Nash Social Welfare in Selfish and Online Load Balancing

VITTORIO BILÒ, University of Salento, Italy
GIANPIERO MONACO, University of L'Aquila, Italy
LUCA MOSCARDELLI, University of Chieti-Pescara, Italy
COSIMO VINCI, University of Salerno and Gran Sasso Science Institute, Italy

8

In load-balancing problems there is a set of clients, each wishing to select a resource from a set of permissible ones to execute a certain task. Each resource has a latency function, which depends on its workload, and a client's cost is the completion time of her chosen resource. Two fundamental variants of load-balancing problems are *selfish load balancing* (a.k.a. *load-balancing games*), where clients are non-cooperative selfish players aimed at minimizing their own cost solely, and *online load balancing*, where clients appear online and have to be irrevocably assigned to a resource without any knowledge about future requests. We revisit both problems under the objective of minimizing the *Nash Social Welfare*, i.e., the geometric mean of the clients' costs. To the best of our knowledge, despite being a celebrated welfare estimator in many social contexts, the Nash Social Welfare has not been considered so far as a benchmarking quality measure in load-balancing problems. We provide tight bounds on the price of anarchy of pure Nash equilibria and on the competitive ratio of the greedy algorithm under very general latency functions, including polynomial ones. For this particular class, we also prove that the greedy strategy is optimal, as it matches the performance of any possible online algorithm.

CCS Concepts: • **Theory of computation → Online algorithms**; **Solution concepts in game theory**; **Quality of equilibria**; **Scheduling algorithms;**

Additional Key Words and Phrases: Congestion games, Nash social welfare, pure Nash equilibrium, price of anarchy, online algorithms

ACM Reference format:
Cosimo Vinci, Vittorio Bilò, Gianpiero Monaco, and Luca Moscardelli. 2022. Nash Social Welfare in Selfish and Online Load Balancing. *ACM Trans. Econ. Comput.* 10, 2, Article 8 (October 2022), 41 pages.
https://doi.org/10.1145/3544978

A preliminary version of this work appeared in the proceedings of "The 16th Conference on Web and Internet Economics (WINE 2020)" [21].
This work was partially supported by the Italian MIUR PRIN 2017 Project ALGADIMAR "Algorithms, Games, and Digital Markets."
The current affiliation of Cosimo Vinci is University of Salerno. A substantial part of this work was done when he was affiliated with Gran Sasso Science Institute (his previous affiliation).
Authors' addresses: C. Vinci (corresponding author), Department of Information Engineering, Electrical Engineering and Applied Mathematics, University of Salerno, Via Giovanni Paolo II 132, 84084 Fisciano (SA) and Gran Sasso Science Institute, Viale Francesco Crispi 7, 67100 L'Aquila, Italy; email: cvinci@unisa.it; V. Biló, Department of Mathematics and Physics "Ennio De Giorgi", University of Salento, Provinciale Lecce-Arnesano, 73100 Lecce, Italy; email: vittorio.bilo@unisalento.it; G. Monaco, Department of Information Engineering Computer Science and Mathematics, University of L'Aquila, Via Vetoio, 67100 Coppito (AQ), Italy; email: gianpiero.monaco@univaq.it; L. Moscardelli, Department of Economic Studies, University of Chieti-Pescara, Viale Pindaro 42, 65127 Pescara, Italy; email: luca.moscardelli@unich.it.

2167-8375/2022/10-ART8 $15.00
https://doi.org/10.1145/3544978

1 INTRODUCTION

In load-balancing problems there is a set of clients, each wishing to select a resource from a set of permissible ones to execute a certain task. Each resource has a latency function, which depends on its workload, and a client's cost is the completion time of her chosen resource. These problems stand at the foundations of the Theory of Computing and have been studied under a variety of objective functions, such as the maximum client's cost (a.k.a. the makespan) [66–68, 75] and the average weighted client's cost (see Reference [39] for an excellent survey).

