

Quantitative Reductions and Vertex-Ranked Infinite Games^{*,**}

Alexander Weinert^a

^a*German Aerospace Center (DLR), Institute for Software Technology,
Linder Höhe, 51147 Köln, Germany*

Abstract

We introduce quantitative reductions, a novel technique for structuring the space of quantitative games and solving them that does not rely on a reduction to qualitative games. We show that such reductions exhibit the same desirable properties as their qualitative counterparts and that they additionally retain the optimality of solutions. Moreover, we introduce vertex-ranked games as a general-purpose target for quantitative reductions and show how to solve them. In such games, the value of a play is determined only by a qualitative winning condition and a ranking of the vertices.

We provide quantitative reductions of quantitative request-response games and of quantitative Muller games to vertex-ranked games, thus showing EXPTIME-completeness of solving the former two kinds of games. In addition, we exhibit the usefulness and flexibility of vertex-ranked games by showing how to use such games to compute fault-resilient strategies for safety specifications. This work lays the foundation for a general study of fault-resilient strategies for more complex winning conditions.

*Supported by the project “TriCS” (ZI 1516/1-1) of the German Research Foundation (DFG) and the Saarbrücken Graduate School of Computer Science.

**This work was mainly performed while the author was employed at the Reactive Systems Group, Saarland University, Germany.

This work is based on work first presented at GandALF ’18 [1]. In particular, the proofs of Theorem 4, Theorem 7, and Lemma 17 have been revised and extended in comparison to that version. Furthermore, Theorem 5, Remark 6, and Section 5.2 have been added. Finally, the proofs of all lemmas and theorems are now included in the main text.

1. Introduction

The study of quantitative infinite games has garnered great interest lately, as they allow for a much more fine-grained analysis and specification of reactive systems than classical qualitative games [2–8]. While there exists previous work investigating quantitative games, the approaches to solving them usually rely on ad-hoc solutions that are tailor-made to the considered winning condition. Moreover, quantitative games are usually solved by reducing them to a qualitative game in a first step, hardcoding a certain value of interest during the reduction. In particular, to the best of our knowledge, there exists no general framework for the analysis of such games that is analogous to the existing one for qualitative games. In this work, we introduce such a framework that disentangles the study of quantitative games from that of qualitative ones.

Qualitative infinite games have been applied successfully in the verification and synthesis of reactive systems [9–12]. They have given rise to a multitude of algorithms that ascertain system correctness and that synthesize correct-by-construction systems. In such a game, two players, called Player 0 and Player 1, move a token in a directed graph. After infinitely many moves, the resulting sequence of vertices is evaluated and one player is declared the winner of the play. For example, in a qualitative request-response game [13], the goal for Player 0 is to ensure that every visit to a vertex denoting some request is eventually followed by a visit to a vertex denoting an answer to that request. In order to solve qualitative games, i.e., to determine a winning strategy for one player, one often reduces a complex game to a potentially larger, but conceptually simpler one. For example, in a multi-dimensional request-response game, i.e., in a request-response game in which there exist multiple conditions that can be requested and answered, one stores the set of open requests and demands that every request is closed at infinitely many positions. As this is a Büchi condition, which is much simpler than the request-response condition, one is able to reduce request-response games to Büchi games.

In recent years the focus of research has shifted from the study of qualitative games, in which one player is declared the winner of a given play, to that of quantitative games, in which the resulting play is assigned some value or cost. Such games allow, for example, modeling systems in which requests have to be answered within a certain number of steps [4, 6, 14–16], systems with one or more finite resources which may be drained and charged [17–20],

or scenarios in which each move incurs a certain cost for either player [5, 8].

In general, Player 0 aims to minimize the cost of the resulting play, i.e., to maximize its value, while Player 1 seeks to maximize the cost, thus minimizing the value. In a quantitative request-response game, for example, it is the goal of Player 0 to minimize the number of steps taken between requests and their corresponding answers. The typical questions asked in the context of such games are “Does there exist an upper bound on the time between requests and responses that Player 0 can ensure?” [6, 14, 16, 21], “Can Player 0 ensure an average cost per step greater than zero?” [8], “What is the minimal time between requests and responses that Player 0 can ensure?” [7], or “What is the minimal average level of the resource that Player 0 can ensure without it ever running out?” [17]. The former two questions can be seen as boundedness questions, while the latter two are asking for optimal solutions.

Such decision problems are usually solved by fixing some bound b on the cost of the resulting plays and subsequently reducing the problem of finding a strategy for Player 0 that enforces a cost of at most b in the quantitative game to the problem of solving a qualitative game, hardcoding the fixed b in the process. The problem of deciding whether or not Player 0 has such a strategy is called the (b) -threshold problem.

For example, in order to determine the winner in a quantitative request-response game as described above for some bound b , we construct a Büchi game in which every time a request is opened, a counter for that request is started which counts up to the bound b and is reset if the request is answered. Once any counter exceeds the value b , we move to a terminal position indicating that Player 0 has lost. We then require that every counter is inactive infinitely often, which is again a Büchi condition and thus much simpler than the original quantitative request-response condition. Thus, Player 0 wins the resulting qualitative game if and only if she can ensure that every request is answered within at most b steps in the quantitative game.

Such reductions are usually specific to the problem being addressed. Furthermore, they immediately abandon the quantitative aspect of the game under consideration, as the bound is hardcoded during the first step of the analysis. Thus, even when only changing the bound one is interested in, the reduction has to be recomputed and the resulting qualitative game has to be solved from scratch. In our request-response example, if one is interested in deciding the b' -threshold problem for some $b' \neq b$, one constructs a new Büchi game for the bound b' . This game is then solved independently of the one previously computed for the bound b .

In this work, we lift the concept of reductions for qualitative games to quantitative games. Such quantitative reductions enable the study of a multitude of optimization problems for quantitative games in a way similar to decision problems for qualitative games. When investigating quantitative request-response games using quantitative reductions, for example, we only compute a single, simpler quantitative game and subsequently check this game for a winning strategy for Player 0 for any bound b . If she has such a strategy in the latter game, the quantitative reduction yields a strategy for her satisfying the same bound in the former one.

In general, we retain the intuitive property of reductions for qualitative games: Using quantitative reductions, the properties of a complex quantitative game can be studied by investigating a potentially larger, but conceptually simpler quantitative game.

Contributions We present the first framework for reductions between quantitative games and we provide vertex-ranked games as general-purpose targets for such reductions. Moreover, we show tight bounds on the complexity of solving vertex-ranked games with respect to a given bound.

Subsequently, we provide three examples illustrating the use of the concepts introduced in this work: First, we define quantitative request-response games and solve them using quantitative reductions to vertex-ranked games. Second, we show how to solve quantitative Muller games as defined by McNaughton [22] via quantitative reductions to vertex-ranked safety games. Third, we illustrate the versatility of vertex-ranked games by using them to compute fault-resilient strategies for safety games with faults. We summarize our contributions with regards to solving quantitative games in Table 1 and we summarize our contributions with regards to solving quantitative games optimally in Table 2.

Structure of this work After introducing qualitative and quantitative games formally in Section 2, we define quantitative reductions in Section 3 and show that they provide a mechanism to determine the minimal bound b such that Player 0 can enforce a cost of at most b in a given quantitative game: If a game \mathcal{G} can be reduced to a game \mathcal{G}' , then we can use a strategy for Player 0 that minimizes the cost of plays in \mathcal{G}' to construct a strategy for her which minimizes the cost of plays in \mathcal{G} .

In Section 4, we define vertex-ranked games, very general classes of quan-

Game	Time	Space	Memory
Vertex-ranked sup-games	$\mathcal{O}(n) + t(\mathcal{G})$	$\mathcal{O}(n) + s(\mathcal{G})$	$\mathcal{O}(\sigma)$
Vertex-ranked lim sup-games	$\mathcal{O}(n^3 + n^2 \cdot t(\mathcal{G}))$	$\mathcal{O}(n + s(\mathcal{G}))$	$\mathcal{O}(\sigma)$
Quantitative request-response games	$\mathcal{O}(n^2 b^{2d} d^2 2^d)$	–	$\mathcal{O}(nb^d d 2^d)$
Quantitative Muller games	$\mathcal{O}((n!)^3)$	–	$\mathcal{O}((n!)^3)$

Table 1: A summary of the proofs in work concerned with solving games. For each game, n denotes the number of vertices of the game. For vertex-ranked games, \mathcal{G} denotes the underlying qualitative game while $t(|\mathcal{G}|)$ and $s(|\mathcal{G}|)$ denote the time and space required to solve that underlying game. Furthermore, σ denotes the size of a winning strategy in the corresponding qualitative game. For request-response games, d and W denote the number of request-response pairs and the largest weight assigned to any edge in the game, respectively. For the sake of brevity and consistency, we use the shorthand $b = d2^d nW$.

Game	Time	Space
Vertex-ranked sup-games	$\mathcal{O}(\log(M)(n + t(\mathcal{G})))$	$\mathcal{O}(M) + s(\mathcal{G})$
Vertex-ranked lim sup-games	$\mathcal{O}(\log(M)(n^3 + n^2 \cdot t(\mathcal{G})))$	$\mathcal{O}(n + s(\mathcal{G}))$
Quantitative request-response games	$\mathcal{O}(\log(b)(n^2 b^{2d} d^2 2^d))$	–
Quantitative Muller games	$\mathcal{O}((n!)^3)$	–

Table 2: A summary of the proofs in work concerned with solving games optimally. For vertex-ranked games, M denotes the number of different ranks assigned to vertices in the game. The remaining symbols are defined as in Table 1.

titative games that can be used as targets for quantitative reductions. The quantitative condition of such games is quite simple in that the cost of a play is determined only by a qualitative winning condition and by a ranking of the vertices of the game. If the resulting play is winning according to the qualitative condition, then its cost is given by the highest rank visited at all or visited infinitely often, depending on the particular variant of vertex-ranked games. Otherwise, the value of the play is infinite. We show that solving such vertex-ranked games is as hard as solving games with the underlying qualitative winning condition.

Finally, in Section 5 we provide three examples of the versatility of vertex-ranked games: First, we define and solve request-response games with costs via quantitative reductions. Second, we recall the definition of quantitative Muller games due to McNaughton [22] and show how to solve these games via quantitative reductions as well. Third, we discuss how to use vertex-ranked games to compute fault-resilient strategies in safety games with faults [23]. In such games, after Player 0 has picked a move, say to vertex v , a fault may occur, which overrides the choice of Player 0 and the game continues in vertex $v' \neq v$ instead. By using vertex-ranked games, we are able to compute strategies that are resilient against as many faults as possible.

2. Preliminaries

We first define notions that are common to both qualitative and quantitative games. Afterwards, we recapitulate the standard notions for qualitative games before defining quantitative games and lifting the notions for qualitative games to the quantitative case.

We denote the non-negative integers by \mathbb{N} and define $[n] = \{0, 1, \dots, n-1\}$ for every $n \geq 1$. Also, we define $\infty > n$ for all $n \in \mathbb{N}$ and $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$. Finally, for any set V , we write $|V|$ and V^ω to denote the cardinality of V and the set of all infinite sequences over V , respectively. An *arena* $\mathcal{A} = (V, V_0, V_1, E, v_I)$ consists of a finite, directed graph (V, E) , a partition (V_0, V_1) of V into the vertices of Player 0 and Player 1, and an initial vertex $v_I \in V$. The size of \mathcal{A} , denoted by $|\mathcal{A}|$, is defined as $|V|$. A *play* in \mathcal{A} is an infinite path $\rho = v_0 v_1 v_2 \dots$ through (V, E) starting in v_I . To rule out finite plays, we require every vertex to be non-terminal, i.e., we require that for every vertex $v \in V$ there exists some vertex $v' \in V$ such that $(v, v') \in E$.

A *strategy* for Player i is a mapping $\sigma: V^*V_i \rightarrow V$ that assigns to each play prefix π ending in a vertex of Player i a vertex $\sigma(\pi)$ to move to. Formally,

we require $(v, \sigma(\pi v)) \in E$ for all $\pi \in V^*$, $v \in V_i$. We say that σ is *positional* if $\sigma(\pi v) = \sigma(v)$ for every $\pi \in V^*$, $v \in V_i$. A play $v_0 v_1 v_2 \dots$ is *consistent* with a strategy σ for Player i , if $v_{j+1} = \sigma(v_0 \dots v_j)$ for all j with $v_j \in V_i$.

A *memory structure* $\mathcal{M} = (M, m_I, \text{Upd})$ for an arena (V, V_0, V_1, E, v_I) consists of a finite set M of memory states, an initial memory state $m_I \in M$, and an update function $\text{Upd}: M \times V \rightarrow M$. We extend the update function to finite play prefixes in the usual way: $\text{Upd}^+(v_I) = m_I$ and $\text{Upd}^+(\pi v) = \text{Upd}(\text{Upd}^+(\pi), v)$ for play prefixes $\pi \in V^+$ and $v \in V$. A next-move function $\text{Nxt}: V_i \times M \rightarrow V$ for Player i has to satisfy $(v, \text{Nxt}(v, m)) \in E$ for all $v \in V_i$, $m \in M$. Each pair of a memory structure \mathcal{M} and a next-move function Nxt induces a strategy σ for Player i with memory \mathcal{M} via $\sigma(v_0 \dots v_j) = \text{Nxt}(v_j, \text{Upd}^+(v_0 \dots v_j))$. A strategy is called *finite-state* if it can be implemented by a memory structure. We define $|\mathcal{M}| = |M|$. In a slight abuse of notation, the size $|\sigma|$ of a finite-state strategy is the size of a memory structure implementing it.

An arena $\mathcal{A} = (V, V_0, V_1, E, v_I)$ together with a memory structure $\mathcal{M} = (M, m_I, \text{Upd})$ for \mathcal{A} induce the extended arena

$$\mathcal{A} \times \mathcal{M} = (V \times M, V_0 \times M, V_1 \times M, E', (v_I, m_I)) ,$$

where E' is defined via $((v, m), (v', m')) \in E'$ if and only if $(v, v') \in E$ and $\text{Upd}(m, v') = m'$. Every play $\rho = v_0 v_1 v_2 \dots$ in \mathcal{A} has a unique extended play $\text{ext}_{\mathcal{M}}(\rho) = (v_0, m_0)(v_1, m_1)(v_2, m_2) \dots$ in $\mathcal{A} \times \mathcal{M}$ defined by $m_0 = m_I$ and $m_{j+1} = \text{Upd}(m_j, v_{j+1})$, i.e., $m_j = \text{Upd}^+(v_0 \dots v_j)$. We omit the index \mathcal{M} if it is clear from the context. The extension of a finite play prefix in \mathcal{A} is defined analogously.

Let \mathcal{A} be an arena, let $\mathcal{M}_1 = (M_1, m_I^1, \text{Upd}_1)$ be a memory structure for \mathcal{A} , and let $\mathcal{M}_2 = (M_2, m_I^2, \text{Upd}_2)$ be a memory structure for $\mathcal{A} \times \mathcal{M}_1$. We define $\mathcal{M}_1 \times \mathcal{M}_2 = (M_1 \times M_2, (m_I^1, m_I^2), \text{Upd})$, where $\text{Upd}((m_1, m_2), v) = (m'_1, m'_2)$ if $\text{Upd}_1(m_1, v) = m'_1$ and $\text{Upd}_2(m_2, (v, m'_1)) = m'_2$. Via a straightforward induction and in a slight abuse of notation we obtain

$$\text{Upd}_{\mathcal{M}_1 \times \mathcal{M}_2}^+(\pi) = (\text{Upd}_{\mathcal{M}_1}^+(\pi), \text{Upd}_{\mathcal{M}_2}^+(\text{ext}_{\mathcal{M}_1}(\pi)))$$

for all finite play prefixes π , where we do not distinguish between the terms (v, m_1, m_2) , $((v, m_1), m_2)$, and $(v, (m_1, m_2))$.

2.1. Qualitative Games

A *qualitative game* $\mathcal{G} = (\mathcal{A}, \text{Win})$ consists of an arena \mathcal{A} with vertex set V and a set $\text{Win} \subseteq (V')^\omega$ of winning plays for Player 0, for some superset

$V' \supseteq V$. We call Win the winning condition of \mathcal{G} . In, e.g., a request-response game as described in Section 1, the winning condition contains the plays ρ in which for every visit to a vertex denoting a request there is a subsequent visit to a vertex denoting a response. The set of winning plays for Player 1 is $V^\omega \setminus \text{Win}$. As our definition of games is very general, the infinite object Win may not be finitely describable. If it is, however, we slightly abuse notation and define $|\mathcal{G}|$ as the sum of $|\mathcal{A}|$ and the size of a description of Win. In, e.g., the case of a request-response game, the set Win can be described via two sets $Q, P \subseteq V$, where Q and P denote the vertices representing requests and responses, respectively. We omit, however, a general definition of the term “description of Win”, as it is irrelevant for this work.

A strategy σ for Player i is a *winning strategy* for her in $\mathcal{G} = (\mathcal{A}, \text{Win})$ if all plays consistent with σ are winning for her. If Player i has a winning strategy, then we say she wins \mathcal{G} . *Solving* a game amounts to determining its winner, if one exists. A game is *determined* if one player has a winning strategy.

2.2. Quantitative Games

Quantitative games extend the classical model of qualitative games. In a quantitative game, plays are not partitioned into winning and losing plays, but rather they are assigned some measure of quality. We keep this definition very general in order to encompass many of the already existing models. In Section 4, we give concrete examples of such games and show how to solve them optimally.

A *quantitative game* $\mathcal{G} = (\mathcal{A}, \text{Cost})$ consists of an arena \mathcal{A} with vertex set V and a cost-function $\text{Cost}: (V')^\omega \rightarrow \mathbb{N}_\infty$ for plays where $V' \supseteq V$ is again some superset of V . Similarly to the winning condition in the qualitative case, Cost is, in general, an infinite object. If it is finitely describable, we, again slightly abusively, define the *size* $|\mathcal{G}|$ of \mathcal{G} as the sum of $|\mathcal{A}|$ and the size of a description of Cost. A cost function may, e.g., be described via a finite Mealy machine [24] that processes the play and outputs a natural number on each step. The cost of a play could then be defined as the supremum, infimum, or average of the resulting sequence of numbers. In such a case, a description of the cost function would consist of the Mealy machine, while the size of the description of the function could be defined as the sum of number of states of the machine and the length of the largest integer that is output by the machine.

