, K., Velner, Y.: Mean-payoff pushdown games. In: LICS (2012)
9. Chatterjee, K., Velner, Y.: Hyperplane separation technique for multidimensional mean-payoff games. In: D'Argenio, P.R., Melgratti, H. (eds.) CONCUR 2013 – Concurrency Theory. LNCS, vol. 8052, pp. 500–515. Springer, Heidelberg (2013)
10. Degorre, A., Doyen, L., Gentilini, R., Raskin, J.-F., Toruńczyk, S.: Energy and mean-payoff games with imperfect information. In: Dawar, A., Veith, H. (eds.) CSL 2010. LNCS, vol. 6247, pp. 260–274. Springer, Heidelberg (2010)
11. Doyen, L.: Games and automata: From boolean to quantitative verification. Memoire d'habilitation, ENS Cachan, France (2012)
12. Tomita, T., Hiura, S., Hagihara, S., Yonezaki, N.: A temporal logic with mean-payoff constraints. In: Aoki, T., Taguchi, K. (eds.) ICFEM 2012. LNCS, vol. 7635, pp. 249–265. Springer, Heidelberg (2012)

13. Velner, Y.: The complexity of mean-payoff automaton expression. In: Czumaj, A., Mehlhorn, K., Pitts, A., Wattenhofer, R. (eds.) ICALP 2012, Part II. LNCS, vol. 7392, pp. 390–402. Springer, Heidelberg (2012)
14. Velner, Y.: Finite-memory strategy synthesis for robust multidimensional mean-payoff objectives. In: CSL-LICS (2014)
15. Velner, Y., Chatterjee, K., Doyen, L., Henzinger, T.A., Rabinovich, A., Raskin, J.-F.: The complexity of multi-mean-payoff and multi-energy games. CoRR (2012)
16. Velner, Y., Rabinovich, A.: Church synthesis problem for noisy input. In: Hofmann, M. (ed.) FOSSACS 2011. LNCS, vol. 6604, pp. 275–289. Springer, Heidelberg (2011)
17. Yaron Velner. Robust multidimensional mean-payoff games are undecidable. CoRR (2015)