Two extensively studied variants of load-balancing problems are *selfish load balancing* [92] (a.k.a. *load-balancing games*) and *online load balancing* [66]. Selfish load balancing, where clients are non-cooperative selfish players aimed at minimizing their own cost solely, constitutes a notable subclass of *weighted congestion games* [83] and, as such, enjoys some nice theoretical properties. For instance, load-balancing games always admit pure Nash Equilibria [69]. Moreover, under the assumption that all tasks have unitary weight (*unweighted congestion games*), any best-response dynamics converges to a pure Nash Equilibrium in polynomial time [1]. In online load balancing, instead, clients appear online and have to be irrevocably assigned to a resource without any knowledge about future requests.

1.1 Social Welfare Functions and the Nash Social Welfare

Interpreting the set of clients of a load-balancing problem as a society, and adopting the terminology of welfare economics, the makespan and the average weighted client's cost objective functions get called, respectively, the ***Egalitarian*** **(ESW)** and the ***Utilitarian Social Welfare*** **(USW)**. In the following, we aim at providing a comprehensive picture of the comparison between the Nash Social Welfare function and other well-studied social welfare functions, such as ESW and USW.

Let us first focus on the case in which utilities represent profits to be maximized. In this case, also the social welfare functions must be maximized. Generally speaking, given an n-dimensional vector $\boldsymbol{x} = (x_1, \ldots, x_n)$ of positive utilities, the USW is defined as $\frac{1}{n}\sum_i x_i$, while the ESW as $\min_i x_i$. Moreover, the ***Nash Social Welfare*** **(NSW)** [80] is defined as $(\prod_i x_i)^{\frac{1}{n}}$, i.e., as the geometric mean of the clients' utilities.

The NSW, besides possessing appealing mathematical properties that make it a powerful tool in many economical scenarios,[1] is also a celebrated welfare measure in all those settings in which the well-being of a population is addressed by a combination of personal profits and, therefore, has to be maximized [30, 48, 64, 81]. In these settings, in fact, NSW satisfies the following set of appealing properties [71, 78, 79]:

- *Pareto optimality and monotonicity*, prescribing that an outcome in which the utility of a player improves while all other utilities do not get worse has to be preferred;
- *Pigou-Dalton transfer principle*, stating that a transfer of utilities from the rich to the poor leaving unchanged the sum of utilities (i.e., the USW) is desirable as long as it does not bring the rich to a poorer situation than the poor;
- *independence of unconcerned agents*, requiring that the preference relation between two utility profiles does not change if a player having the same utility in both of them changes her utility to a same value in both profiles;
- *independence of common utility scale*, stating that the preference relation between two utility profiles does not change if both of them are multiplied by the same value.

[1]For instance, the NSW is related to the computation of equilibria in Fisher markets [12, 30, 46] and fair outcomes in allocation problems [4, 28, 30, 35, 37].

Pareto optimality is a measure of *efficiency* that is satisfied by the USW, but not achieved by the ESW (think, for instance, to profit vectors $(10, 3)$ and $(3, 3)$; while USW prefers the first, ESW is indifferent between the two). Conversely, the Pigou–Dalton transfer principle is a measure of *social equity* realized by the ESW, but not achieved by the USW (given profit vectors $(3, 3)$ and $(5, 1)$, ESW prefers the first, while USW is indifferent between the two). Thus, with respect to these two properties, we can say that NSW is a better social welfare function, given that it is able to achieve at the same time the equity guaranteed by the ESW and the efficiency proper of the USW.

Moreover, while independence of common utility scale is also satisfied by both the USW and the ESW, independence of unconcerned agents is satisfied by the USW, but it is not achieved by ESW[2] (given profit vectors $(1, 3)$ and $(1, 2)$, ESW is indifferent between the two, but if the first agent changes her utility to 4, then the preference relation between the two profiles changes, as $(4, 3)$ is preferred by ESW to $(4, 2)$, but not vice versa).