A play ρ in \mathcal{A} is *winning* for Player 0 in \mathcal{G} if $\text{Cost}(\rho) < \infty$. Winning strategies, the winner of a game, and solving a game are defined as in the qualitative case. In order to simplify the presentation, we only consider the case in which Player 0 aims to minimize the cost of a play. All concepts in this work can, however, be easily adapted to the dual case in which Player 0 aims to maximize the cost of a play.

We extend the cost-function over plays to strategies by defining $\text{Cost}(\sigma) = \sup_{\rho} \text{Cost}(\rho)$ and $\text{Cost}(\tau) = \inf_{\rho} \text{Cost}(\rho)$, where ρ ranges over the plays consistent with the strategy σ for Player 0 and over the plays consistent with the strategy τ for Player 1, respectively. Moreover, we say that a strategy σ for Player 0 is *optimal* if its cost is minimal among all strategies for her. Dually, a strategy τ for Player 1 is optimal for him if its cost is maximal among all strategies for him.

For every strategy σ for Player 0, $\text{Cost}(\sigma) < \infty$ implies that σ is winning for Player 0. However, the converse does not hold true: Each play consistent with some strategy σ may have finite cost, while for every $n \in \mathbb{N}$ there exists a play ρ consistent with σ with $\text{Cost}(\rho) \geq n$. In contrast, a strategy τ for Player 1 has $\text{Cost}(\tau) = \infty$, if and only if τ is winning for him.

We say that Player 0 wins \mathcal{G} with respect to b if she has a strategy σ with $\text{Cost}(\sigma) \leq b$. Dually, if Player 1 has a strategy τ with $\text{Cost}(\tau) > b$, then we say that he wins \mathcal{G} with respect to b . Solving a quantitative game \mathcal{G} with respect to b amounts to deciding whether or not Player 0 wins \mathcal{G} with respect to b . We call this decision problem the *b-threshold problem* of \mathcal{G} and omit the b if it is clear from the context.

If Player 0 has a strategy σ with $\text{Cost}(\sigma) \leq b$, then for all strategies τ for Player 1 we have $\text{Cost}(\tau) \leq b$. Dually, if Player 1 has a strategy τ with $\text{Cost}(\tau) > b$, then for all strategies σ for Player 0 we have $\text{Cost}(\sigma) > b$. We say that a quantitative game is *determined* if for each $b \in \mathbb{N}$, either Player 0 has a strategy with cost at most b , or Player 1 has a strategy with cost strictly greater than b .

We say that $b \in \mathbb{N}$ is a *cap* of a quantitative game \mathcal{G} if Player 0 winning \mathcal{G} implies that she has a strategy with cost at most b . A cap b for a game \mathcal{G} is *tight* if it is minimal.

3. Quantitative Reductions

Before defining quantitative reductions, we first recall the definition of qualitative ones. To this end, let $\mathcal{G} = (\mathcal{A}, \text{Win})$ and $\mathcal{G}' = (\mathcal{A}', \text{Win}')$ be qual-

itative games. We say that \mathcal{G} is reducible to \mathcal{G}' via the memory structure \mathcal{M} for \mathcal{A} if $\mathcal{A}' = \mathcal{A} \times \mathcal{M}$ and if $\rho \in \text{Win}$ if and only if $\text{ext}(\rho) \in \text{Win}'$. Then, Player 0 wins \mathcal{G} if and only if she wins \mathcal{G}' . Moreover, if σ' is a winning strategy for Player 0 in \mathcal{G}' that is implemented by \mathcal{M}' , then a winning strategy for her in \mathcal{G} is implemented by $\mathcal{M} \times \mathcal{M}'$.

We now define quantitative reductions as an analogous technique for the study of quantitative games and show that they exhibit the same properties as qualitative reductions. Intuitively, given two quantitative games \mathcal{G} and \mathcal{G}' , we aim to say that \mathcal{G} is reducible to \mathcal{G}' if plays in one game can be translated into plays in the other, retaining their cost along the transformation. In fact, two such associated plays do not need to carry identical cost, but it suffices that the order on plays induced by their cost is retained.

To capture such order-retaining transformations of cost functions, we introduce b -correction functions. Let $b \in \mathbb{N}_\infty$. A function $f: \mathbb{N}_\infty \rightarrow \mathbb{N}_\infty$ is a b -correction function if

- for all $b'_1 < b'_2 < b$ we have $f(b'_1) < f(b'_2)$,
- for all $b' < b$ we have $f(b') < f(b)$, and
- for all $b' \geq b$ we have $f(b') \geq f(b)$.

Thus, intuitively, a b -correction function is strictly monotonic up to, but not including, b , and $f(b)$ is a lower bound for all values $f(b')$ with $b' \geq b$.

For $b = \infty$ these requirements degenerate to demanding that f is strictly monotonic, which in turn implies $f(\infty) = \infty$ and $f(b) \neq \infty$ for all $b \neq \infty$. Dually, if $b = 0$, we only require that $f(0)$ bounds the values of $f(b)$ from below. As an example, for each $b \in \mathbb{N}_\infty$ we define the function cap_b as follows:

$$\text{cap}_b(b') = \begin{cases} \min\{b, b'\} & \text{if } b' \neq \infty \text{ and} \\ \infty & \text{otherwise} \end{cases} .$$

Then, the function cap_b is a b -correction function.

Leveraging the notion of b -correction functions, we are now able to define quantitative reductions. Let $\mathcal{G} = (\mathcal{A}, \text{Cost})$ and $\mathcal{G}' = (\mathcal{A}', \text{Cost}')$ be quantitative games, let \mathcal{M} be some memory structure for \mathcal{A} , let $b \in \mathbb{N}_\infty$, and let $f: \mathbb{N}_\infty \rightarrow \mathbb{N}_\infty$ be some function. We say that \mathcal{G} is b -reducible to \mathcal{G}' via \mathcal{M} and f if

- $\mathcal{A}' = \mathcal{A} \times \mathcal{M}$,

- f is a b -correction function,
- $\text{Cost}'(\text{ext}(\rho)) = f(\text{Cost}(\rho))$ for all plays ρ of \mathcal{A} with $\text{Cost}(\rho) < b$, and
- $\text{Cost}'(\text{ext}(\rho)) \geq f(b)$ for all plays ρ of \mathcal{A} with $\text{Cost}(\rho) \geq b$.

We write $\mathcal{G} \leq_{\mathcal{M},f}^b \mathcal{G}'$ in this case. Moreover, we use cap_b as a “default” function: If $f = \text{cap}_b$, we omit stating f explicitly and write $\mathcal{G} \leq_{\mathcal{M}}^b \mathcal{G}'$. The penultimate condition implies that for each play $\text{ext}(\rho)$ in \mathcal{A}' with $\text{Cost}'(\text{ext}(\rho)) \leq f(b)$ there exists some b' such that $\text{Cost}'(\text{ext}(\rho)) = f(b')$. Clearly, quantitative reductions are downward-closed with respect to the parameter b : If $\mathcal{G} \leq_{\mathcal{M},f}^b \mathcal{G}'$ for some $b \in \mathbb{N}_\infty$, then, for all $b' \leq b$, we have $\mathcal{G} \leq_{\mathcal{M},f}^{b'} \mathcal{G}'$.

Qualitative reductions retain whether or not a play is winning. In quantitative games, the notion of winning is refined to the notion of cost of a play. Hence, our next aim is to show that quantitative reductions indeed retain the costs of strategies. To this end, we first demonstrate that correction functions tie the cost of plays in \mathcal{G}' to that of plays in \mathcal{G} .

Lemma 1. *Let \mathcal{G} and \mathcal{G}' be quantitative games such that $\mathcal{G} \leq_{\mathcal{M},f}^b \mathcal{G}'$, for some $b \in \mathbb{N}_\infty$, some memory structure \mathcal{M} , and some b -correction function f . All of the following hold true for all $b' \in \mathbb{N}$ and all plays ρ in \mathcal{G} :*

1. *If $b' < b$ and $\text{Cost}'(\text{ext}(\rho)) < f(b')$, then $\text{Cost}(\rho) < b'$.*
2. *If $b' < b$ and $\text{Cost}'(\text{ext}(\rho)) = f(b')$, then $\text{Cost}(\rho) = b'$.*
3. *If $b' < b$ and $\text{Cost}'(\text{ext}(\rho)) > f(b')$, then $\text{Cost}(\rho) > b'$.*
4. *If $\text{Cost}'(\text{ext}(\rho)) \geq f(b)$, then $\text{Cost}(\rho) \geq b$.*

Proof. 1) Let $b' < b$ and let ρ be such that $\text{Cost}'(\text{ext}(\rho)) < f(b')$. Towards a contradiction assume $\text{Cost}(\rho) = b'' \geq b'$. We have $f(b'') = f(\text{Cost}(\rho)) = \text{Cost}'(\text{ext}(\rho))$. If $b'' < b$, then we obtain $f(b') \leq f(b'')$, which implies $f(b') \leq \text{Cost}'(\text{ext}(\rho))$, contradicting $\text{Cost}'(\text{ext}(\rho)) < f(b')$. If, however, $b'' \geq b$, then $\text{Cost}'(\text{ext}(\rho)) = f(b'') \geq f(b) > f(b')$, again contradicting $\text{Cost}'(\text{ext}(\rho)) < f(b')$.

2) Let $b' < b$ and let ρ be such that $\text{Cost}'(\text{ext}(\rho)) = f(b')$. Towards a contradiction assume $\text{Cost}(\rho) = b'' \neq b'$. We again have $f(b'') = \text{Cost}'(\text{ext}(\rho))$. First assume $b'' < b'$. Then we have $b'' < b' < b$, which implies $f(b'') < f(b')$, contradicting $\text{Cost}'(\text{ext}(\rho)) = f(b')$. If $b' < b'' < b$, we obtain the contradiction $\text{Cost}'(\text{ext}(\rho)) > f(b')$ analogously. Finally, if $b \leq b''$, then

$\text{Cost}'(\text{ext}(\rho)) = f(b'') \geq f(b) > f(b')$, which again contradicts $\text{Cost}'(\text{ext}(\rho)) = f(b')$.

3) Let $b' < b$ and let ρ be such that $\text{Cost}'(\text{ext}(\rho)) > f(b')$. Towards a contradiction, assume $\text{Cost}(\rho) \leq b'$. We then obtain $f(\text{Cost}(\rho)) \leq f(b')$ due to $b' < b$ and due to f being a b -correction function. Furthermore, due to $\text{Cost}(\rho) \leq b'$, we obtain $\text{Cost}(\rho) < b$. Hence, we have $f(\text{Cost}(\rho)) = \text{Cost}'(\text{ext}(\rho))$ due to the third property from the definition of b -reducibility. Since we furthermore have $f(\text{Cost}(\rho)) \leq f(b')$ as argued above. This, in turn, directly implies $\text{Cost}'(\text{ext}(\rho)) \leq f(b')$, which contradicts the assumption $\text{Cost}'(\text{ext}(\rho)) > f(b')$.

4) Let ρ be such that $\text{Cost}'(\text{ext}(\rho)) \geq f(b)$. Towards a contradiction assume $\text{Cost}(\rho) = b' < b$. We again have $f(b') = \text{Cost}'(\text{ext}(\rho))$. However, we obtain $f(b') < f(b)$ due to f being a b -correction function. This contradicts $\text{Cost}'(\text{ext}(\rho)) = f(b') \geq f(b)$. \square

These properties of correction functions when used in quantitative reductions enable us to state and prove the main result of this section, which establishes quantitative reductions as the quantitative counterpart of qualitative reductions: If $\mathcal{G} \leq_{\mathcal{M},f}^{b+1} \mathcal{G}'$, then all plays of cost at most b in \mathcal{G} are “tracked” precisely in \mathcal{G}' . Hence, as long as the cost of a strategy in \mathcal{G} is at most b , it is possible to construct a strategy in \mathcal{G}' with cost at most $f(b)$. This holds true for both players.

If a strategy has cost greater than b , however, we do not have a direct correspondence between costs of plays in \mathcal{G} and \mathcal{G}' anymore. If, however, b additionally is a cap of \mathcal{G} , and if \mathcal{G} is determined, then we can still show that Player 1 has a strategy of infinite cost in \mathcal{G} if he has a strategy of cost greater than $f(b)$ in \mathcal{G}' .

Theorem 2. *Let \mathcal{G} and \mathcal{G}' be determined quantitative games with $\mathcal{G} \leq_{\mathcal{M},f}^{b+1} \mathcal{G}'$ for some $b, \mathcal{M},$ and f , where $b \in \mathbb{N}$ is a cap of \mathcal{G} .*

1. *Let $b' < b + 1$. Player i has a strategy σ' in \mathcal{G}' with $\text{Cost}'(\sigma') = f(b')$ if and only if they have a strategy σ in \mathcal{G} with $\text{Cost}(\sigma) = b'$.*
2. *If Player 1 has a strategy τ' in \mathcal{G}' with $\text{Cost}'(\tau') \geq f(b + 1)$, then he has a strategy τ in \mathcal{G} with $\text{Cost}(\tau) = \infty$.*

Proof. 1) We first show the direction from left to right. To this end, let σ' be a strategy for Player i in \mathcal{G}' with $\text{Cost}'(\sigma') = f(b')$ for some $b' \leq b$. We define the strategy σ for Player i in \mathcal{G} for all play prefixes π ending in a vertex in V_i

via $\sigma(\pi) = v$ if $\sigma'(\text{ext}(\pi)) = (v, m)$ for some $m \in M$ and claim $\text{Cost}(\sigma) = b'$. To this end, we first show $\text{Cost}(\sigma) \leq b'$ for the case $i = 0$ and $\text{Cost}(\sigma) \geq b'$ for the case $i = 1$.

Let ρ be an infinite play consistent with σ . A straightforward induction shows that $\rho' = \text{ext}(\rho)$ is consistent with σ' . If $i = 0$, i.e., if σ is a strategy for Player 0, then $\text{Cost}'(\rho') = \text{Cost}'(\text{ext}(\rho)) \leq \text{Cost}'(\sigma') = f(b')$. Since we furthermore have $b' < b + 1$ by assumption and since f is a $b + 1$ -correction function, we obtain $\text{Cost}(\rho) \leq b'$, due to Lemma 1.1 and Lemma 1.2. Since we picked ρ arbitrarily from the plays consistent with σ , this in turn implies $\text{Cost}(\sigma) \leq b'$. If, however $i = 1$, we directly obtain $\text{Cost}'(\text{ext}(\rho)) \geq f(b')$ due to $\text{Cost}(\sigma) \geq f(b')$. Hence, we furthermore obtain $\text{Cost}(\rho) \geq b'$ due to Lemma 1.2 and Lemma 1.3. Since we again picked ρ arbitrarily from the plays consistent with σ , this in turn implies $\text{Cost}(\sigma) \geq b'$. It remains to show $\text{Cost}(\sigma) \geq b'$ for the case $i = 0$ and $\text{Cost}(\sigma) \leq b'$ for the case $i = 1$.

First, since $b' < b + 1 < \infty$, we obtain $f(b') < \infty$: If $f(b') = \infty$, we obtain $f(b' + 1) = \infty$, which contradicts strict monotonicity of f up to and including $b + 1 \geq b' + 1$. Since $\text{Cost}'(\sigma') = f(b') < \infty$, there exists a play ρ' that is consistent with σ' such that $\text{Cost}'(\rho') = f(b')$.

Further, let ρ be the unique play such that $\text{ext}(\rho) = \rho'$. By induction, we obtain that ρ is consistent with σ . Additionally, we have $\text{Cost}(\rho) = b'$ due to $\text{Cost}'(\rho') = \text{Cost}'(\text{ext}(\rho)) = f(b')$ and Lemma 1.2. Hence, $\text{Cost}(\sigma) \geq b'$ if $i = 0$, and $\text{Cost}(\sigma) \leq b'$ if $i = 1$, which concludes the direction from left to right.

In order to show the inverse direction of the statement, let σ be a strategy in \mathcal{G} with $\text{Cost}(\sigma) = b' < b + 1$. We define the strategy σ' for Player i in \mathcal{G}' for all play prefixes $\text{ext}(\pi) = (v_0, m_0) \cdots (v_j, m_j)$ ending in a vertex in $V_i \times M$ as $\sigma'(\text{ext}(\pi)) = (v, \text{Upd}(m_j, v))$ if $\sigma(\pi) = v$ and claim $\text{Cost}'(\sigma') = f(b')$.

Let $\text{ext}(\rho)$ be a play consistent with σ' . A straightforward induction yields that ρ is consistent with σ , hence, if $i = 0$, then $\text{Cost}(\rho) \leq b'$ and thus $\text{Cost}'(\text{ext}(\rho)) \leq f(b')$ due to $b' < b + 1$. Dually, if $i = 1$, then $\text{Cost}(\rho) \geq b'$ and $\text{Cost}'(\text{ext}(\rho)) \geq f(b')$. Hence, we obtain $\text{Cost}'(\sigma') \leq f(b')$ if $i = 0$ as well as $\text{Cost}'(\sigma') \geq f(b')$ if $i = 1$. It remains to show $\text{Cost}'(\sigma') \geq f(b')$ and $\text{Cost}'(\sigma') \leq f(b')$ in the former and latter case, respectively.

Now let ρ be a play consistent with σ such that $\text{Cost}(\rho) = b'$. Since $b' < \infty$, such a play exists. Via another straightforward induction we obtain that $\text{ext}(\rho)$ is consistent with σ' . As $\text{Cost}'(\text{ext}(\rho)) = f(b')$, we furthermore obtain $\text{Cost}'(\sigma') \geq f(b')$ if $i = 0$ and $\text{Cost}'(\sigma') \leq f(b')$ if $i = 1$, which concludes the proof of this statement.