Another interesting property of the NSW, which is not fulfilled by the ESW and the USW, is *player-specific scale-independence*, stating that the preference relation between two utility profiles does not change if the utilities of different players are scaled by player-specific values in both profiles. More formally, if profit vector $(x_1, \ldots, x_n)$ is preferred to profit vector $(x'_1, \ldots, x'_n)$, then it also holds that, given any n positive real constants $c_1, \ldots, c_n$, profit vector $(c_1x_1, \ldots, c_nx_n)$ is preferred to profit vector $(c_1x'_1, \ldots, c_nx'_n)$. This property, which strengthens that of the independence of common utility scale, yields a stronger social equity with respect to that of the Pigou–Dalton transfer principle. In fact, while the latter implicitly assumes that players' utilities are in the same scale (and therefore, the more equal they are, the fairer the outcome is), player-specific scale-independence applies to settings in which the player's utilities may intrinsically differ by several orders of magnitude. To this respect, consider a situation in which the range of utilities of different players is very wide, and this discrepancy is not due to an unfair outcome, but is rather a direct consequence of the different nature of the players' strategies. Here, the utility of few influential players may be able to determine almost completely the value of the ESW or USW. More precisely, on the one hand, players with an intrinsic low profit are taken in higher consideration by ESW: Consider 10 players and the profit vectors $(11, 1{,}000, 1{,}000, \ldots, 1{,}000)$ and $(10, 2{,}000, 2{,}000, \ldots, 2{,}000)$, in the context of a game in which the first player's maximum possible profit is 11 and the other players' maximum possible profit is 2,000. ESW prefers the first vector, in which the first player is experiencing her best possible profit and the other players one half of their best possible profit, even if in the second vector almost all players are experiencing their best possible profit (and the first player a profit being 10/11 of her maximum possible profit). On the other hand, players with an intrinsic high profit may be able to determine almost completely the value of the USW: Consider 10 players and the profit vectors $(2{,}000, 10, 10, \ldots, 10)$ and $(1{,}900, 20, 20, \ldots, 20)$, in the context of a game in which the first player's maximum possible profit is 2,000 and the other players' maximum possible profit is 20. USW prefers the first vector, in which the first player is experiencing her best possible profit and the other players' one half of their best possible profit, even if in the second vector almost all players are experiencing their best possible profit (and the first player a profit being 19/20 of her maximum possible profit). It is worth noticing that in both the above-mentioned scenarios, the NSW prefers the second vector, thus taking more properly into account the well-being of all players, independently on how large (or small) is the order of magnitude of their utilities.

[2]Actually, if we consider the leximin ordering social function [78, 79] that is a refinement of ESW, then the property of independence of unconcerned agents is recovered. More precisely, in the leximin social function, once sorted in non-decreasing order, profit vectors are compared as follows: Consider the first component for which one vector is different from another one: It is preferred the vector having this component greater than the corresponding one of the other vector. Roughly speaking, ties are resolved looking at further components of the sorted vectors.

When moving from profit maximization to cost minimization, as it happens in the setting of congestion and load-balancing games considered in this work, the social welfare functions need to be minimized and things slightly change. First, while the definitions of USW and NSW remain unchanged, the ESW has to be defined as $\max_i x_i$. Moreover, while USW and ESW keep verifying the same properties satisfied in the profit maximization case, it is worth noticing that the NSW no longer satisfies the Pigou-Dalton transfer principle (given cost vectors $(5, 1)$ and $(3, 3)$, NSW prefers the first, while the second should be favored). However, NSW still enjoys the remaining aforementioned properties: Pareto optimality, independence of unconcerned agents, independence of common scale, and player-specific scale-independence. Among these, we consider the latter of fundamental importance, and we provide in the following a clarifying example, along the same lines of the ones provided in the case of profit maximization. Recall that, in the case of profit maximization, players with an intrinsic low profit are taken in higher consideration by the ESW, while USW is mostly influenced by those with an intrinsic high profit. Interestingly, unlike the case of profit maximization, when considering cost minimization it holds that the players being able to determine almost completely the values of the ESW and the USW are, for both social welfare functions, those with an intrinsic higher cost.

To this respect, consider a communication or transportation network in which there are some "heavy" clients aiming at communicating at a global scale (e.g., national traffic) and other "light" ones corresponding to more local communication requests (e.g., local traffic). In the context of load-balancing games, a similar situation arises when there are both "heavy" clients, owning highly time-consuming tasks, and "light" clients, requiring for the processing of faster tasks. In these situations, the cost of few specific clients (the heavy ones) may determine almost completely the values of the ESW and the USW, while the NSW is able to take into account the costs of all players, including the light ones. For example, consider 10 clients such that the first has a minimum possible cost of 1,000 and the others have a minimum possible cost of 10, and we have two alternative cost vectors $(1{,}400, 20, 20, \dots, 20)$ and $(1{,}500, 10, 10, \dots, 10)$. Both ESW and USW prefer the first vector, while NSW prefers the second one. In this alternative, 9 players are experiencing the best possible cost, with the first player charged 1.5 times her minimum possible cost. In the first vector, instead, while the first player is only slightly better (i.e., she is experiencing a cost of 1.4 times the optimal one), all other players are doubling their cost. In fact, in the context of load-balancing problems, since clients with big jobs need to wait longer times either way, improving their cost by some amount will not have the same impact as it would have if we reduced the cost of small job owners by the same amount. In general, the benefit of a speedup is often better captured by the relative improvement (e.g., a one-minute improvement for a one-hour job will not have the same impact as a one-minute improvement for a five-minute job). Therefore, the NSW objective is able to distribute the speedup, and in this way it aims to balance fairness and efficiency.

The above example can be exploited to obtain evidence of the fact that this type of equity is a corollary of player-specific scale-independence. In fact, the NSW is a robust social welfare function and does not change its induced preference relation if we scale the costs of the first player by a factor equal to $1/100$, thus making the players' minimum costs in the same scale. More generally, in load-balancing games, we could define a client's cost as the ratio between the completion time of her chosen resource and the completion time she could obtain when being the only client in the system (i.e., when she is the unique user of the fastest resource in her strategy set), and with this respect the NSW is the only correct measure to use when averaging normalized results, that is, results that are presented as ratios to reference values [57].

For all these reasons, we believe that, although losing some of its tremendous expressiveness when moving from profit maximization to cost minimization, the NSW remains a significant enough measure which should be considered for investigation, as done with the ESW and the USW.

1.2 Related Work

Selfish Load Balancing. The literature concerning the efficiency of Nash equilibria in selfish load balancing is highly tied with that of its superclass of congestion games. In the following, we first focus on results for the mostly studied case of the USW. In this setting, it is assumed that all clients selecting the same resource experience the same cost.

The efficiency of pure Nash equilibria in congestion games has been first considered in References [7, 42], where it has been independently shown that the price of anarchy is 5/2 and $(3 + \sqrt{5})/2$ for, respectively, unweighted and weighted congestion games with affine latency functions. These bounds have been extended to load-balancing games in Reference [33]. However, under the additional assumption that the game is symmetric (i.e., all resources are available to any client), the price of anarchy improves to 4/3 [76]. Exact bounds for both weighted and unweighted congestion games with polynomial latency functions have been given in Reference [2] and extended to even unweighted load-balancing games and symmetric weighted load-balancing games in References [14, 62], respectively. These results have been further generalized in Reference [23], where it is proved that, under general latency functions encompassing polynomial ones, the worst-case price of anarchy of both symmetric weighted congestion games and unweighted congestion games is attained by load-balancing instances. This worst-case behavior, however, does not occur under identical resources, where load-balancing games exhibit better performance with respect to general congestion games. For instance, for affine latency functions, the price of anarchy drops to 2.012067 for unweighted games [33, 90] and to 9/8 for symmetric weighted games [76]. Tight bounds for this last class of games under polynomial and more general latency functions have been given in References [23, 61].