2) Let τ' be a strategy for Player 1 in \mathcal{G}' with $\text{Cost}'(\tau') \geq f(b+1)$. We define the strategy τ for Player 1 in \mathcal{G} via $\tau(\pi) = v$ if $\tau'(\text{ext}(\pi)) = (v, m)$ for all play prefixes π in \mathcal{G} . Let ρ be a play consistent with τ and define $\rho' = \text{ext}(\rho)$. A straightforward induction yields that ρ' is consistent with τ' . Since $\text{Cost}'(\tau') \geq f(b+1)$, we obtain $\text{Cost}'(\rho') \geq f(b+1)$. Then, we obtain $\text{Cost}(\rho) \geq b+1$ due to Lemma 1.4. Since we picked ρ arbitrarily from the plays consistent with τ , we directly obtain $\text{Cost}(\tau) \geq b+1$. Since b is a cap of \mathcal{G} and due to determinacy of \mathcal{G} , this implies that there exists a strategy τ'' for Player 1 in \mathcal{G} such that $\text{Cost}(\tau'') = \infty$. \square

We proved Theorem 2 by constructing optimal strategies for Player 0 in \mathcal{G} from optimal strategies for her in \mathcal{G}' . These strategies use the set of all play prefixes of \mathcal{G}' as memory states and may thus be of infinite size. If Player 0 can achieve a certain cost in \mathcal{G}' using a finite-state strategy, however, then she can achieve the corresponding cost in \mathcal{G} with a finite-state strategy as well.

Theorem 3. *Let \mathcal{G} and \mathcal{G}' be quantitative games such that $\mathcal{G} \leq_{\mathcal{M}_1, f}^b \mathcal{G}'$ for some b , \mathcal{M}_1 , and f and let either $b' < b$ or $b' = b = \infty$. If Player i has a finite-state strategy σ' with $\text{Cost}'(\sigma') = f(b')$ in \mathcal{G}' that is implemented by \mathcal{M}_2 , then she has a finite-state strategy σ with $\text{Cost}(\sigma) = b'$ in \mathcal{G} that is implemented by $\mathcal{M}_1 \times \mathcal{M}_2$.*

Proof. Let $\mathcal{G} = (\mathcal{A}, \text{Cost})$, $\mathcal{G}' = (\mathcal{A}', \text{Cost}')$, $\mathcal{M}_1 = (M_1, m_1^1, \text{Upd}_1)$, and $\mathcal{M}_2 = (M_2, m_2^2, \text{Upd}_2)$ such that σ' is implemented by \mathcal{M}_2 with the next-move function $\text{Nxt}' : (V \times M_1) \times M_2 \rightarrow (V \times M_1)$. We define $\text{Nxt}(v, (m_1, m_2)) = v^*$ if $\text{Nxt}'((v, m_1), m_2) = (v^*, \text{Upd}_1(m_1, v^*))$. We moreover define σ as the strategy that is implemented by $\mathcal{M}_1 \times \mathcal{M}_2$ with the next-move function Nxt .

Let $\rho = v_0 v_1 v_2 \dots$ be a play consistent with σ , let

$$\text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho) = (v_0, m_1^0, m_2^0)(v_1, m_1^1, m_2^1)(v_2, m_1^2, m_2^2) \dots$$

be its extension with respect to $\mathcal{M}_1 \times \mathcal{M}_2$, and let $j \in \mathbb{N}$ be such that $v_j \in V_i$. We obtain $v_{j+1} = \sigma(v_0 \dots v_j) = \text{Nxt}(v_j, (m_1^j, m_2^j))$. Due to the definition of Nxt , this implies $\text{Nxt}'((v_j, m_1^j), m_2^j) = (v_{j+1}, m_1^{j+1})$, where $m_1^{j+1} = \text{Upd}_1(m_1^j, v_{j+1})$ due to the construction of $\mathcal{A} \times \mathcal{M}_1$. Hence, $\text{ext}_{\mathcal{M}_1}(\rho)$ is consistent with σ' , i.e., $\text{Cost}'(\text{ext}_{\mathcal{M}_1}(\rho)) \leq f(b')$, which in turn implies $\text{Cost}(\rho) \leq b'$ for $i = 0$ due to Lemma 1.1 and Lemma 1.2, and $\text{Cost}'(\text{ext}_{\mathcal{M}_1}(\rho)) \geq f(b')$ and $\text{Cost}(\rho) \geq b'$ for $i = 1$.

Due to similar reasoning, for each play $\text{ext}_{\mathcal{M}_1}(\rho)$ consistent with σ' , the play ρ is consistent with σ . If $i = 1$ or $b' < \infty$, this concludes the proof. If, however, $i = 0$ and $b' = \infty$, then we furthermore obtain that $b = \infty$ and that f is a strictly monotonic function with $f(\infty) = \infty$. Hence, if there exists a play $\text{ext}_{\mathcal{M}_1}(\rho)$ consistent with σ' with $\text{Cost}'(\text{ext}_{\mathcal{M}_1}(\rho)) = \infty$, then $\text{Cost}(\rho) = \infty$ and, hence, $\text{Cost}(\sigma) = \infty$. If, however, the costs of the plays consistent with σ' diverges, then the cost of the plays consistent with σ diverges as well and we obtain $\text{Cost}(\sigma) = \infty$. \square

Theorem 2 and Theorem 3 show that quantitative reductions indeed exhibit properties analogous to those of qualitative reductions in the quantitative setting. Recall that in addition to retaining winning plays and to allowing the implementation of finite-state strategies, qualitative reductions furthermore are transitive: If \mathcal{G} , \mathcal{G}' , and \mathcal{G}'' are qualitative games such that $\mathcal{G} \leq \mathcal{G}'$, and such that $\mathcal{G}' \leq \mathcal{G}''$, then $\mathcal{G} \leq \mathcal{G}''$. We now show that quantitative reductions are transitive as well.

Theorem 4. *Let $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ be quantitative games such that $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$ and $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$ for some $b_1, b_2 \in \mathbb{N}_\infty$, some memory structures $\mathcal{M}_1, \mathcal{M}_2$, and some b_1 - and b_2 -correction functions f_1 and f_2 , respectively.*

Then, we have $\mathcal{G}_1 \leq_{\mathcal{M}, f}^b \mathcal{G}_3$, where $\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2$, $f = f_2 \circ f_1$, and $b = b_1$ if $b_2 \geq f_1(b_1)$ and $b = \max\{b' \mid f_1(b') \leq b_2\}$ otherwise.

Proof. For each $j \in \{1, 2, 3\}$, let $\mathcal{G}_j = (\mathcal{A}_j, \text{Cost}_j)$. Recall that in order to show $\mathcal{G}_1 \leq_{\mathcal{M}, f}^b \mathcal{G}_3$ we have to show that

- $\mathcal{A}_3 = \mathcal{A}_1 \times \mathcal{M}_1 \times \mathcal{M}_2$,
- $f_2 \circ f_1$ is a b -reduction function,
- for all plays ρ of \mathcal{A}_1 with $\text{Cost}_1(\rho) < b$ we have $\text{Cost}_3(\text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho)) = (f_2 \circ f_1)(\text{Cost}_1(\rho))$, and
- for all plays ρ of \mathcal{A}_1 with $\text{Cost}_1(\rho) \geq b$ we have $\text{Cost}_3(\text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho)) \geq (f_2 \circ f_1)(b)$.

We show all these items individually.

Clearly, we have

$$\mathcal{A}_3 = \mathcal{A}_2 \times \mathcal{M}_2 = \mathcal{A}_1 \times \mathcal{M}_1 \times \mathcal{M}_2 .$$

We now show that $f_2 \circ f_1$ is a b -correction function, for b defined as in the statement of the theorem. Recall that to this end we have to show that

- for all $x < x' < b$ we have $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(x')$,
- for all $x < b$ we have $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(b)$, and
- for all $x \geq b$ we have $(f_2 \circ f_1)(x) \geq (f_2 \circ f_1)(b)$.

Furthermore recall that we defined

$$b = \begin{cases} b_1 & \text{if } b_2 \geq f_1(b_1) \text{ and} \\ \max\{b' \mid f_1(b') \leq b_2\} & \text{otherwise .} \end{cases}$$

We treat both cases of this definition separately.

First, assume $b_2 \geq f_1(b_1)$. In this case we have $b = b_1$. We show the three items of the definition of a b -correction function independently.

- First, pick x and x' such that $x < x' < b = b_1$. We show $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(x')$. Since f_1 is a b_1 -correction function, we obtain $f_1(x) < f_1(x')$, $f_1(x) < f_1(b_1)$, and $f_1(x') < f_1(b_1)$. Since, furthermore, f_2 is a b_2 -correction function and as $f_1(b_1) \leq b_2$ by assumption, we moreover obtain $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(x')$.
- Now pick some x such that $x < b = b_1$. We show $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(b)$. Since f_1 is a b_1 -correction function, we directly obtain $f_1(x) < f_1(b_1)$. Since $f_1(b_1) \leq b_2$ by assumption and since f_2 is a b_2 -correction function, this directly yields $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(b_1) = (f_2 \circ f_1)(b)$.
- Finally, pick some x such that $x \geq b = b_1$. Then $f_1(x) \geq f_1(b_1)$, since, again, f_1 is a b_1 -correction function. If $f_1(x) < b_2$, then $(f_2 \circ f_1)(x) \geq (f_2 \circ f_1)(b_1)$. If, however, $f_1(x) \geq b_2$, then $(f_2 \circ f_1)(x) \geq f_2(b_2) \geq (f_2 \circ f_1)(b_1)$, where the latter inequality follows from the assumption $b_2 \geq f_1(b_1)$. This concludes this part of the proof.

Now assume $b_2 < f_1(b_1)$ and let b be maximal such that $f_1(b) \leq b_2$. We first argue that we have $b \leq b_1$ in this case. Towards a contradiction assume $b_1 < b$. This implies $f_1(b_1) \leq f_1(b)$ due to f_1 being a b_1 -correction function. However, we have $f_1(b) \leq b_2 < f_1(b_1)$, where the former inequality results from the definition of b , while the latter one is due to our initial assumption. This directly contradicts $f_1(b_1) \leq f_1(b)$, hence we obtain $b \leq b_1$.

We again show that $f_2 \circ f_1$ is a b -correction function by showing the three items of the definition independently.

- First, pick x and x' such that $x < x' < b$. Since $b \leq b_1$ we obtain $f_1(x) < f_1(x') < f_1(b)$ due to f_1 being a b_1 -correction function. As $f_1(b) \leq b_2$ due to the definition of b , we directly obtain $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(x')$.
- Now, pick x such that $x < b$. We obtain $(f_2 \circ f_1)(x) < (f_2 \circ f_1)(b)$ via reasoning analogous to the previous case.
- Finally, pick x such that $b \leq x$. We show $(f_2 \circ f_1)(b) \leq (f_2 \circ f_1)(x)$. To this end, we first observe that we have $f_1(b) \leq f_1(x)$ by leveraging different properties of f_1 being a b_1 -correction function, depending on whether we have $b \leq x < b_1$ or $b \leq b_1 \leq x$. The case $b_1 < b \leq x$ is excluded due to $b \leq b_1$ as argued above. We furthermore obtain $(f_2 \circ f_1)(b) \leq (f_2 \circ f_1)(x)$ by similar reasoning, using different properties of f_2 being a b_2 -correction function, depending on whether we have $f_1(b) \leq b_2 \leq f_1(x)$ or $f_1(b) \leq f_1(x) < f_1(b_2)$. Again, the case $b_2 < f_1(b) \leq f_1(x)$ is excluded due to the definition of b .

Thus, we have shown that $f_2 \circ f_1$ is indeed a b -correction function. It remains to prove the latter two conditions from the definition of $\mathcal{G}_1 \leq_{\mathcal{M},f}^b \mathcal{G}_3$. We first aim to show that for all plays ρ of \mathcal{A}_1 with $\text{Cost}_1(\rho) < b$ we have $\text{Cost}_3(\text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho)) = (f_2 \circ f_1)(\text{Cost}_1(\rho))$. To this end, let ρ be a play of \mathcal{A}_1 with $\text{Cost}_1(\rho) < b$. For the sake of consistency and readability, we define $\rho_1 = \rho$, $\rho_2 = \text{ext}_{\mathcal{M}_1}(\rho_1)$, and $\rho_3 = \text{ext}_{\mathcal{M}_2}(\rho_2) = \text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho_1)$. We again treat both cases of the definition of b separately.

We again first consider the case that $b_2 \geq f_1(b_1)$. We then directly obtain $b = b_1$ due to the definition of b . This yields $\text{Cost}_1(\rho_1) < b_1$, which in turn implies $f_1(\text{Cost}_1(\rho_1)) = \text{Cost}_2(\rho_2)$ due to $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$. Furthermore, since $\text{Cost}_1(\rho_1) < b_1$ and since f_1 is a b_1 -correction function we have $f_1(\text{Cost}_1(\rho_1)) < f_1(b_1)$, which directly yields $\text{Cost}_2(\rho_2) < f_1(b_1)$ via the equation above. Since $f_1(b_1) \leq b_2$, we further obtain $\text{Cost}_2(\rho_2) < b_2$. Due to $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$ this then implies $\text{Cost}_3(\rho_3) = f_2(\text{Cost}_2(\rho_2))$. By again applying $f_1(\text{Cost}_1(\rho_1)) = \text{Cost}_2(\rho_2)$, we obtain the desired result of $\text{Cost}_3(\rho_3) = (f_2 \circ f_1)(\text{Cost}_1(\rho_1))$.

Now consider the case that $b_2 < f_1(b_1)$. In this case, we have $b = \max\{b' \mid f_1(b') \leq b_2\}$ by definition of b . As argued above, we have $b \leq b_1$ in this case, which directly implies $\text{Cost}_1(\rho_1) < b_1$. Due to $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$ this yields $f_1(\text{Cost}_1(\rho_1)) = \text{Cost}_2(\rho_2)$. We also obtain $f_1(\text{Cost}_1(\rho_1)) < f_1(b)$. Since we again have $f_1(b) \leq b_2$ by definition of b , we obtain the desired result of

$\text{Cost}_3(\rho_3) = (f_2 \circ f_1)(\text{Cost}_1(\rho_1))$ via reasoning analogously to the previous case.

It remains to show the final item in the definition of b -reducibility, i.e., that we indeed have $\text{Cost}_3(\text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho)) \geq (f_2 \circ f_1)(\text{Cost}_1(\rho))$ for all plays ρ of \mathcal{A}_1 with $\text{Cost}_1(\rho) \geq b$. To this end, let ρ be a play of \mathcal{A}_1 with $\text{Cost}_1(\rho) \geq b$. For the sake of brevity and consistency we again define $\rho_1 = \rho$, $\rho_2 = \text{ext}_{\mathcal{M}_1}(\rho)$, and $\rho_3 = \text{ext}_{\mathcal{M}_2}(\rho_2) = \text{ext}_{\mathcal{M}_1 \times \mathcal{M}_2}(\rho_1)$. We again treat the two cases from the definition of b separately.

First, assume $b_2 \geq f_1(b_1)$. We directly obtain $b = b_1$ by definition of b , which in turn implies $\text{Cost}_2(\rho_2) \geq f_1(b_1)$ due to $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$. If $b_2 > \text{Cost}_2(\rho_2)$, then we directly obtain $f_2(\text{Cost}_2(\rho_2)) \geq (f_2 \circ f_1)(b_1)$ since f_2 is a b_2 -correction function. Since we have $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$, we moreover obtain $f_2(\text{Cost}_2(\rho_2)) = \text{Cost}_3(\rho_3)$, which in turn yields $\text{Cost}_3(\rho_3) \geq (f_2 \circ f_1)(b_1) = (f_2 \circ f_1)(b)$. If, however, $\text{Cost}_2(\rho_2) \geq b_2 \geq f_1(b_1)$, then we obtain $\text{Cost}_3(\rho_3) \geq f_2(b_2) \geq (f_2 \circ f_1)(b_1) = (f_2 \circ f_1)(b)$, where the former inequality is implied by $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$ while we obtain the latter due to f_2 being a b_2 -correction function. Since the final case, i.e., the case $\text{Cost}_2(\rho_2) \geq f_1(b_1) > b_2$ would contradict our initial assumption $b_2 \geq f_1(b_1)$, this concludes the first case of this proof.

Now consider the case $b_2 < f_1(b_1)$. Here, we again distinguish two sub-cases. If $\text{Cost}_1(\rho_1) \geq b_1$, then we obtain $\text{Cost}_2(\rho_2) \geq f_1(b_1) > b_2 \geq f_1(b)$. The three inequalities result from $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$, from the assumption above, and from the definition of b , respectively. We furthermore obtain $\text{Cost}_3(\rho_3) \geq f_2(b_2) \geq (f_2 \circ f_1)(b)$, where the former inequality is due to $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$ and $\text{Cost}_2(\rho_2) > b_2$, while the latter one results from f_2 being a b_2 -correction function and from $b_2 \geq f_1(b)$.

It remains to consider the case $b_1 > \text{Cost}_1(\rho_1)$. In this case, we directly obtain $\text{Cost}_2(\rho_2) = f_1(\text{Cost}_1(\rho_1)) \geq f_1(b)$. The former equality is due to $\mathcal{G}_1 \leq_{\mathcal{M}_1, f_1}^{b_1} \mathcal{G}_2$, while the latter results from the assumption $\text{Cost}_1(\rho_1) \geq b$ and the fact that f_1 is a b_1 -correction function. We again distinguish two cases based on the relation of b_2 to $f_1(b)$ and $\text{Cost}_2(\rho_2)$: Either we have $\text{Cost}_2(\rho_2) \geq b_2 \geq f_1(b)$, or we have $b_2 > \text{Cost}_2(\rho_2) \geq f_1(b)$. In the former case, we obtain $\text{Cost}_3(\rho_3) \geq f_2(b_2) \geq (f_2 \circ f_1)(b)$, where the former and latter inequality are due to $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$ and due to f_2 being a b_2 -reduction function, respectively. In the latter case, we similarly obtain $f_2(\text{Cost}_2(\rho_2)) = \text{Cost}_3(\rho_3) \geq (f_2 \circ f_1)(b)$, where the former equality again results from $\mathcal{G}_2 \leq_{\mathcal{M}_2, f_2}^{b_2} \mathcal{G}_3$, while the latter inequality is again due to f_2 being a b_2 -correction function. \square

By using quantitative reductions we are able to structure the space of quantitative games similarly to that of qualitative games. While for qualitative games, there exist direct solutions to a number of well-studied winning conditions, for quantitative games, no such direct solutions for the threshold problems exist to the best of our knowledge. Instead, the threshold problem for quantitative games is usually solved by reducing the quantitative game to a qualitative game for a fixed bound b .