For the class of non-atomic congestion games (a variant assuming that each client's task is infinitesimally small with respect to the workload required by the whole society and suited to model communication and transportation networks), a number of papers [13, 84, 87, 88] provide bounds on the price of anarchy under general latency functions and prove that, under mild assumptions, they are tight even for a two-node network with two parallel links. An interesting connection between load-balancing games and non-atomic congestion games has been uncovered in Reference [58], where it is shown that, under fairly general latency functions, the price of anarchy of unweighted symmetric load-balancing games coincides with that of non-atomic congestion games.

Less has been done for the ESW. The study of the price of anarchy was initiated in Reference [74], where weighted congestion games of m parallel links with linear latency functions are considered. The price of anarchy in this case is $\Theta(\frac{\log m}{\log \log m})$. The lower bound was shown in Reference [74] and the upper bound in Reference [50]. For load-balancing games, the price of anarchy is $\Theta(\frac{\log n}{\log \log n})$, where n is the number of players [61], while for unweighted congestion games it is $\Theta(\sqrt{n})$ [42]. The price of anarchy of non-atomic congestion games with general non-decreasing latency function is proven to be $\Omega(n)$ in Reference [85].

Online Load Balancing. The performance of greedy load balancing, with respect to the USW and under affine latency functions, has been studied in References [8, 33, 90]. The authors of Reference [8] consider a more general model where each client has a load vector denoting her impact on each resource (i.e., how much her assignment to a resource will increase its load), and the objective is to minimize the L_p norm of the load of the resources. Their results, together with Reference [33], imply a competitive ratio of the greedy algorithm equal to $3 + 2\sqrt{2} \approx 5.8284$ for the USW. This bound carries over also to the case of weighted clients where the objective is to minimize the average weighted latency. A combination of the results in References [33, 90] shows that the competitiveness of greedy load balancing is 17/3 for different resources and between 4 and $\frac{2}{3}\sqrt{21}+1 \approx 4.05505$ for identical resources. The competitive ratio of the greedy algorithm applied to congestion games with general latency functions has been characterized in Reference [23].

A different online algorithm (usually termed one-round walk starting from the empty state) for load balancing is analyzed in References [18, 44]. Its competitive ratio is shown to be $2+\sqrt{5}$ under affine latency functions. Bounds for the case of polynomial latencies are given in References [15, 24, 73], while more general latency functions are considered in References [23, 91], with respect to atomic and non-atomic congestion games, respectively.

Concerning the ESW, most of the literature investigates the case of identical resources, usually termed *machines* [3, 10, 53, 56, 63, 66, 72]. We notice that the scheduling problem with related (respectively, identical) machines is a special case of the weighted load-balancing problem with linear latency functions (respectively, identical resources with linear latency functions). For m identical machines, the greedy algorithm achieves a competitive ratio of exactly $2-\frac{1}{m}$ [66], and this bound is proven the best possible one for $m=2,3$ in Reference [53]. The currently best-known algorithm achieves a competitive ratio of 1.9201 [56] for any m, and no algorithm can achieve a competitive ratio better than 1.88 [89]. For related machines, a tight bound of $\log m$ is shown in References [6, 9], while the case of unrelated machines with the objective of minimizing the norm of the machine loads is considered in Reference [31].

1.3 Our Contribution

We revisit both selfish and online load balancing under the objective of minimizing the NSW. To the best of our knowledge, this is the first work adopting the NSW as a benchmarking quality measure in load-balancing problems. Indeed, the performances of Nash equilibria in load-balancing games, as well as the competitive ratio for online load balancing, have been widely studied under the USW and the ESW (see Related Work section for further details), but never under the NSW. Furthermore, most of the literature on NSW is about the problem of allocating a set of items among players with the aim of maximizing the NSW [30, 35, 37, 46, 48, 64], while in this work the NSW is considered as a quality measure to be minimized.

We analyze the price of anarchy [74] of pure Nash equilibria (the loss in optimality due to selfish behavior) and the competitive ratio of online algorithms (the loss in optimality due to lack of information) under very general latency functions. These questions have been widely addressed under the USW and the ESW, but never under the NSW.

We notice that, by adopting the NSW as a new metric, we are not going to modify the set of Nash equilibria, but only their social values. The main difference between the NSW and the classical notion of USW consists in the fact that, while in the latter the players' costs are summed, in the former they are multiplied. This may lead to think that, by turning the costs into their logarithms, a classical utilitarian analysis can be easily adapted to deal with the NSW. Actually, this is not the case. In fact, on the one hand, using this idea for bounding a performance ratio (e.g., the price of anarchy or the competitive ratio), one obtains a bound on the ratio between two logarithms (each one having the product of the players' costs as argument). On the other hand, we are interested in bounding the ratio between the argument of these logarithms, and there is no direct correlation between these two ratios (notice that logarithm of the latter ratio is equal to the difference between the corresponding utilitarian social costs, and therefore it is not related to the former one). Thus, the analysis of the NSW requires different proof arguments. To have another evidence of this fact, it is worth noticing that the results obtained for the NSW substantially differ from the ones holding

Table 1. Tight Bounds on the Performance of Load Balancing with Polynomial Latency Functions of Maximum Degree p, under the NSW and the USW

	NSW	USW
Weighted	2^p	$(\Phi_p)^{p+1} \sim \Theta\left(\frac{p}{\log(p)}\right)^{p+1}$, [2]
Unweighted	2^p	$\frac{(k+1)^{2p+1}-k^{p+1}(k+2)^p}{(k+1)^{p+1}-(k+2)^p+(k+1)^p-k^{p+1}} \sim \Theta\left(\frac{p}{\log(p)}\right)^{p+1}$, [2]
Non-atomic	$\left(e^{\frac{1}{e}}\right)^p$	$\left(1-p(p+1)^{-(p+1)/p}\right)^{-1} \sim \Theta\left(\frac{p}{\log(p)}\right)$, [84]
Online	4^p	$(2^{1/(p+1)}-1)^{-(p+1)} \sim \Theta(p)^{p+1}$, [31]

Φ_p denotes the unique solution of equation $x^{p+1} = (x+1)^p$, and $k := \lfloor \Phi_p \rfloor$. We observe that the performance under the NSW case is definitely better (even asymptotically) than that under the USW case, except for the non-atomic setting.

for the USW, not only from a quantitative point of view, but also from a qualitative one. In fact, while it is well known (see Reference [33]) that for the USW the simpler combinatorial structure of load-balancing games does not improve the price of anarchy of general congestion games, our Theorem A.1 (deferred to the Appendix) and Corollary 3.5 show that, for the NSW, even for the case of linear latency functions, the price of anarchy drops from n to 2.

All upper bounds shown in this article are quite general, given that they hold for any family of non-decreasing and positive latency functions. Moreover, the provided matching lower bounds hold for latency functions verifying mild assumptions; it is worth to remark that they are satisfied by the well-studied class of polynomial latency functions and by many other ones.

In particular, Theorem 3.1 provides an upper bound to the price of anarchy for the case of weighted load-balancing games, while Theorem 3.3 gives a matching lower bound that holds even for symmetric games under mild assumptions. Similarly, we focus on unweighted games (a special case of weighted ones) by providing tight bounds that, in general, are lower than the ones that can be obtained for weighted games (see Section 3.2). However, Corollaries 3.5 (or 3.6) and 3.9 show that, when considering polynomial latency functions of degree p, the two analyses (for weighted games and for unweighted ones) give the same tight bound of 2^p. Furthermore, when considering weighted games, the tight bound of 2^p holds even for symmetric games (Corollary 3.5) and for games with identical resources (Corollary 3.6).