Hence, there does not yet exist a “foundation” of the space of quantitative winning conditions analogous to that of the space of qualitative winning conditions, i.e., there is no canonical simple class of quantitative games that provides a natural target for quantitative reductions. In the following section, we provide such a foundation in the form of vertex-ranked games.

4. Vertex-Ranked Games

We introduce two very simple kinds of quantitative games, which we call vertex-ranked games. In such games, the cost of a play is determined solely by a qualitative winning condition and a ranking of the vertices of the arena by natural numbers. We show that solving the threshold problem for either kind of game is possible with only a polynomial overhead over solving the underlying qualitative game.

Furthermore, we show that the memory structures implementing winning strategies for either player only incur a polynomial overhead in comparison to the memory structures implementing winning strategies for the underlying conditions. Finally, we briefly discuss the optimization problem for such games, i.e., the problem of determining the minimal b such that Player 0 has a strategy of cost at most b in such a game. We argue that determining such b incurs only a polynomial overhead over solving the underlying qualitative game.

Let \mathcal{A} be an arena with vertex set V , let $\text{Win} \subseteq (V)^\omega$ be a qualitative winning condition, and let $\text{rk}: V \rightarrow \mathbb{N}$ be a ranking function on vertices. We define the quantitative vertex-ranked sup-condition

$$\text{RANK}^{\text{sup}}(\text{Win}, \text{rk}): v_0v_1v_2 \cdots \mapsto \begin{cases} \sup_{j \rightarrow \infty} \text{rk}(v_j) & \text{if } v_0v_1v_2 \cdots \in \text{Win and} \\ \infty & \text{otherwise ,} \end{cases}$$

as well as its prefix-independent version, the vertex-ranked lim sup-condition

$$\text{RANK}^{\text{lim}}(\text{Win}, \text{rk}): v_0 v_1 v_2 \cdots \mapsto \begin{cases} \limsup_{j \rightarrow \infty} \text{rk}(v_j) & \text{if } v_0 v_1 v_2 \cdots \in \text{Win} \text{ and} \\ \infty & \text{otherwise .} \end{cases}$$

A vertex-ranked sup- or lim sup-game $\mathcal{G} = (\mathcal{A}, \text{RANK}^X(\text{Win}, \text{rk}))$ with $X \in \{\text{sup}, \text{lim}\}$ consists of an arena \mathcal{A} with vertex set V , a qualitative winning condition Win , and a vertex-ranking function $\text{rk}: V \rightarrow \mathbb{N}$.

If $\mathcal{G}_X = (\mathcal{A}, \text{RANK}^X(\text{Win}, \text{rk}))$ is a vertex-ranked sup- or lim sup-game, we call the game $(\mathcal{A}, \text{Win})$ the qualitative game corresponding to \mathcal{G}_X . Moreover, if \mathcal{G}_{sup} is a vertex-ranked sup-game, we denote the vertex-ranked lim sup-game with the same arena, winning condition, and rank function by \mathcal{G}_{lim} and vice versa. In either case, we denote the corresponding qualitative game by \mathcal{G} .

Before showing how to solve vertex-ranked sup- and lim sup-games, we argue that these games allow straightforward adaptations of qualitative winning conditions to quantitative ones. This is witnessed by the following theorem.

Theorem 5. *Let $\mathcal{G} = (\mathcal{A}, \text{Win})$ and $\mathcal{G}' = (\mathcal{A}', \text{Win}')$ be qualitative games and let \mathcal{M} be a memory structure such that $\mathcal{G} \leq_{\mathcal{M}} \mathcal{G}'$. Moreover, let rk be a ranking function on vertices of \mathcal{G} and let b be the maximal rank assigned to a vertex of \mathcal{G} by rk . Then, $\mathcal{G}_{\text{sup}} \leq_{\mathcal{M}}^{b+1} \mathcal{G}'_{\text{sup}}$, where $\mathcal{G}_{\text{sup}} = (\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Win}, \text{rk}))$ and $\mathcal{G}'_{\text{sup}} = (\mathcal{A}', \text{RANK}^{\text{sup}}(\text{Win}', \text{rk}'))$ with $\text{rk}'(v, m) = \text{rk}(v)$.*

Proof. Clearly, we have $\mathcal{A}' = \mathcal{A} \times \mathcal{M}$ due to $\mathcal{G} \leq \mathcal{G}'$. Moreover, as argued above, cap_{b+1} is a $b+1$ -correction function. Hence, it remains to show that the latter two conditions in the definition of a quantitative reduction are satisfied.

To this end, let ρ be a play in \mathcal{A} . If the cost of ρ is less than $b+1$, then we obtain $\rho \in \text{Win}$ by definition of b and by definition of the vertex-ranked sup-condition. This, in turn, implies $\text{ext}(\rho) \in \text{Win}'$ due to $\mathcal{G} \leq \mathcal{G}'$. Hence, we obtain

$$\begin{aligned} \text{cap}_{b+1}(\text{RANK}^{\text{sup}}(\text{Win}, \text{rk})(\rho)) &= \text{RANK}^{\text{sup}}(\text{Win}, \text{rk})(\rho) = \\ & \text{RANK}^{\text{sup}}(\text{Win}', \text{rk}')(\text{ext}(\rho)) . \end{aligned}$$

If, however, the cost of ρ is at least $b + 1$, then we have $\rho \notin \text{Win}$, again due to the definition of b , which implies both $\text{RANK}^{\text{sup}}(\text{Win}, \text{rk})(\rho) = \infty$ as well as $\text{ext}(\rho) \notin \text{Win}$ due to $\mathcal{G} \leq \mathcal{G}'$. Hence, we obtain

$$\text{RANK}^{\text{sup}}(\text{Win}', \text{rk}')(\text{ext}(\rho)) = \infty \geq \text{cap}_{b+1}(b + 1) = b + 1 \quad ,$$

which concludes the proof of the fourth condition of quantitative reductions. \square

Clearly, the above proof can be adapted in a very straightforward way to show the analogous result for the case of vertex-ranked lim sup-games.

Remark 6. *Let $\mathcal{G} = (\mathcal{A}, \text{Win})$ and $\mathcal{G}' = (\mathcal{A}', \text{Win}')$ be qualitative games and let \mathcal{M} be a memory structure such that $\mathcal{G} \leq_{\mathcal{M}} \mathcal{G}'$. Moreover, let rk be a ranking function on vertices of \mathcal{G} and let b be the maximal rank assigned to a vertex of \mathcal{G} by rk . Then, $\mathcal{G}_{\text{lim}} \leq_{\mathcal{M}}^{b+1} \mathcal{G}'_{\text{lim}}$, where $\mathcal{G}_{\text{lim}} = (\mathcal{A}, \text{RANK}^{\text{lim}}(\text{Win}, \text{rk}))$ and $\mathcal{G}'_{\text{lim}} = (\mathcal{A}', \text{RANK}^{\text{lim}}(\text{Win}', \text{rk}'))$ with $\text{rk}'(v, m) = \text{rk}(v)$.*

The remainder of this section is dedicated to providing bounds on the complexity of solving vertex-ranked games with respect to some given bound. In particular, we show that vertex-ranked sup-games can be solved with only an additive linear blowup compared to the complexity of solving the corresponding qualitative games. Vertex-ranked lim sup-games, on the other hand, can be solved while incurring only a polynomial blowup compared to solving the corresponding qualitative games.

4.1. Solving Vertex-Ranked sup-Games

We begin by observing that solving vertex-ranked sup-games is at least as hard as solving the underlying qualitative games, since the former subsumes the latter. This is due to the fact that Player 0 has a winning strategy in $(\mathcal{A}, \text{Win})$ if and only if she has a strategy with cost at most zero in $(\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Win}, \text{rk}))$, where rk is the constant function assigning zero to every vertex.

We now turn our attention to finding an upper bound for the complexity of the threshold problem for vertex-ranked sup-games. To achieve a general treatment of such games, we first introduce some notation. Let \mathfrak{G} be a class of qualitative games. We define the extension of \mathfrak{G} to vertex-ranked sup-games as

$$\mathfrak{G}_{\text{sup}}^{\text{RNK}} = \{(\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Win}, \text{rk})) \mid (\mathcal{A}, \text{Win}) \in \mathfrak{G}, \text{rk is vertex-ranking function for } \mathcal{A}\} \quad .$$

We first show that we can use a decision procedure solving games from \mathfrak{G} to solve games from $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ with respect to a given b . To this end, we remove all vertices from which Player 1 can enforce a visit to a vertex of rank greater than b and proclaim that Player 0 wins the quantitative game with respect to b if and only if she wins the qualitative game corresponding to the resulting quantitative game. To ensure that we are able to solve the resulting qualitative game, we assume some closure properties of \mathfrak{G} . To this end, we first introduce some notation.

Let \mathcal{G} be a qualitative or quantitative game with vertex set V . For each $v \in V$, we write \mathcal{G}_v to denote the game \mathcal{G} with its initial vertex replaced by v . All other components, i.e., the structure of the arena and the cost-function, remain unchanged. Let $\mathcal{A} = (V, V_0, V_1, E, v_I)$ and $\mathcal{A}' = (V', V'_0, V'_1, E', v'_I)$ be arenas. We say that \mathcal{A}' is a *sub-arena* of \mathcal{A} if $V' \subseteq V$, $V'_0 \subseteq V_0$, $V'_1 \subseteq V_1$, $E' \subseteq E$, and $v_I = v'_I$ and write $\mathcal{A}' \sqsubseteq \mathcal{A}$ in this case.

We call a class of qualitative (or quantitative) games \mathfrak{G} *proper* if

- for each $(\mathcal{A}, \text{Win})$ (or $(\mathcal{A}, \text{Cost})$) in \mathfrak{G} and each sub-arena $\mathcal{A}' \sqsubseteq \mathcal{A}$ the game $(\mathcal{A}', \text{Win}')$ (or $(\mathcal{A}', \text{Cost}')$), where Win' (or Cost') is the restriction of Win (or Cost) to plays from \mathcal{A}' , is a member of \mathfrak{G} as well, if
- for each game $\mathcal{G} \in \mathfrak{G}$ and each vertex v of \mathcal{G} we have $\mathcal{G}_v \in \mathfrak{G}$, if
- all games in \mathfrak{G} are determined, and if
- all $\mathcal{G} \in \mathfrak{G}$ are finitely representable.

Intuitively, the first condition ensures that games obtained by removing vertices or edges from games in \mathfrak{G} are members of \mathfrak{G} as well, whereas the latter three conditions are very weak technical requirements. In particular the requirement that all games included in the class must be finitely representable serves mainly to enable us to talk about the size of a game.

Using this notion of proper classes of games, we are now able to formulate the main result of this section regarding vertex-ranked sup-games.

Theorem 7. *Let \mathfrak{G} be a proper class of qualitative games \mathcal{G} that can be solved in time $t(|\mathcal{G}|)$ and space $s(|\mathcal{G}|)$, where t and s are monotonic functions.*

Then, the following problem can be solved in time $\mathcal{O}(n) + t(|\mathcal{G}|)$ and space $\mathcal{O}(n) + s(|\mathcal{G}|)$: “Given some game $\mathcal{G}_{\text{sup}} \in \mathfrak{G}_{\text{sup}}^{\text{RNK}}$ with n vertices and some bound $b \in \mathbb{N}$, does Player 0 win \mathcal{G}_{sup} with respect to b ?”

Intuitively, in order to prove Theorem 7, we show that Player 0 wins $\mathcal{G}_{\text{sup}} \in \mathfrak{G}_{\text{sup}}^{\text{RNK}}$ with respect to some bound b if and only if

- Player 1 cannot enforce a visit to vertices of rank greater than b from v_I , and if
- she is able to win the game \mathcal{G}_{sup} without visiting any vertices from which Player 1 is able to enforce a visit to a vertex of rank greater than b .

We formalize the idea of removing vertices from which one player can enforce a visit to some set of vertices by first recalling the attractor construction. Let $\mathcal{A} = (V, V_0, V_1, E, v_I)$ be an arena with n vertices and let $X \subseteq V$. We define $\text{Attr}_i(X) = \text{Attr}_i^n(X)$ inductively with $\text{Attr}_i^0(X) = X$ and

$$\begin{aligned} \text{Attr}_i^j(X) = \{ & v \in V_i \mid \exists v' \in \text{Attr}_i^{j-1}(X). (v, v') \in E \} \cup \\ & \{ v \in V_{1-i} \mid \forall (v, v') \in E. v' \in \text{Attr}_i^{j-1}(X) \} \cup \text{Attr}_i^{j-1}(X) . \end{aligned}$$

Intuitively, the i -attractor $\text{Attr}_i(X)$ is the set of all vertices from which Player i can enforce a visit to X . The set $\text{Attr}_i(X)$ can be computed in linear time in $|E|$ and Player i has a positional strategy σ such that each play starting in some vertex in $\text{Attr}_i(X)$ and consistent with σ eventually encounters some vertex from X [25]. We call σ an attractor strategy towards X .

We furthermore formalize the notion of removing attractors from arenas: Let \mathcal{A} be an arena with vertex set V , let $X \subseteq V$, and let $A = \text{Attr}_i(X)$. If $v_I \notin A$, then we define

$$\mathcal{A} \setminus A = (V \setminus A, V_0 \setminus A, V_1 \setminus A, \{(v, v') \in E \mid v \notin A \text{ and } v' \notin A\}, v_I) ,$$

which is again an arena. We lift this notation to qualitative (and quantitative) games $\mathcal{G} = (\mathcal{A}, \text{Win})$ (or $(\mathcal{A}, \text{Cost})$) by defining $\mathcal{G} \setminus A = (\mathcal{A} \setminus A, \text{Win} \cap (V \setminus A)^\omega)$ (or $(\mathcal{A} \setminus A, \text{Cost}|_{(V \setminus A)^\omega})$, where $\text{Cost}|_{(V \setminus A)^\omega}$ denotes the restriction of Cost to the domain $(V \setminus A)^\omega$). This restriction of the winning condition and the cost function to vertices of V is not strictly necessary due to our definition of $\text{Win} \subseteq (V')^\omega$ and of $\text{Cost}: (V')^\omega \rightarrow \mathbb{N}_\infty$, but it makes the resulting objects easier to reason about.

If $v_I \in A$, however, then both $\mathcal{A} \setminus A$ and $\mathcal{G} \setminus A$ are undefined. The game $\mathcal{G} \setminus A$ can be constructed in linear time and is of size at most $|\mathcal{G}|$.

As a first step towards the proof of Theorem 7, we show that vertex-ranked sup-games can be solved by using a single attractor construction and considering the qualitative game obtained by removing the resulting attractor.

Lemma 8. *Let \mathfrak{G} be a proper class of qualitative games, let $\mathcal{G}_{\text{sup}} = (\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Win}, \text{rk})) \in \mathfrak{G}_{\text{sup}}^{\text{RNK}}$ with vertex set V and initial vertex v_I , and let $b \in \mathbb{N}$.*

Player 0 has a strategy with cost at most b in \mathcal{G}_{sup} if and only if $v_I \notin A$ and if she has a winning strategy in the qualitative game $\mathcal{G}' = \mathcal{G} \setminus A$, where $\mathcal{G} = (\mathcal{A}, \text{Win})$ and $A = \text{Attr}_1(\{v \in V \mid \text{rk}(v) > b\})$.

Proof. Let $X_b = \{v \in V \mid \text{rk}(v) > b\}$. We first show the direction from right to left, i.e., that, if $v_I \notin A$ and if Player 0 wins \mathcal{G}' , say with strategy σ' , then she has a strategy of cost at most b in \mathcal{G}_{sup} . To this end, define $\mathcal{A}' = \mathcal{A} \setminus A$. Since $\mathcal{A}' \sqsubseteq \mathcal{A}$, the strategy σ' is a strategy for Player 0 in \mathcal{G}_{sup} as well, due to Player 0 being able to keep the play inside \mathcal{A}' using σ' . Hence, each play consistent with σ' in \mathcal{G}' is consistent with σ' in \mathcal{G} as well as vice versa. Let ρ be a play in \mathcal{G}_{sup} consistent with σ' . Since σ' is winning for Player 0 in \mathcal{G}' , we have $\rho \in \text{Win} \cap (V \setminus A)^\omega \subseteq \text{Win}$. Moreover, since $X_b \subseteq A$, and as ρ visits only vertices occurring in \mathcal{G}' , we obtain $\text{RANK}^{\text{sup}}(\text{Win}, \text{rk})(\rho) \leq b$ and thus $\text{Cost}(\sigma') \leq b$, which concludes this direction of the proof.

We show the other direction via contraposition: To this end, first assume $v_I \in A$ and let τ_A be an attractor strategy towards X_b for Player 1. We show that Player 0 does not have a strategy with cost at most b in \mathcal{G}_{sup} by showing that τ_A has cost exceeding b . We obtain $\text{Cost}(\tau_A) > b$ in \mathcal{G}_{sup} : By playing consistently with τ_A , Player 1 forces the play to eventually reach a vertex in X_b , i.e., a vertex v with $\text{rk}(v) > b$. Thus, $\text{Cost}(\tau_A) > b$, i.e., $\text{Cost}(\sigma) > b$ for all strategies σ of Player 0.

Now assume that Player 0 does not have a winning strategy in \mathcal{G}' . Towards a contradiction, assume that she has a strategy σ with cost at most b in \mathcal{G}_{sup} . We first observe that no play consistent with σ visits any vertex from A . Otherwise, playing consistently with his attractor strategy towards X_b from the first visit to A , Player 1 would be able to construct a play consistent with σ , but with cost greater than b . Thus, σ is a strategy for Player 0 in \mathcal{G}' and we obtain that all plays consistent with σ in \mathcal{A} are consistent with σ in \mathcal{A}' and vice versa. Since $\text{Cost}(\sigma) \leq b$, we obtain $\text{RANK}^{\text{sup}}(\text{Win}, \text{rk})(\rho) < \infty$, i.e., $\rho \in \text{Win}$ for all plays ρ consistent with σ . Thus, σ is a winning strategy for Player 0 in \mathcal{G}' , a contradiction. \square

Using this lemma, we are able to construct a decision procedure solving games from $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ using a decision procedure solving games from \mathfrak{G} .

Proof of Theorem 7. Since \mathfrak{G} is proper, $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ is proper as well. Given the vertex-ranked sup-game $\mathcal{G}_{\text{sup}} = (\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Win}, \text{rk}))$, let $X_b = \{v \in V \mid \text{rk}(v) > b\}$ and let $A = \text{Attr}_1(X_b)$. We define the decision procedure dec_{sup} deciding the given problem such that it returns false if $v_I \in A$. Otherwise, dec_{sup} returns true if and only if Player 0 wins $\mathcal{G} \setminus A$. Since $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ is proper and due to the assumption of the theorem, $\mathcal{G} \setminus A$ can be solved in time at most $t(|\mathcal{G}|)$ and space at most $s(|\mathcal{G}|)$. The procedure dec_{sup} indeed decides the given decision problem due to Lemma 8.

Since we can compute and remove the Player-1-attractor A in linear time in $|\mathcal{A}|$ [25], the decision procedure dec_{sup} indeed requires time $\mathcal{O}(|\mathcal{A}|) + t(|\mathcal{G}|)$ and space $\mathcal{O}(|\mathcal{A}|) + s(|\mathcal{G}|)$. \square

This theorem provides an upper bound on the complexity of solving vertex-ranked sup-games. Intuitively, we prove Theorem 7 by showing that, for any vertex-ranked sup-game \mathcal{G}_{sup} , a winning strategy for Player 0 in \mathcal{G} that never moves to the Player 1-attractor towards vertices of rank greater than b has cost at most b . Thus, an upper bound on the size of winning strategies for Player 0 for games from \mathfrak{G} provides an upper bound for strategies of finite cost in $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ as well. Moreover, if the decision procedure deciding \mathfrak{G} constructs winning strategies for one or both players, we can adapt the decision procedure deciding $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ to construct strategies of cost at most (greater than) b for Player 0 (Player 1) as well.

Corollary 9. *Let $\mathfrak{G}_{\text{sup}}$ be a proper class of vertex-ranked sup-games and let $\mathcal{G}_{\text{sup}} \in \mathfrak{G}_{\text{sup}}$. If σ is a finite-state winning strategy for Player i in \mathcal{G} , then Player i has a finite-state winning strategy σ_{sup} in \mathcal{G}_{sup} with $|\sigma_{\text{sup}}| \in \mathcal{O}(|\sigma|)$. Furthermore, if σ is effectively constructible, then σ_{sup} is effectively constructible.*

Finally, the procedure constructed in the proof of Theorem 7 enables us to solve the optimization problem for vertex-ranked sup-games from $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$: Recall that if Player 0 wins \mathcal{G}_{sup} with respect to some b , she wins it with respect to all $b' \geq b$ as well. Hence, using a binary search, $\log(M)$ invocations of the decision procedure from the proof of Theorem 7 suffice to determine the minimal b such that Player 0 wins \mathcal{G}_{sup} with respect to b , where M denotes the number of ranks assigned to vertices of \mathcal{G}_{sup} by its ranking function. Hence,

it is possible to determine the minimal such b in time $\mathcal{O}(\log(M)(n + t(|\mathcal{G}|)))$ and space $\mathcal{O}(M) + s(|\mathcal{G}|)$.

4.2. Solving Vertex-Ranked lim sup-Games

We now turn our attention to solving vertex-ranked lim sup-games. Solving these games is again at least as hard as solving their corresponding qualitative games, due to the same reasoning as for vertex-ranked sup-games. Thus, we again only provide upper bounds on the complexity of solving such games. To this end, given some class \mathfrak{G} of games, we define the corresponding class of vertex-ranked lim sup-games

$$\mathfrak{G}_{\text{lim}}^{\text{RNK}} = \{(\mathcal{A}, \text{RANK}^{\text{lim}}(\text{Win}, \text{rk})) \mid (\mathcal{A}, \text{Win}) \in \mathfrak{G}, \text{rk is vertex-ranking function for } \mathcal{A}\} .$$

We identify two criteria on classes of qualitative games \mathfrak{G} , each of which is sufficient for quantitative games in $\mathfrak{G}_{\text{lim}}^{\text{RNK}}$ to be solvable with respect to some given b . More precisely, we provide decision procedures for $\mathfrak{G}_{\text{lim}}^{\text{RNK}}$ for the case that

- games from \mathfrak{G} can be solved in conjunction with coBüchi-conditions, and for the case that
- the winner of a play ρ in a game from \mathfrak{G} depends only on an infinite suffix of ρ .

In order to show the former case, fix some class of games \mathfrak{G} and let $\mathcal{G}_{\text{lim}} = (\mathcal{A}, \text{RANK}^{\text{lim}}(\text{Win}, \text{rk})) \in \mathfrak{G}_{\text{lim}}^{\text{RNK}}$ be a vertex-ranked lim sup-game with vertex set V . Furthermore, recall that a play in Win has cost at most b in \mathcal{G}_{lim} if it visits vertices of rank greater than b only finitely often.

In the qualitative case, the behavior of visiting a certain set of vertices only finitely often is formalized by the qualitative co-Büchi condition

$$\text{CoBÜCHI}(F) = \{\rho \in V^\omega \mid \text{inf}(\rho) \cap F = \emptyset\} ,$$

where $\text{inf}(\rho)$ denotes the set of vertices occurring infinitely often in ρ . Clearly, Player 0 has a strategy of cost at most b in \mathcal{G}_{lim} if and only if she wins $(\mathcal{A}, \text{Win} \cap \text{CoBÜCHI}(\{v \in V \mid \text{rk}(v) > b\}))$. This observation gives rise to the following remark.

Remark 10. Let \mathfrak{G} be a class of qualitative games such that the games in $\{(\mathcal{A}, \text{Win} \cap \text{CoBüchi}(F)) \mid (\mathcal{A}, \text{Win}) \in \mathfrak{G}, F \subseteq V, V \text{ is vertex set of } \mathcal{A}\}$ can be solved in time $t(|\mathcal{G}|, |F|)$ and space $s(|\mathcal{G}|, |F|)$, where t and s are monotonic functions.

Then, the following problem can be solved in time $t(|\mathcal{G}_{\text{lim}}|, n)$ and space $s(|\mathcal{G}_{\text{lim}}|, n)$: “Given some game $\mathcal{G}_{\text{lim}} \in \mathfrak{G}_{\text{lim}}^{\text{RNK}}$ with n vertices as well as some bound $b \in \mathbb{N}$, does Player 0 win \mathcal{G}_{lim} with respect to b ?”

In this case, we solve vertex-ranked lim sup-games via a decision procedure for solving qualitative games as-is. Such a procedure trivially exists if the winning conditions of games from \mathfrak{G} are closed under intersection with co-Büchi conditions. Thus, we obtain solvability of a wide range of classes of vertex-ranked lim sup-games, e.g., co-Büchi-, parity-, Muller-, Streett- and Rabin games.

We now turn our attention to the latter case described above: We consider classes $\mathfrak{G}_{\text{lim}}^{\text{RNK}}$ where a play is only determined to be winning or losing in a game from \mathfrak{G} due to some infinite suffix. Formally, we say that a qualitative winning condition $\text{Win} \subseteq V^\omega$ is *prefix-independent* if for all infinite plays $\rho \in V^\omega$ and all play prefixes $\pi \in V^*$, we have $\rho \in \text{Win}$ if and only if $\pi\rho \in \text{Win}$. A qualitative game is prefix-independent if its winning condition is prefix-independent. A class of games is prefix-independent if every game in the class is prefix-independent. This notion allows us to formalize the claim made in the second bullet point above.

Theorem 11. Let \mathfrak{G} be a proper prefix-independent class of qualitative games where each $\mathcal{G} \in \mathfrak{G}$ can be solved in time $t(|\mathcal{G}|)$ and space $s(|\mathcal{G}|)$, where t and s are monotonic functions.

Then, the following problem can be solved in time $\mathcal{O}(n^3 + n^2 \cdot t(|\mathcal{G}_{\text{lim}}|))$ and space $\mathcal{O}(n + s(|\mathcal{G}_{\text{lim}}|))$: “Given some game $\mathcal{G}_{\text{lim}} \in \mathfrak{G}_{\text{lim}}^{\text{RNK}}$ with n vertices and some bound $b \in \mathbb{N}$, does Player 0 win \mathcal{G}_{lim} with respect to b ?”

Let \mathfrak{G} be a proper prefix-independent class of games and let $\mathcal{G}_{\text{lim}} \in \mathfrak{G}_{\text{lim}}^{\text{RNK}}$. Moreover, let $b \in \mathbb{N}$. Intuitively, in order to solve the b -threshold problem for \mathcal{G}_{lim} , we adapt the classic algorithm for solving prefix-independent qualitative games (cf., e.g., the work by Chatterjee, Henzinger, and Piterman [26]). Thereby, we repeatedly compute the set of vertices from which Player 0 has a strategy of cost at most b in the corresponding vertex-ranked sup-game \mathcal{G}_{sup} and remove their 0-attractor from the game similarly to the construction of a decision procedure for vertex-ranked sup-games in the proof of Theorem 7.

We claim that Player 0 has a strategy with cost at most b in \mathcal{G}_{lim} if and only if v_I was removed during that above construction.

In order to prove Theorem 11, we first show that, if Player 0 does not win a sup-game from any vertex, then she also does not win the corresponding lim sup-game from any vertex. Recall that for a qualitative or quantitative game \mathcal{G} with vertex set V we write \mathcal{G}_v to denote the game \mathcal{G} with its initial vertex replaced by $v \in V$. All other components, i.e., the structure of the arena and the cost-function, remain unchanged. We write $W_i^b(\mathcal{G})$ to denote the set of all vertices v such that Player i has a strategy of cost at most b , if $i = 0$, or greater than b , if $i = 1$, in \mathcal{G}_v .

Lemma 12. *Let $\mathcal{G}_{\text{lim}} = (\mathcal{A}, \text{RANK}^{\text{lim}}(\text{Win}, \text{rk}))$ be a vertex-ranked lim sup-game with vertex set V such that Win is prefix-independent and such that for each $v \in V$ the vertex-ranked sup-game $(\mathcal{G}_{\text{sup}})_v$ is determined. If $W_0^b(\mathcal{G}_{\text{sup}}) = \emptyset$, then $W_0^b(\mathcal{G}_{\text{lim}}) = \emptyset$.*

Proof. Let V be the vertex set of \mathcal{G}_{sup} and \mathcal{G}_{lim} . Since $W_0^b(\mathcal{G}_{\text{sup}}) = \emptyset$ and since for all $v \in V$ the game $(\mathcal{G}_{\text{sup}})_v$ is determined, we obtain $W_1^b(\mathcal{G}_{\text{sup}}) = V$. For each $v \in V$, let τ'_v be a strategy for Player 1 in $(\mathcal{G}_{\text{sup}})_v$ with cost greater than b . We now define a single strategy τ for Player 1 in \mathcal{G}_{lim} with cost greater than b . For each $\pi = v_0 \cdots v_j \in V^*$ we define $\tau(\pi) = \tau'_{v_k}(v_k \cdots v_j)$, where $k = \max\{k' \mid \text{rk}(v_{k'-1}) > b\}$, with $\max \emptyset = 0$. We claim that τ has cost greater than b in all $(\mathcal{G}_{\text{lim}})_v$. Since the cost-function Cost is identical in all $(\mathcal{G}_{\text{lim}})_v$ this claim is formalized as $\text{Cost}(\tau) > b$.

Let $\rho = v_0 v_1 v_2 \cdots$ be a play of \mathcal{G}_v consistent with τ . If there are infinitely many positions j with $\text{rk}(v_j) > b$, then $\text{Cost}(\rho) > b$. Thus, assume the opposite and let j be the maximal position with $\text{rk}(v_j) > b$. Then the suffix $\rho' = v_{j+1} v_{j+2} v_{j+3} \cdots$ of ρ is consistent with $\tau'_{v_{j+1}}$. Since ρ' does not encounter any vertices of rank greater than b , while $\text{Cost}(\rho') > b$ due to ρ' being consistent with a strategy of cost greater than b , we obtain $\rho' \notin \text{Win}$. This implies $\rho \notin \text{Win}$ due to prefix-independence of Win . Hence, $\text{Cost}(\rho) = \infty$, which, together with the statement above, implies $\text{Cost}(\tau) > b$. \square

We are now able to prove Theorem 11 using Lemma 12 as a building block for showing the correctness of the approach outlined above.

Proof of Theorem 11. Given $\mathcal{G}_{\text{lim}} = (\mathcal{A}, \text{RANK}^{\text{lim}}(\text{Win}, \text{rk}))$ with vertex set V of size n , we define $\mathcal{G}_0 = \mathcal{G}_{\text{sup}}$, as well as $X_j = W_0^b(\mathcal{G}_j)$, $A_j = \text{Attr}_0(X_j)$, which is computed in the arena of \mathcal{G}_j , and $\mathcal{G}_{j+1} = \mathcal{G}_j \setminus A_j$ for all $j \in \mathbb{N}$. As we only

remove vertices from the games \mathcal{G}_j , we obtain $\mathcal{G}_{j+1} \sqsubseteq \mathcal{G}_j$. Thus, the series of games stabilizes at $j = n$ at the latest, i.e., $\mathcal{G}_j = \mathcal{G}_n$ for all $j \geq n$. We define $A = \bigcup_{j \leq n} A_j$ and $\mathcal{G}' = \mathcal{G}_n$ and claim that Player 0 has a strategy with cost at most b in \mathcal{G} if and only if $v_I \in A$. We first argue that this suffices to show the desired result.

First note that since \mathfrak{G} is proper, $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$ is proper as well. Thus, Theorem 7 is applicable to $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$. Let dec_{sup} be the decision procedure deciding whether or not Player 0 has a strategy with cost at most b in games from $\mathfrak{G}_{\text{sup}}^{\text{RNK}}$, as constructed in the proof of that theorem. The decision procedure dec_{sup} can be easily modified to return $W_0^b(\mathcal{G}_j)$ instead of a yes/no-answer by applying it to each $(\mathcal{G}_j)_v$ individually. This, however, is only possible since we assume \mathfrak{G} to be proper, as the second condition of the definition of a proper family of games allows us to solve each $(\mathcal{G}_j)_v$. This modified procedure dec'_{sup} runs in time at most $\mathcal{O}(n^2 + n \cdot t(|\mathcal{G}|))$ and space $\mathcal{O}(n) + s(|\mathcal{G}|)$, where $t(|\mathcal{G}|)$ and $s(|\mathcal{G}|)$ are the time and space required to solve \mathcal{G} , respectively.

For $j \in \{0, \dots, n\}$, the decision procedure dec_{lim} first computes \mathcal{G}_j in linear time in n and reusing the space used for solving \mathcal{G}_{j-1} . It then computes X_j requiring a single call to the modified dec_{sup} . It subsequently computes A_j in time $\mathcal{O}(n)$ and space $\mathcal{O}(n)$. Finally, it returns false if and only if v_I is in the arena of \mathcal{G}_n . In total, we obtain a runtime of dec_{lim} of $\mathcal{O}(n^3 + n^2 \cdot t(|\mathcal{G}|))$. The only additional memory required by dec_{lim} is that for storing the sets X_j and A_j , the size of which is bounded from above by n . The games \mathcal{G}_j can be stored by reusing the memory occupied by \mathcal{G} , due to $\mathcal{G}_j \sqsubseteq \mathcal{G}_{j-1}$. Hence, the procedure dec_{lim} requires space $\mathcal{O}(n) + s(|\mathcal{G}|)$.

It remains to show that Player 0 indeed has a strategy with cost at most b in \mathcal{G} if and only if $v_I \notin \mathcal{A}_n$, i.e., if $v_I \in A$. To this end, first assume $v_I \in A$ and note that we have $A_j \supseteq X_j$. However, for each two $j \neq j'$, we have $A_j \cap A_{j'} = \emptyset$ and, in particular, $X_j \cap X_{j'} = \emptyset$. Hence, for each $v \in A$ there exists a unique j such that $v \in A_j$.

We define the strategy σ for Player 0 in \mathcal{G} inductively such that any play consistent with σ only descends through the X_j . Formally, we construct σ such that it satisfies the following invariant:

Let $\rho = v_0 v_1 v_2 \dots$ be a play consistent with σ and let $k \in \mathbb{N}$. If $v_k \in A_j \setminus X_j$, then $v_{k+1} \in \bigcup_{j' \leq j} A_{j'} \cup X_{j'}$. Moreover, if $v_k \in (A_j \setminus X_j) \cap V_0$, then the move to v_{k+1} is the move prescribed by the attractor strategy of Player 0 towards X_j . If $v_k \in X_j$, then $v_{k+1} \in X_j \cup \bigcup_{j' < j} A_{j'} \cup X_{j'}$.

Clearly, this invariant holds true for $\pi = v_I$. Thus, let $\pi = v_0 \cdots v_k$ be a play prefix consistent with σ . If $v_k \in V_1$, let v^* be an arbitrary successor of v_k in \mathcal{G} and assume towards a contradiction that πv^* violates the invariant. If $v_k \in A_j \setminus X_j$, then in \mathcal{G}_j there exists an edge from v_k leading to some vertex $v^* \notin A_j$, a contradiction to the definition of the attractor. If, however, $v_k \in X_j$ and $v^* \notin X_j \cup \bigcup_{j' < j} A_{j'} \cup X_{j'}$, then Player 1 has a strategy τ in $(\mathcal{G}_j)_{v^*}$ with cost greater than b . Thus, a play that begins in v_k , moves to v^* and is consistent with τ afterwards has cost greater b , i.e., Player 0 does not have a strategy with cost at most b in $(\mathcal{G}_j)_{v_k}$, a contradiction to $v_k \in X_j = W_0^b(\mathcal{G}_j)$. Hence, πv^* satisfies the invariant for each successor v^* of $v_k \in V_1$.