We also provide a tight analysis holding for non-atomic games (see Section 3.3), and a tight lower-bound, under mild assumptions, is attained by a simple Pigou-like network [82] (as well as for the utilitarian social welfare [84]); for the case of polynomial latency functions of degree p, Corollary 3.12 shows that the price of anarchy is $(e^{\frac{1}{e}})^p \simeq (1.44)^p$.

For the online setting, we analyze the greedy algorithm that assigns every client to a resource minimizing the total cost of the instance revealed up to the time of its appearance. We provide a tight analysis of the competitive ratio of the greedy algorithm, and we show that, when considering polynomial latency functions of degree p, there exists no online algorithm achieving a competitive ratio better than the one of the greedy algorithm, which is equal to 4^p (see Section 4). In Table 1, we consider the case of polynomial latency functions, and we compare the performance under the NSW with that under the USW studied in some previous works.

1.4 Article Organization

The rest of the article is structured as follows: Section 2 introduces the model. Sections 3 and 4 are devoted to the performance analysis of the price of anarchy and of the competitive ratio, under the selfish and the online setting, respectively. Finally, in Section 5, we give some conclusive remarks and state some interesting open problems.

2 MODEL

Given $k \in \mathbb{N}$, let $[k] := \{1, 2, \ldots, k\}$. A class C of functions is called *ordinate-scaling* if, for any $f \in C$ and $\alpha \geq 0$, the function g such that $g(x) = \alpha f(x)$ for any $x \geq 0$, belongs to C; *abscissa-scaling* if, for any $f \in C$ and $\alpha \geq 0$, the function g such that $g(x) = f(\alpha x)$ for any $x \geq 0$, belongs to C; *all-constant-including* if it contains all the constant functions (i.e., all functions f such that $f(x) = c$ for some $c > 0$); *unbounded-including* if all the latency functions f, except for the constant ones, verify $\lim_{x\to\infty} f(x) = \infty$. Let $\mathcal{P}(p)$ denote the class of polynomial latencies of maximum degree p, i.e., the class of functions $f(x) = \sum_{d=0}^{p} \alpha_d x^d$, with $\alpha_d \geq 0$ for any $d \in [p] \cup \{0\}$ and $\alpha_d > 0$ for some $d \in [p] \cup \{0\}$. A function f is *quasi-log-convex* if $x \ln(f(x))$ is convex. We first deal with *selfish load balancing*, and then we turn our attention to the online setting.

2.1 Selfish Load Balancing

(Atomic) Load-balancing Games. A *weighted (atomic) load-balancing game*, or *load-balancing game* for brevity, is a tuple $\mathsf{LB} = (N, R, (\ell_j)_{j\in R}, (w_i)_{i\in N}, (\Sigma_i)_{i\in N})$, where N is a set of $n \geq 1$ players (corresponding to clients), R is a finite set of resources, $\ell_j : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ is the (non-decreasing and positive) latency function of resource $j \in R$, and, for each $i \in N$, $w_i \in \mathbb{R}_{>0}$ is the weight of player i and $\Sigma_i \subseteq R$ (with $\Sigma_i \neq \emptyset$) is her set of strategies (or admissible resources). As usual, we assume that each latency function ℓ verifies $\ell(0) = 0$.

An *unweighted load-balancing game* is a weighted load-balancing game with unitary weights. A *symmetric weighted load-balancing game* is a load-balancing game in which each player can select all the resources, i.e., $\Sigma_i = R$ for any $i \in N$.

Given a class C of latency functions, let $\mathsf{ULB}(C)$ be the class of unweighted load-balancing games, $\mathsf{WLB}(C)$ be the class of weighted load-balancing games, and $\mathsf{SWLB}(C)$ be the class of weighted symmetric load-balancing games, all having latency functions in the class C. We say that resources are *identical* if all of them have the same latency function.