Now assume $v_k \in V_0$ and first let $v_0 \in A_j \cup X_j$ for some $j \in \mathbb{N}$. Let σ_j^A be an attractor strategy for Player 0 towards X_j . If $v_k \in A_j \setminus X_j$, we define $\sigma(\pi) = \sigma_j^A(v_k)$, which satisfies the invariant due to the definition of the attractor strategy. If, however, $v_k \in X_j$, let k' be minimal such that $v_{k''} \in X_j$ for all k'' with $k' \leq k'' \leq k$. Moreover, let σ_j^v be a strategy for Player 0 such that every play consistent with σ_j^v in \mathcal{G}_j with initial vertex v has cost at most b . Such a strategy exists due to $X_j = W_0^b(\mathcal{G}_j)$. We define $\sigma(\pi) = \sigma_j^{v_{k'}}(v_{k'} \cdots v_k)$, which satisfies the invariant to similar reasoning as above.

In order to show $\text{Cost}(\sigma) \leq b$, let $\rho = v_0 v_1 v_2 \cdots$ be a play consistent with σ . Due to the invariant of σ and since $v_0 \in A$, the play ρ descends through the A_j and the X_j , i.e., once it encounters some X_j , it never moves to any $A_{j'} \setminus X_{j'}$ with $j' \geq j$ nor to any $X_{j'}$ with $j' > j$. Also, ρ stabilizes in some X_j , i.e., there exists a $k \in \mathbb{N}$ such that $v_{k'} \in X_j$ for all $k' \geq k$, as σ prescribes moves according to the attractor strategy towards X_j when in $A_j \setminus X_j$. Moreover, due to the definition of σ , the suffix $\rho' = v_k v_{k+1} v_{k+2} \cdots$ is consistent with $\sigma_j^{v_k}$, i.e., we obtain $\rho' \in \text{Win}$ and that the maximal vertex-rank encountered in ρ is at most b . As Win is prefix-independent, we obtain $\rho \in \text{Win}$ as well as $\limsup_{k \rightarrow \infty} \text{rk}(v_k) \leq b$. Hence, $\text{RANK}^{\text{lim}}(\text{Win}, \text{rk})(\rho) \leq b$, which concludes this direction of the proof.

Now assume $v_I \notin A$ and consider \mathcal{G}' with vertex set $V \setminus A$. Since the construction of the \mathcal{G}_j stabilized, we have $A_j = X_j = W_0^b(\mathcal{G}') = \emptyset$, i.e., Player 1 has a strategy with cost greater than b from any starting vertex in \mathcal{G}' . Due to Lemma 12, this implies that he has such a strategy from every vertex in $\mathcal{G}_{\text{sup}} \setminus A$, call it τ . Note that there exists no Player-0-vertex in $V \setminus A$ that has an outgoing edge leading into A , as this would contradict the definition of the Player-0-attractors A_j . Hence, τ is a strategy for Player 1 in \mathcal{G} as well and we retain $\text{Cost}(\tau) > b$. \square

Intuitively, we prove Theorem 11 by constructing a strategy σ for Player 0 by “stitching together” the attractor-strategies towards her winning regions in the decreasing vertex-ranked sup-games and the winning strategies for her in the respective vertex-ranked sup-games. As each play consistent with that strategy descends down the hierarchy of sup-games thus constructed, we can reuse the memory states of the winning strategies in these games when implementing σ . Thus, a monotonic upper bound on the size of strategies with cost at most b in \mathcal{G}_{sup} is an upper bound on the size of such strategies in \mathcal{G}_{lim} as well.

Corollary 13. *Let \mathfrak{G} be a proper prefix-independent class of qualitative games such that, if Player 0 wins \mathcal{G} , then she has a finite-state winning strategy of size at most $m(|\mathcal{G}|)$, where m is a monotonic function.*

If Player 0 wins $\mathcal{G}_{\text{lim}} \in \mathfrak{G}_{\text{lim}}^{\text{RNK}}$, then she has a finite-state winning strategy σ_{lim} with $|\sigma_{\text{lim}}| \in \mathcal{O}(m(|\mathcal{G}_{\text{lim}}|))$ in \mathcal{G}_{lim} . Furthermore, if winning strategies for Player 0 in the games in \mathfrak{G} are effectively constructible, then σ_{lim} is effectively constructible.

Moreover, in order to find the optimal b such that Player 0 wins \mathcal{G}_{lim} with respect to b , we can again employ a binary search analogously to the case of vertex-ranked sup-games. Thus, we can determine the optimal such b in time $\mathcal{O}(\log(M)(n^3 + n^2 \cdot t(|\mathcal{G}_{\text{lim}}|)))$ and space $\mathcal{O}(n + s(|\mathcal{G}_{\text{lim}}|))$, where M again denotes the number of ranks assigned to vertices in a given vertex-ranked lim sup-game.

Having thus defined both quantitative reductions and a canonical target for such reductions, we now give examples of how to solve quantitative games using this tools.

5. Applications

In this section, we give examples of how to use quantitative reductions and vertex-ranked games to solve quantitative games. First, in Section 5.1, we formally introduce a quantitative variant of request-response games, which we call request-response games with costs, and show how to solve such games using quantitative reductions and vertex-ranked sup-request-response games. Second, in Section 5.2, we recall the definition of quantitative Muller games due to McNaughton [22] and show how to reduce such games to vertex-ranked safety games via quantitative reductions. Finally, in Section 5.3, we show that vertex-ranked games are useful in their own right, by showing how to use them to synthesize controllers that are resilient against disturbances.

5.1. Reducing Request-Response Games with Costs to Vertex-Ranked Request-Response Games

Recall that a play satisfies the qualitative request-response condition if every request that is opened is eventually answered. We extend this condition to a quantitative one by equipping the edges of the arena with costs and measuring the maximal cost incurred between opening and answering a request.

Fix some arena \mathcal{A} with vertex set V and set E of edges. Formally, the qualitative request-response condition $\text{REQRES}(\Gamma)$ consists of a family of so-called request-response pairs $\Gamma = (Q_c, P_c)_{c \in [d]}$, where $d \in \mathbb{N}$, $d > 0$, and where $Q_c, P_c \subseteq V$ for all $c \in [d]$. Player 0 wins a play according to this condition if each visit to some vertex from Q_c is answered by some later visit to a vertex from P_c , i.e., we define

$$\text{REQRES}((Q_c, P_c)_{c \in [d]}) = \{v_0 v_1 v_2 \cdots \in V^\omega \mid \forall c \in [d] \forall j \in \mathbb{N}. v_j \in Q_c \text{ implies } \exists j' \geq j. v_{j'} \in P_c\} .$$

We say that a visit to a vertex from Q_c *opens a request for condition c* and that the first visit to a vertex from P_c afterwards *answers the request for that condition*.

Proposition 14 ([13]). *Request-response games with n vertices and d request-response pairs can be solved in time $\mathcal{O}(n^2 d^2 2^d)$.*

Furthermore, let \mathcal{G} be a request-response game with d request-response pairs. If Player 0 has a winning strategy in \mathcal{G} , then she has a finite-state winning strategy of size at most $d2^d$.

We extend this qualitative winning condition to a quantitative one using families of cost functions $\text{Cost} = (\text{Cost}_c)_{c \in [d]}$, where $\text{Cost}_c: E \rightarrow \mathbb{N}$ for each $c \in [d]$ and lift the cost functions Cost_c to play infixes π in \mathcal{A} by adding up the costs along π . The cost-of-response for a request for condition c at position j is then defined as

$$\text{REQRESCOR}_c(v_0 v_1 v_2 \cdots, j) = \begin{cases} \min\{\text{Cost}_c(v_j \cdots v_{j'}) \mid j' \geq j \text{ and } v_{j'} \in P_c\} & \text{if } v_j \in Q_c , \\ 0 & \text{otherwise ,} \end{cases}$$

with $\min \emptyset = \infty$, which naturally extends to the (total) *cost-of-response*

$$\text{REQRESCOR}(\rho, j) = \max_{c \in [d]} \text{REQRESCOR}_c(\rho, j) .$$

Finally, we define the *request-response condition* with costs as

$$\text{COSTREQRES}(\Gamma, \text{Cost})(\rho) = \sup_{j \rightarrow \infty} \text{REQRESCOR}(\rho, j) ,$$

i.e., this condition measures the maximal cost incurred by any request in ρ .

We call a game $\mathcal{G} = (\mathcal{A}, \text{COSTREQRES}(\Gamma, \text{Cost}))$ a request-response game with costs. We denote the largest cost assigned to any edge by W . As we assume the functions Cost_c to be given in binary encoding, the largest cost W assigned to an edge may be exponential in the size of \mathcal{G} .

If all Cost_c assign zero to every edge, then the request-response condition with costs coincides with the qualitative request-response condition. In general, however, the request-response condition with costs is a strengthening of the classic request-response condition: If some play ρ has finite cost according to the condition with costs, then it is winning for Player 0 according to the qualitative condition, but not vice versa.

Remark 15. *Let $\mathcal{G} = (\mathcal{A}, \text{COSTREQRES}(\Gamma, \text{Cost}))$ be a request-response game with costs. If a strategy σ for Player 0 in \mathcal{G} has finite cost, then σ is a winning strategy for Player 0 in the qualitative game $(\mathcal{A}, \text{REQRES}(\Gamma))$.*

This remark together with a detour via qualitative request-response games yield a cap for request-response games with costs.

Lemma 16. *Let \mathcal{G} be a request-response game with costs with n vertices, d request-response pairs, and largest cost of an edge W . If Player 0 has a strategy with finite cost in \mathcal{G} , then she also has a strategy with cost at most $d2^d nW$.*

Proof. Let $\mathcal{G} = (\mathcal{A}, \text{COSTREQRES}(\Gamma, \text{Cost}))$ and let $\mathcal{G}' = (\mathcal{A}, \text{REQRES}(\Gamma))$ be a qualitative request-response game obtained by disregarding the cost functions of \mathcal{G} . Moreover, let σ be a strategy with finite cost for \mathcal{G} . Due to Remark 15, the strategy σ is winning for Player 0 in \mathcal{G}' as well, hence Player 0 wins \mathcal{G}' . Thus, due to Proposition 14, she has a winning strategy σ' of size at most $d2^d$ in \mathcal{G}' . Let σ' be implemented by the memory structure \mathcal{M} and let $b = d2^d nW$. We show $\text{COSTREQRES}(\Gamma, \text{Cost})(\sigma') \leq b$.

Let $\rho = v_0 v_1 v_2 \dots$ be a play consistent with σ' and assume towards a contradiction $\text{COSTREQRES}(\Gamma, \text{Cost})(\rho) > b$. Then there exist $c \in [d]$ and $j \in \mathbb{N}$ such that $\text{REQRESCOR}_c(\rho, j) > b$. As each edge has cost at most W , the request for condition c opened at position j is not answered for at least $d2^d n$ steps, i.e., we obtain $v_{j'} \notin P_c$ for all $j' \leq j + d2^d n$. Let $\text{ext}(\rho) =$

$(v_0, m_0)(v_1, m_1)(v_2, m_2) \cdots$. Since $|\mathcal{M}| \leq d2^d$, there exists a vertex repetition on the play infix $(v_j, m_j) \cdots (v_{j+d2^d n}, m_{j+d2^d n})$ of $\text{ext}(\rho)$, say at positions k and k' with $j \leq k < k' \leq j + d2^d n$. Thus, the play $\rho' = v_0 \cdots v_k (v_{k+1} \cdots v_{k'})^\omega$ is consistent with σ' .

In ρ' , however, a request for condition c is opened at position $j \leq k$. Since we have $j \leq k' \leq j + d2^d n$, this request is not answered in the play infix $v_j \cdots v_k \cdots v_{k'}$, i.e., it is never answered. Hence, $\rho' \notin \text{REQRES}(\Gamma)$, which contradicts σ' being a winning strategy for Player 0 in \mathcal{G}' . \square

Having obtained a cap for request-response games with costs, we can now turn to the main result of this section: Request-response games with costs are reducible to vertex-ranked sup-request-response games. In order to show this, we use a memory structure that keeps track of the costs incurred by the requests open at each point in the play [7].

Lemma 17. *Let \mathcal{G} be a request-response game with costs with n vertices, d request-response pairs, and highest cost of an edge W . Then $\mathcal{G} \leq_{\mathcal{M}}^{b+1} \mathcal{G}'$ for $b = d2^d nW$, some memory structure \mathcal{M} of size $\mathcal{O}(2nb^d)$, and a vertex-ranked sup-request-response game \mathcal{G}' with d request-response pairs.*

Proof. Let $\mathcal{G} = (\mathcal{A}, \text{COSTREQRES}(\Gamma, \text{Cost}))$ with initial vertex v_I . Recall that b is a cap of \mathcal{G} due to Lemma 16. We first define the memory structure \mathcal{M} . Intuitively, we use it to keep track of the currently open requests and the costs they have incurred up to the cap b . Once the cost of a single request incurs a cost greater than b , the memory structure raises a Boolean flag, which indicates that Player 1 can unbound the cost of that request.

Let $r: [d] \rightarrow \{\perp\} \cup [b+1] = \{\perp, 0, \dots, b\}$ be a function mapping conditions c to the cost $r(c) \in [b+1]$ they have incurred so far, or to $r(c) = \perp$ if no request for that condition is pending. We call such a function a *request-function* and denote the set of all request functions by R . We define the initial request function r_I such that $r_I(c) = 0$ if $v_I \in Q_c$ and $r_I(c) = \perp$ otherwise. In order to be able to access the current vertex during the update of the memory structure, we store it in the memory structure as well. By accessing the current vertex together with the vertex that we move to, we are thus able to obtain the cost of the traversed edge. Finally, we store a flag that indicates whether or not the bound b has been exceeded. Hence, we define the set of memory states $M = V \times R \times \{0, 1\}$ with the initial memory state $m_I = (v_I, r_I, 0)$.

We define the update function $\text{Upd}((v, r, f), v') = (v', r', f')$ by performing the following steps in order:

- For each $c \in [d]$, if $v \in P_c$, set $r'(c) = \perp$. Otherwise, set $r'(c) = r(c)$.
- For each $c \in [d]$, if $r'(c) \neq \perp$, set $r'(c) = r'(c) + \text{Cost}_c((v, v'))$.
- Now, if there exists a condition c such that $r'(c) > b$, then set $r'(c) = \perp$ for all c and set $f' = 1$. Otherwise, set $f' = f$.
- For each $c \in [d]$, if $v' \in Q_c$, set $r'(c)$ to $\max\{r'(c), 0\}$ where $\max\{\perp, 0\} = 0$.

Note that we replicate the current vertex v in the memory state in order to be able to access it in the update of the memory state during the move to v' , thereby attaining access to the traversed edge (v, v') .

We obtain $\mathcal{M} = (M, m_I, \text{Upd})$. Clearly, we have $|\mathcal{M}| \in \mathcal{O}(2nb^d)$, i.e., \mathcal{M} is of exponential size in d , but only of polynomial size in n and W .

Using this definition, we obtain that if $\text{COSTREQRES}(\Gamma, \text{Cost})(\rho) \leq b$, then the extended play $\text{ext}(\rho)$ remains in vertices of the form $(v, v, r, 0)$. Dually, if $\text{COSTREQRES}(\Gamma, \text{Cost})(\rho) > b$, then $\text{ext}(\rho)$ eventually moves to vertices of the form $(v, v, r, 1)$ and remains there ad infinitum.

Let $\Gamma = (Q_c, P_c)_{c \in [d]}$. It remains to define the vertex-ranking function $\text{rk}: V \times M \rightarrow \mathbb{N}$, as well as the family of request-response pairs Γ' for \mathcal{G}' . We define the former as

$$\text{rk}(v, v, r, f) = \begin{cases} \max\{0, \max_{c \in [d]} r(c)\} & \text{if } f = 0 \text{ and} \\ b + 1 & \text{otherwise} \end{cases}$$

and the latter as

$$\Gamma' = (Q'_c, P'_c)_{c \in [d]}, \text{ where } Q'_c = Q_c \times Q_c \times R \times \{0, 1\} \text{ and} \\ P'_c = P_c \times P_c \times R \times \{0, 1\} \text{ for all } c \in [d].$$

Note that $\rho \in \text{REQRES}(\Gamma)$ if and only if $\text{ext}(\rho) \in \text{REQRES}(\Gamma')$.

We first argue that the ranking functions included in the memory state indeed contain information about the cost incurred so far by open requests as long as the cost of the play does not exceed b . In order to do so, let $\rho = v_0 v_1 v_2 \cdots$ be some play in \mathcal{G} with $\text{COSTREQRES}(\Gamma, \text{Cost})(\rho) \leq b$ and let $\text{ext}(\rho) = (v_0, v_0, r_0, f_0)(v_1, v_1, r_1, f_1)(v_2, v_2, r_2, f_2) \cdots$ be its extension.

Intuitively, since the cost of ρ does not exceed the bound b , the flags f_j are not raised and the request functions track the cost of all requests precisely.

More formally if, for some $c \in [d]$ and some $j \in \mathbb{N}$ with $v_j \in Q_c$, we have $\text{REQRES}\text{COR}_c(\rho, j) = b'$, then $r_{j'}(c) = b'$, where $j' \geq j$ is the earliest position at which the request for c opened at position j is answered. Dually, if $r_{j'}(c) = b'$ for some $j' \in \mathbb{N}$, some $b' \in \mathbb{N}$, and some $c \in [d]$, then $\text{REQRES}\text{COR}_c(\rho, j) \geq b'$, where $j \leq j'$ is the earliest position at which the request for condition c is opened without being answered prior to position j' . In particular, if $r_{j'}(c) = b'$ with j' , b' , and c as above and we additionally have $v_{j'} \in P_c$, then $\text{REQRES}\text{COR}_c(\rho, j) = b'$.

We define $\mathcal{G}' = (\mathcal{A} \times \mathcal{M}, \text{RANK}^{\text{sup}}(\text{REQRES}(\Gamma'), \text{rk}))$. Moreover, since Γ' is the extension of Γ to the vertices of $\mathcal{A} \times \mathcal{M}$, the game \mathcal{G}' contains d many request-response pairs.

It remains to show $\mathcal{G} \leq_{\mathcal{M}}^{b+1} \mathcal{G}'$. Recall that, since we do not name a $(b+1)$ -correction function explicitly, we implicitly use the $(b+1)$ -correction-function cap_{b+1} . Clearly, the first and second condition of the definition of a quantitative reduction hold true, i.e., the arena of \mathcal{G}' is $\mathcal{A} \times \mathcal{M}$ and cap_{b+1} is a $(b+1)$ -correction function. It remains to show the two latter conditions. To this end, let $\rho = v_0 v_1 v_2 \cdots \in V^\omega$ be a play in \mathcal{G} and let $\text{ext}(\rho) = (v_0, v_0, r_0, f_0)(v_1, v_1, r_1, f_1)(v_2, v_2, r_2, f_2) \cdots$ be its unique extended play in \mathcal{G}' . We use the shorthands $\text{Cost}_{\mathcal{G}} = \text{COSTREQRES}(\Gamma, \text{Cost})$ as well as $\text{Cost}_{\mathcal{G}'} = \text{RANK}^{\text{sup}}(\text{REQRES}(\Gamma'), \text{rk})$.

We first show $\text{Cost}_{\mathcal{G}}(\rho) = \text{Cost}_{\mathcal{G}'}(\text{ext}(\rho))$ for all ρ with $\text{Cost}_{\mathcal{G}}(\rho) < b+1$. Let $\text{Cost}_{\mathcal{G}}(\rho) = b' < b+1$ and note that this implies $\rho \in \text{REQRES}(\Gamma)$ and $\text{ext}(\rho) \in \text{REQRES}(\Gamma')$ as well as $f_j = 0$ for all $j \in \mathbb{N}$. As argued above, we obtain $\text{rk}(v_j, v_j, r_j, f_j) \leq b'$ for all j , which implies $\text{Cost}_{\mathcal{G}'}(\text{ext}(\rho)) \leq b'$. Moreover, let $c \in [d]$ and $j \in \mathbb{N}$ with $v_j \in Q_c$ such that $\text{REQRES}\text{COR}_c(\rho, j) = b'$. Since $b' < \infty$, such c and j exist. The play $\text{ext}(\rho)$ visits a vertex of rank b' at the position at which the request for condition c opened at position j is answered for the first time. Thus, $\text{Cost}_{\mathcal{G}'}(\text{ext}(\rho)) \geq b'$, which concludes this part of the proof.

It remains to show that $\text{Cost}_{\mathcal{G}'}(\text{ext}(\rho)) \geq \text{cap}_{b+1}(b+1) = b+1$ holds true for all ρ with $\text{Cost}_{\mathcal{G}}(\rho) \geq b+1$. To this end, let $\text{Cost}_{\mathcal{G}}(\rho) = b' \geq b+1$. As argued above, the extended play $\text{ext}(\rho)$ eventually moves to vertices of the form $(v, v, r, 1)$ and remains there. Hence, $\text{Cost}_{\mathcal{G}'}(\text{ext}(\rho)) = b+1$ if $\rho \in \text{REQRES}(\Gamma)$, i.e., if $\text{ext}(\rho) \in \text{REQRES}(\Gamma')$. If, however, $\rho \notin \text{REQRES}(\Gamma)$, then $\text{ext}(\rho) \notin \text{REQRES}(\Gamma')$ and hence, $\text{Cost}_{\mathcal{G}'}(\rho) = \infty > b+1$. \square

Thus, in order to solve a request-response game with costs with respect to some b , it suffices to solve a vertex-ranked sup-request-response game with respect to b . This, in turn, can be done by reducing the problem to that of solving a request-response game as shown in Theorem 7. Using this reduction together with the framework of quality-preserving reductions, we are able to provide an upper bound on the complexity of solving request-response games with costs with respect to some bound b .

Theorem 18. *The following decision problem is in EXPTIME: “Given some request-response game with costs \mathcal{G} and some bound $b \in \mathbb{N}$, does Player 0 have a strategy σ with $\text{Cost}(\sigma) \leq b$ in \mathcal{G} ?”*

Proof. Let \mathcal{G} contain n vertices, d request-response pairs, and let W be the largest cost assigned to any edge. We first construct the vertex-ranked sup-request-response game \mathcal{G}' from \mathcal{G} as shown in Lemma 17. Recall that \mathcal{G}' contains $\mathcal{O}(n(d2^d nW)^d)$ vertices and d request-response pairs. Due to the instantiation of Theorem 7 with the decision procedure for qualitative request-response games from Proposition 14, the game \mathcal{G}' can be solved with respect to b in time

$$\begin{aligned} \mathcal{O}(n(d2^d nW)^d + (n(d2^d nW)^d)^2 d^2 2^d) = \\ \mathcal{O}(n(d2^d nW)^d + n^2 (d2^d nW)^{2d} d^2 2^d) = \mathcal{O}(n^2 (d2^d nW)^{2d} d^2 2^d) . \end{aligned}$$

Due to $W^{2d} \in \mathcal{O}((2^{|\mathcal{G}|})^{|\mathcal{G}|}) = \mathcal{O}(2^{|\mathcal{G}|^2})$, this is exponential in the description length of \mathcal{G} . \square

Moreover, solving request-response games is known to be EXPTIME-hard [27]. Thus, solving quantitative request-response games with costs via quantitative reductions is asymptotically optimal.

Furthermore, by leveraging our results on the sizes of memory structures in vertex-ranked sup-games we obtain an upper bound on the size of strategies with a given cost in request-response games with costs.

Lemma 19. *Let \mathcal{G} be a request-response game with costs with n vertices, d request-response pairs, and largest cost of an edge W . If Player 0 has a strategy in \mathcal{G} with finite cost, then she also has a strategy in \mathcal{G} with finite cost of size at most $\mathcal{O}(nb^d d^2)$, where $b = d2^d nW$.*

Proof. Let σ be a strategy for Player 0 in \mathcal{G} with finite cost. Due to Lemma 16, Player 0 has a strategy σ' in \mathcal{G} with cost at most $d2^d nW$. Let $\mathcal{G}'_{\text{sup}}$ be

the vertex-ranked sup-request-response game constructed in the proof of Lemma 17 and recall that $\mathcal{G}'_{\text{sup}}$ has d request-response pairs as well.

Due to Theorem 2, and since b is a cap of \mathcal{G} due to Lemma 16, Player 0 has a strategy σ'_{sup} with cost at most $d2^d nW$ in $\mathcal{G}'_{\text{sup}}$. Let \mathcal{G}' be the qualitative request-response game corresponding to $\mathcal{G}'_{\text{sup}}$, i.e., the game played in the same arena as $\mathcal{G}'_{\text{sup}}$ in which a play is winning for Player 0 if and only if it has finite cost in $\mathcal{G}'_{\text{sup}}$.

Clearly, the strategy σ'_{sup} is winning for Player 0 in \mathcal{G}' as well. By applying Proposition 14 we obtain that Player 0 has a winning strategy of size at most $d2^d$ in \mathcal{G}' . By furthermore applying Corollary 9 and Theorem 3, we obtain that Player 0 has a strategy of finite cost of size $\mathcal{O}(2nb^d d2^d) = \mathcal{O}(nb^d d2^d)$. \square

Finally, the optimization problem of finding the minimal b' such that Player 0 wins a request-response game with costs \mathcal{G} with respect to b' can be solved in exponential time as well. Recall that if Player 0 wins \mathcal{G} with respect to some b' , then she also wins it with respect to all $b'' \geq b'$. Since we can assume $b' \leq b = d2^d nW$, we can perform a binary search for b' on the interval $\{0, \dots, b\}$. Hence, the optimal b' can be found in time $\mathcal{O}(\log(b)(n^2 b^{2d} d^2 2^d))$.

5.2. Reducing Quantitative Muller Games to Vertex-Ranked Safety Games

Having shown how our framework can be used to find optimal strategies in request-response games with costs in a structured and modular way, we now show how it can be used to greatly simplify existing methods for finding such strategies. To this end, we show how to reduce quantitative Muller games to vertex-ranked safety games, i.e., vertex-ranked games in which it is the aim of Player 0 to avoid a certain set of undesirable vertices. In order to do so, we leverage techniques introduced by Neider, Rabinovich, and Zimmermann [28].

Let \mathcal{A} be some arena with vertex set V and recall that the qualitative Muller condition is defined via a partition of 2^V into $(\mathcal{F}_0, \mathcal{F}_1)$ as

$$\text{MULLER}(\mathcal{F}_0, \mathcal{F}_1) = \{\rho \in V^\omega \mid \text{inf}(\rho) \in \mathcal{F}_0\} ,$$

where $\text{inf}(\rho)$ denotes the set of vertices that are visited infinitely often by ρ . Thus, Player i wins ρ if and only if $\text{inf}(\rho) \in \mathcal{F}_i$.

McNaughton introduced a quantitative characterization of the Muller condition by assigning a score to each prefix of a play and each subset of the set of vertices [22]. In order to characterize the set of vertices visited

infinitely often during a play, the score of a subset F measures how often F has been visited completely without leaving it. For a play ρ , the limit inferior of the score of $\text{inf}(\rho)$ tends towards infinity, while the limit inferior of the score for all other sets is zero [22].

Formally, for any set $F \subseteq V$ with $F \neq \emptyset$, the score $\text{Score}_F(\pi)$ is defined inductively using an accumulator that stores the vertices of F that have already been visited, as follows:

$$\begin{aligned} (\text{Acc}_F(\varepsilon), \text{Score}_F(\varepsilon)) &= (\emptyset, 0) \\ (\text{Acc}_F(\pi v), \text{Score}_F(\pi v)) &= \\ &\begin{cases} (\emptyset, 0) & \text{if } v \notin F \\ (\emptyset, \text{Score}_F(\pi) + 1) & \text{if } v \in F \text{ and } \text{Acc}_F(\pi) = F \setminus \{v\} \\ (\text{Acc}_F(\pi) \cup \{v\}, \text{Score}_F(\pi)) & \text{otherwise} \end{cases} \end{aligned}$$

We generalize the score-function to families \mathcal{F} of subsets of vertices, i.e., $\mathcal{F} \subseteq 2^V$, by defining $\text{Score}_{\mathcal{F}}(\pi) = \max_{F \in \mathcal{F}}(\text{Score}_F(\pi))$ and to infinite plays by defining $\text{Score}_{\mathcal{F}}(v_0 v_1 v_2 \dots) = \sup_{j \rightarrow \infty} \text{Score}_{\mathcal{F}}(v_0 \dots v_j)$. This definition inspires the quantitative Muller condition, which is defined as

$$\text{QUANTMULLER}(\mathcal{F}_0, \mathcal{F}_1)(\rho) = \text{Score}_{\mathcal{F}_1}(\rho) .$$

We obtain a cap for such games via leveraging a result by Fearnley and Zimmermann [29].

Lemma 20. *Let $\mathcal{G} = (\mathcal{A}, \text{Cost})$ be a quantitative Muller game. If Player 0 has a strategy σ with finite cost in \mathcal{G} , then she has a strategy σ' with $\text{Cost}(\sigma') \leq 2$.*

Proof. Let $\mathcal{G} = (\mathcal{A}, \text{QUANTMULLER}(\mathcal{F}_0, \mathcal{F}_1))$. Since $\text{Cost}(\sigma) < \infty$, for every play ρ consistent with σ and every prefix π of ρ , we have that there exists an upper bound on $\text{Score}_F(\pi)$ for all $F \in \mathcal{F}_1$. Moreover, as the score of $\text{inf}(\rho)$ tends towards ∞ , this implies $\text{inf}(\rho) \in \mathcal{F}_0$, i.e., σ is a winning strategy for the qualitative Muller game $\mathcal{G}' = (\mathcal{A}, \text{MULLER}(\mathcal{F}_0, \mathcal{F}_1))$.

It is known that, since Player 0 wins \mathcal{G}' , she has a strategy σ' with $\text{Score}_{\mathcal{F}_1}(\pi) \leq 2$ for all prefixes π of all plays consistent with σ' [29]. Thus, we directly obtain $\text{QUANTMULLER}(\mathcal{F}_0, \mathcal{F}_1)(\sigma') \leq 2$. \square

We now show how to reduce quantitative Muller games to vertex-ranked sup-safety games based on previous work by Neider, Rabinovich, and Zimmermann [28]. Recall that a safety game is a very simple qualitative game,

in which it is Player 0's goal to avoid a certain set of undesirable vertices. Formally, the safety condition is defined via a set $U \subseteq V$ as

$$\text{Safety}(U) = \{v_0v_1v_2 \cdots \in V^\omega \mid \forall j \in \mathbb{N} v_j \notin U\} .$$

In order to construct the safety game, we define an equivalence relation over play prefixes, such that two play prefixes are equivalent if they have the same accumulator and the same score with respect to all $F \in \mathcal{F}_1$. The constructed safety game uses as vertices representatives of the equivalence classes of all play prefixes that have a cost of at most 2 for all $F \in \mathcal{F}_1$. Moreover, it mimics play prefixes π of cost at most 2 in the Muller game by moving to some vertex π' such that the score and the accumulator are equal in π and π' for all $F \in \mathcal{F}_1$. We show how to lift this qualitative construction to the setting of quantitative games by providing a quantitative reduction from quantitative Muller games to vertex-ranked sup-safety games.

Lemma 21. *Let \mathcal{G} be a quantitative Muller game with n vertices. There exists a memory structure \mathcal{M} of size at most $(n!)^3$ and a vertex-ranked sup-safety game \mathcal{G}' such that $\mathcal{G} \leq_{\mathcal{M}}^3 \mathcal{G}'$.*

Proof. Let $\mathcal{G} = (\mathcal{A}, \text{Cost})$ with $\text{Cost} = \text{QUANTMULLER}(\mathcal{F}_0, \mathcal{F}_1)$. We say that two play prefixes π and π' are \mathcal{F}_1 -equivalent if they end in the same vertex and if, for each $F \in \mathcal{F}_1$, we have $\text{Acc}_F(\pi) = \text{Acc}_F(\pi')$ and $\text{Score}_F(\pi) = \text{Score}_F(\pi')$. In this case, we write $\pi \approx_{\mathcal{F}_1} \pi'$. For each play prefix π , we denote the \mathcal{F}_1 -equivalence-class of π by $[\pi]_{\approx_{\mathcal{F}_1}} = \{\pi' \in V^* \mid \pi \approx_{\mathcal{F}_1} \pi'\}$. Furthermore, for each set $\Pi \subseteq V^*$ of play prefixes we define the $\approx_{\mathcal{F}_1}$ -quotient of Π as $(\Pi / \approx_{\mathcal{F}_1}) = \{[\pi]_{\approx_{\mathcal{F}_1}} \mid \pi \in \Pi\}$. For the sake of readability, we omit the index \mathcal{F}_1 of the score-function and the index $\approx_{\mathcal{F}_1}$ of the equivalence class for the remainder of this proof wherever possible without introducing ambiguity.

Let $\text{Plays}_{\leq 2} = \{\pi \in V^* \mid \text{Score}_{\mathcal{F}_1}(\pi) \leq 2\}$ be the set of play prefixes whose score is at most two. We define the set of memory states $M = (\text{Plays}_{\leq 2} / \approx) \cup \{\perp\}$, the initial memory state $m_I = [v_I]$, and the update function Upd as $\text{Upd}(\perp, v) = \perp$, $\text{Upd}(\pi, v) = [\pi v]$ if $\text{Score}(\pi v) \leq 2$ and $\text{Upd}(\pi, v) = \perp$ otherwise. We obtain $|M| \in \mathcal{O}(|\text{Plays}_{\leq 2} / \approx|)$. Since $|\text{Plays}_{\leq 2} / \approx| \leq (n!)^3$ due to Neider, Rabinovich, and Zimmermann [28], the memory structure \mathcal{M} is of size at most $(n!)^3$ as well.

A straightforward induction shows that this memory structure tracks the score of a play precisely as long as it does not exceed the value two on any prefix. More formally, it satisfies the following invariant:

Let $\pi = v_0 \cdots v_j$ be a play prefix in \mathcal{G} such that $\text{Score}(v_0 \cdots v_k) \leq 2$ for all k with $0 \leq k \leq j$. Moreover, let $\text{Upd}^+(\pi) = \pi'$. Then $\pi \approx \pi'$.

Recall $\text{Safety}(U) = \{v_0 v_1 v_2 \cdots \in V^\omega \mid \forall j \in \mathbb{N}. v_j \notin U\}$. We define the vertex-ranked sup-safety game $\mathcal{G}' = (\mathcal{A} \times \mathcal{M}, \text{RANK}^{\text{sup}}(\text{Safety}(V \times \{\perp\}), \text{rk}))$, with $\text{rk}(v, \pi) = \text{Score}(\pi)$ for all $\pi \in \text{Plays}_{\leq 2}$, and $\text{rk}(v, \perp) = 3$.

Let $\text{Cost}' = \text{RANK}^{\text{sup}}(\text{Safety}(V \times \{\perp\}), \text{rk})$. Clearly, the first two items of the definition of $\mathcal{G} \leq_{\mathcal{M}}^3 \mathcal{G}'$ hold true. It remains to show $\text{Cost}(\rho) = \text{Cost}'(\text{ext}(\rho))$ for all ρ with $\text{Cost}(\rho) < 3$ and $\text{Cost}'(\text{ext}(\rho)) \geq 3$ for all other ρ .

First, let $\rho = v_0 v_1 v_2 \cdots$ be some play with $\text{Cost}(\rho) \leq 2$ and let $\text{ext}(\rho) = (v_0, m_0)(v_1, m_1)(v_2, m_2) \cdots$. Then $\text{Score}(v_0 \cdots v_j) \leq 2$ for all $j \in \mathbb{N}$. Thus, due to the invariant above and the definition of rk , we obtain $\text{rk}(v_j, m_j) = \text{Score}(v_0 \cdots v_j)$ for all $j \in \mathbb{N}$, which implies $\text{Cost}'(\text{ext}(\rho)) = \text{Cost}(\rho)$.

Towards a proof of the latter statement, let $\rho = v_0 v_1 v_2 \cdots$ be a play with $\text{Cost}(\rho) \geq 3$ and let j be the minimal position such that $\text{Cost}(v_0 \cdots v_j) = 3$. Since $\text{Cost}(v_0 \cdots v_j) = \text{Score}_F(v_0 \cdots v_j)$ for some $F \in \mathcal{F}_1$ and since the score is at most incremented by one during each step, we obtain $\text{Score}_F(v_0 \cdots v_{j-1}) = 2$, $\text{Acc}_F(v_0 \cdots v_{j-1}) = F \setminus \{v_j\}$, and $v_j \in F$. Let $\text{Upd}^+(v_0 \cdots v_{j-1}) = \pi'$. Due to the invariant we obtain $\text{Score}_F(\pi') = 2$ and $\text{Acc}_F(\pi') = F \setminus \{v_j\}$. Thus, $\text{Score}_F(\pi' v_j) = 3$, hence $\text{ext}(v_0 \cdots v_j) = (v_0, m_0) \cdots (v_j, \perp)$, which implies $\text{ext}(\rho) \notin \text{Safety}(V \times \{\perp\})$, which in turn yields $\text{Cost}'(\text{ext}(\rho)) = \infty > 3$. \square

Thus, in order to solve a quantitative Muller game with respect to some b , it suffices to solve a vertex-ranked sup-safety game \mathcal{G}' with respect to b . Recall that this is only constructive if Player 0 wins \mathcal{G}' with respect to $b < 3$, i.e., only in this case are we able to construct a strategy with cost at most b for her in \mathcal{G} . Otherwise, Theorem 2 yields that there exists a strategy of cost ∞ for Player 1 in \mathcal{G} , but we cannot construct such a strategy from his strategy of cost greater than two in \mathcal{G}' . This is consistent with results of Neider et al. [28] and with the fact that Muller conditions are in a higher level of the Borel hierarchy than safety conditions. Hence, qualitative Muller games cannot be reduced to safety games.

We can, however, solve the resulting vertex-ranked sup-safety game with respect to a given bound by solving a qualitative safety game as shown in Theorem 7. Using this reduction together with the framework of quality-preserving reductions, we obtain an upper bound on the complexity of solving quantitative Muller games with respect to some bound b .

Theorem 22. *The following problem can be solved in time $\mathcal{O}((n!)^3)$: “Given some quantitative Muller game \mathcal{G} with n vertices and some bound $b \in \mathbb{N}$, does Player 0 win \mathcal{G} with respect to b ?”*

Proof. Given \mathcal{G} , we first construct the vertex-ranked sup-safety game \mathcal{G}' as shown in Lemma 21. Recall that \mathcal{G}' contains at most $(n!)^3$ vertices. Due to Theorem 7 and the fact that safety games can be solved in linear time in the number of vertices, \mathcal{G}' can indeed be solved in time at most $\mathcal{O}((n!)^3)$ with respect to a given bound b . \square

Analogously to the reasoning leading to Corollary 9 on Page 25 and to Corollary 13 on Page 31, we are now also able to provide an upper bound on the size of strategies for Player 0 in quantitative Muller games. Since both players have positional winning strategies in safety games, an application of Theorem 3 to Lemma 21 yields that if Player 0 has a strategy with cost at most three in a quantitative Muller game \mathcal{G} with n vertices, then she also has a strategy in \mathcal{G} with the same cost and of size at most exponential in n .

Moreover, we can bound the complexity of the optimization problem for quantitative Muller games as follows: Finding the minimal b such that Player 0 has a strategy of cost at most b in \mathcal{G} requires solving at most three safety games of size in $\mathcal{O}((n!)^3)$. Thus, the optimization problem for quantitative Muller games can be solved in factorial time.

5.3. Fault Resilient Strategies for Safety Games

We now demonstrate the flexibility and versatility of vertex-ranked games in their own right. To this end, we consider the problem of synthesizing a controller for a reactive system that is embedded into some environment. This setting is typically modeled as an infinite game in which Player 0 and Player 1 take the roles of the controller and of the environment, respectively. Here, we consider safety games, i.e., we assume that the specification for the controller is given as a game in which it is the aim of Player 0 to keep the play inside a safe subset of the vertices.

Dallal, Neider, and Tabuada [23] argue that this setting is not sufficiently expressive to correctly model a real-world scenario, since it assumes that Player 0 can accurately predict the effect of her actions on the state of the system. In a realistic setting, in contrast, faults may occur, i.e., an action chosen by a controller may be executed incorrectly, or it may not be executed at all.

In order to model such faults, Dallal, Neider, and Tabuada introduce arenas with faults $\mathcal{A}_F = (V, V_0, V_1, E, F, v_I)$, which consist of an arena without faults (V, V_0, V_1, E, v_I) and a set of faults $F \subseteq V_0 \times V$. In such an arena, whenever it is the turn of Player 0, say at vertex v , a fault $(v, v') \in F$ may occur, resulting in the play continuing in vertex v' instead of that chosen by Player 0. Moreover, Dallal et al. consider safety conditions, i.e., qualitative winning conditions of the form $\text{Safety}(U) = \{v_0 v_1 v_2 \cdots \mid \forall j \in \mathbb{N}. v_j \notin U\}$ for some $U \subseteq V$. Hence, it is the aim of Player 0 to keep the play outside the “unsafe” set of vertices U . If the play enters the set U , it is declared winning for Player 1. The task at hand is to compute a fault-resilient strategy for Player 0 that forces the play to remain inside $V \setminus U$ and that can “tolerate” as many faults as possible.

Safety games without faults are solved by a simple attractor construction: As soon as the play enters $W_1 = \text{Attr}_1(U)$, Player 1 can play consistently with his attractor strategy towards U in order to win the play. Thus, it is the aim of Player 0 to keep the play inside $W_0 = V \setminus W_1$.

Dallal, Neider, and Tabuada solve the problem of computing fault-resilient strategies for safety games by adapting the classic algorithm for solving safety games to this setting. In doing so, they obtain a value $\text{val}(v)$ for each vertex v that denotes the minimal number of faults that need to occur in order for the play to reach W_1 , if Player 0 plays well. Furthermore, they show that val can be computed in polynomial time in $|V|$. Finally, due to the existence of positional winning strategies for both players in safety games, they obtain $\text{val}(v) \in [n] \cup \{\infty\}$ for all $v \in V$. Then, a fault-resilient strategy for Player 0 is one that maximizes the minimal value $\text{val}(v)$ witnessed during any play. Dallal, Neider, and Tabuada construct such a strategy on the fly during the computation of $\text{val}(v)$ [23].

This task can, however, easily be reframed as a vertex-ranked game in the arena $\mathcal{A} = (V, V_0, V_1, E, v_I)$, which we obtain from \mathcal{A}_F by omitting the faults. In that game, we assign to each vertex the rank $\text{rk}(v) = |V| - \text{val}(v)$ if $\text{val}(v) \in [n]$ and $\text{rk}(v) = 0$ otherwise, i.e., if $\text{val}(v) = \infty$. Then, Player 0 has a strategy with cost at most b in $\mathcal{G}' = (\mathcal{A}, \text{RANK}^{\text{sup}}(\text{Safety}(U), \text{rk}))$ if and only if she has a winning strategy in the original safety game with faults that tolerates at least $|V| - b$ faults.

This formulation as a vertex-ranked game enables further study of games in arenas with faults. Here, we require the winning condition to be a safety condition in order to compute $\text{val}(v)$. In recent work, we have shown how to compute this value for more complex qualitative winning conditions [30].

If $\text{val}(v)$ is effectively computable for a given qualitative winning condition, one can easily obtain fault-resilient strategies by formulating the task as a vertex-ranked game as demonstrated.

Finally, the formulation as a vertex-ranked game yields a method to compute eventually-fault-resilient strategies, i.e., strategies that are resilient to a large number of faults after a finite “start-up” phase. In order to obtain such strategies, it suffices to view the resulting vertex-ranked games as a lim sup-game instead of a sup-game and to solve it optimally as described in Section 4.2.

6. Conclusion

In this work, we have lifted the concept of reductions, which has yielded a multitude of results in the area of qualitative games, to quantitative games. We have shown that this novel concept exhibits the same useful properties for quantitative games as it does for qualitative ones and that it furthermore retains the quality of strategies.

Additionally, we have provided two very general types of quantitative games that serve as targets for quantitative reductions, namely vertex-ranked sup games and vertex-ranked lim sup-games. For both kinds of games we have shown a polynomial overhead on the complexity of solving them with respect to some bound, on the memory necessary to achieve a given cost, and on the complexity of determining the optimal cost that either player can ensure.

Finally, we have demonstrated the versatility of these tools by using them to solve quantitative request-response games and quantitative Muller games and by showing how to solve the problem of computing fault-resilient strategies in safety games via vertex-ranked games. This last formulation enables a general study of games with faults, even in the presence of more complex winning conditions than the safety condition considered by Dallal et al. [23] and in this work. We are currently investigating how to leverage vertex-ranked games for the synthesis of fault-resistant strategies in parity games.

Further research continues in two additional directions: Firstly, while the framework of quantitative reductions and vertex-ranked games yields upper bounds on the complexity of solving quantitative games, it does not directly yield lower bounds on the complexity of the problems under investigation. Consider, for example, the threshold problem for parity games with costs, which is PSPACE-complete [7]. It is possible to reduce this problem to that of solving a vertex-ranked parity game of exponential size and linearly

many colors similarly to the reduction presented in this work, which yields an EXPTIME-algorithm. It remains open how to use quantitative reductions to obtain an algorithm for this problem that only requires polynomial space.

Secondly, another goal for future work is the establishment of an analogue to the Borel hierarchy for quantitative winning conditions. In the qualitative case, this hierarchy establishes clear boundaries for reductions between infinite games, i.e., a game whose winning condition is in one level of the Borel hierarchy cannot be reduced to one with a winning condition in a lower level. Also, each game with a winning condition in the hierarchy is known to be determined [31]. To the best of our knowledge, it is open how to define such a hierarchy for quantitative winning conditions which exhibit similar properties.

Acknowledgements I would like to thank Martin Zimmermann for many fruitful discussions. Furthermore, I would like to thank the anonymous reviewers for their thorough and constructive comments which greatly improved this work.

References

- [1] A. Weinert, Quantitative Reductions and Vertex-Ranked Infinite Games, in: A. Orlandini, M. Zimmermann (Eds.), GandALF 2018, volume 277 of *EPTCS*, 2018, pp. 1–15. doi:10.4204/EPTCS.277.1.
- [2] L. Brim, J. Chaloupka, L. Doyen, R. Gentilini, J.-F. Raskin, Faster algorithms for mean-payoff games, *Form. Meth. in Sys. Des.* 38 (2011) 97–118. doi:10.1007/s10703-010-0105-x.
- [3] V. Bruyère, E. Filiot, M. Randour, J. Raskin, Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games, *Inf. Comput.* 254 (2017) 259–295. doi:10.1016/j.ic.2016.10.011.
- [4] V. Bruyère, Q. Hautem, M. Randour, Window parity games: an alternative approach toward parity games with time bounds, in: D. Cantone, G. Delzanno (Eds.), GandALF 2016, volume 226 of *EPTCS*, 2016, pp. 135–148. doi:10.4204/EPTCS.226.10.
- [5] A. Ehrenfeucht, J. Mycielski, Positional Strategies for Mean Payoff Games, *Internat. J. Game Theory* 8 (1979) 109–113. doi:10.1007/BF01768705.

- [6] N. Fijalkow, M. Zimmermann, Parity and Streett Games with Costs, *Log. Meth. in Comp. Sci.* 10 (2014). doi:10.2168/LMCS-10(2:14)2014.
- [7] A. Weinert, M. Zimmermann, Easy to Win, Hard to Master: Optimal Strategies in Parity Games with Costs, *Log. Meth. in Comp. Sci.* 13 (2017). doi:10.23638/LMCS-13(3:29)2017.
- [8] U. Zwick, M. Paterson, The complexity of mean payoff games on graphs, *Theo. Comp. Sci.* 158 (1996) 343–359. doi:10.1016/0304-3975(95)00188-3.
- [9] R. Bloem, K. Chatterjee, K. Greimel, T. A. Henzinger, G. Hofferek, B. Jobstmann, B. Könighofer, R. Könighofer, Synthesizing robust systems, *Act. Inf.* 51 (2014) 193–220. doi:10.1007/s00236-013-0191-5.
- [10] R. Bloem, K. Chatterjee, T. A. Henzinger, B. Jobstmann, Better Quality in Synthesis through Quantitative Objectives, in: *CAV 2009*, volume 5643 of *LNCS*, Springer, 2009, pp. 140–156. doi:10.1007/978-3-642-02658-4_14.
- [11] P. Bouyer, N. Markey, J. Olschewski, M. Ummels, Measuring Permissiveness in Parity Games: Mean-Payoff Parity Games Revisited, in: T. Bultan, P. Hsiung (Eds.), *ATVA 2011*, volume 6996 of *LNCS*, Springer, 2011, pp. 135–149. doi:10.1007/978-3-642-24372-1_11.
- [12] P. Cerný, K. Chatterjee, T. A. Henzinger, A. Radhakrishna, R. Singh, Quantitative Synthesis for Concurrent Programs, in: G. Gopalakrishnan, S. Qadeer (Eds.), *CAV 2011*, volume 6806 of *LNCS*, Springer, 2011. doi:10.1007/978-3-642-22110-1_20.
- [13] N. Wallmeier, P. Hütten, W. Thomas, Symbolic Synthesis of Finite-State Controllers for Request-Response Specifications, in: *CIAA 2003*, volume 2759 of *LNCS*, Springer, 2003, pp. 11–22. doi:10.1007/3-540-45089-0_3.
- [14] K. Chatterjee, T. A. Henzinger, F. Horn, Finitary winning in ω -regular games, *ACM Trans. Comput. Log.* 11 (2009). doi:10.1145/1614431.1614432.
- [15] P. Faymonville, M. Zimmermann, Parametric Linear Dynamic Logic, *Inf. Comput.* 253 (2017) 237–256. doi:10.1016/j.ic.2016.07.009.

- [16] O. Kupferman, N. Piterman, M. Y. Vardi, From liveness to promptness, *Form. Meth. in Sys. Des.* 34 (2009) 83–103. doi:10.1007/s10703-009-0067-z.
- [17] P. Bouyer, N. Markey, M. Randour, K. G. Larsen, S. Laursen, Average-energy games, *Act. Inf.* (2016). doi:10.1007/s00236-016-0274-1.
- [18] K. Chatterjee, L. Doyen, Energy parity games, *Theo. Comp. Sci.* 458 (2012) 49–60. doi:10.1016/j.tcs.2012.07.038.
- [19] K. Chatterjee, L. Doyen, T. A. Henzinger, J.-F. Raskin, Generalized Mean-payoff and Energy Games, in: K. Lodaya, M. Mahajan (Eds.), *FSTTCS 2010*, volume 8 of *LIPICs*, 2010, pp. 505–516. doi:10.4230/LIPICs.FSTTCS.2010.505.
- [20] Y. Velner, K. Chatterjee, L. Doyen, T. A. Henzinger, A. M. Rabinovich, J. Raskin, The complexity of multi-mean-payoff and multi-energy games, *Inf. Comput.* 241 (2015) 177–196. doi:10.1016/j.ic.2015.03.001.
- [21] S. Schewe, A. Weinert, M. Zimmermann, Parity Games with Weights, in: D. R. Ghica, A. Jung (Eds.), *CSL 2018*, volume 119 of *LIPICs*, Schloss Dagstuhl - LZI, 2018, pp. 36:1–36:17. doi:10.4230/LIPICs.CSL.2018.36.
- [22] R. McNaughton, Playing Infinite Games in Finite Time, in: A. Salomaa, D. Wood, S. Yu (Eds.), *A Half-Century of Automata Theory: Celebration and Inspiration*, World Scientific, 2000, pp. 73–91.
- [23] E. Dallal, D. Neider, P. Tabuada, Synthesis of safety controllers robust to unmodeled intermittent disturbances, in: *CDC 2016, IEEE*, 2016, pp. 7425–7430. doi:10.1109/CDC.2016.7799416.
- [24] G. H. Mealy, A Method for Synthesizing Sequential Circuits, *The Bell System Technical Journal* 34 (1955) 1045–1079. doi:10.1002/j.1538-7305.1955.tb03788.x.
- [25] A. Nerode, J. B. Remmel, A. Yakhnis, McNaughton Games and Extracting Strategies for Concurrent Programs, *Ann. Pure Appl. Logic* 78 (1996) 203–242. doi:10.1016/0168-0072(95)00032-1.

- [26] K. Chatterjee, T. A. Henzinger, N. Piterman, Algorithms for Büchi Games, in: GDV 2006, 2006. Available at <http://arxiv.org/abs/0805.2620>.
- [27] K. Chatterjee, T. A. Henzinger, F. Horn, The Complexity of Request-Response Games, in: LATA 2011, volume 6638 of *LNCS*, Springer, 2011, pp. 227–237. doi:10.1007/978-3-642-21254-3_17.
- [28] D. Neider, R. Rabinovich, M. Zimmermann, Down the Borel hierarchy: Solving Muller games via safety games, *Theor. Comput. Sci.* 560 (2014) 219–234. doi:10.1016/j.tcs.2014.01.017.
- [29] J. Fearnley, M. Zimmermann, Playing Muller Games in a Hurry, *Int. J. Found. Comput. Sci.* 23 (2012) 649–668. doi:10.1142/S0129054112400321.
- [30] D. Neider, A. Weinert, M. Zimmermann, Synthesizing Optimally Resilient Controllers, in: D. R. Ghica, A. Jung (Eds.), *CSL 2018*, volume 119 of *LIPICs*, Schloss Dagstuhl - LZI, 2018, pp. 36:1–36:17. doi:10.4230/LIPICs.CSL.2018.34.
- [31] D. A. Martin, Borel Determinacy, *Ann. of Math.* 102 (1975) 363–371. doi:10.2307/1971035.