Non-atomic Load balancing Games. The counterpart of the class of atomic load-balancing games is that of *non-atomic load-balancing games* [11, 82, 93]: These games are a good approximation for atomic ones when players become infinitely many and the contribution of each player to social welfare becomes infinitesimally small. A *non-atomic load-balancing game* is a tuple $\mathsf{NLB} = (N, R, (\ell_j)_{j\in R}, (r_i)_{i\in N}, (\Sigma_i)_{i\in N})$, where N is a set of $n \geq 1$ *types* of players, R is a finite set of resources, $\ell_j : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ is the (non-decreasing and positive) latency function of resource $j \in R$; moreover, given $i \in N$, $r_i \in \mathbb{R}_{\geq 0}$ is the amount of players of type i and $\Sigma_i \subseteq R$ is the set of strategies of every player of type i.

Given a class C of latency functions, let $\mathsf{NLB}(C)$ be the class of non-atomic load-balancing games, and $\mathsf{SNLB}(C)$ be the class of symmetric non-atomic load-balancing games, all having latency functions in the class C.

Strategy Profiles and Cost Functions. In atomic load-balancing games, a *strategy profile* is an n-tuple $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$, where $\sigma_i \in \Sigma_i$ is the resource chosen by player $i \in N$ in $\boldsymbol{\sigma}$. Given a strategy profile $\boldsymbol{\sigma}$, let $k_j(\boldsymbol{\sigma}) := \sum_{i\in N:\sigma_i=j} w_i$ be the *congestion* of resource $j \in R$ in $\boldsymbol{\sigma}$, and let $cost_i(\boldsymbol{\sigma}) := \ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}))$ be the *cost* of player $i \in N$ in $\boldsymbol{\sigma}$.

In non-atomic load-balancing games, a *strategy profile* is an n-tuple $\boldsymbol{\Delta} = (\Delta_1, \ldots, \Delta_n)$, where $\Delta_i : R \to \mathbb{R}_{\geq 0}$ is a function denoting, for each resource $j \in R$, the amount $\Delta_i(j)$ of players of type i selecting resource j, so $\sum_{j\in\Sigma_i} \Delta_i(j) = r_i$. Observe that $\Delta_i(j) = 0$ if $j \notin \Sigma_i$. For a strategy profile $\boldsymbol{\Delta}$, the congestion of resource $j \in R$ in $\boldsymbol{\Delta}$, denoted as $k_j(\boldsymbol{\Delta}) := \sum_{i\in N} \Delta_i(j)$, is the total amount of players using resource j in $\boldsymbol{\Delta}$ and its cost is given by $cost_j(\boldsymbol{\Delta}) = \ell_j(k_j(\boldsymbol{\Delta}))$. The cost of a player of type i selecting a resource $j \in \Sigma_i$ is equal to $cost_j(\boldsymbol{\Delta})$ and each player aims at minimizing it.

We stress that, in both types of games, the congestion of a resource is a non-negative real number, except for the case of unweighted atomic games, where it takes integer values.

Nash Social Welfare. In atomic load-balancing games, the ***Nash Social Welfare (NSW)*** of a strategy profile $\boldsymbol{\sigma}$ is defined as:

$$\mathsf{NSW}(\boldsymbol{\sigma}) := \left(\prod_{i\in N} cost_i(\boldsymbol{\sigma})^{w_i}\right)^{\frac{1}{\sum_{i\in N} w_i}}.$$

Using the previous definition, for unweighted games, we get $\mathsf{NSW}(\boldsymbol{\sigma}) = (\prod_{i\in N} cost_i(\boldsymbol{\sigma}))^{\frac{1}{n}}$. Given a strategy profile $\boldsymbol{\sigma}$, let $R(\boldsymbol{\sigma}) := \{j \in R : k_j(\boldsymbol{\sigma}) > 0\}$. For weighted load-balancing games, we get:

$$\mathsf{NSW}(\boldsymbol{\sigma}) = \left(\prod_{i\in N} cost_i(\boldsymbol{\sigma})^{w_i}\right)^{\frac{1}{\sum_{i\in N} w_i}} = \left(\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j(\boldsymbol{\sigma}))^{k_j(\boldsymbol{\sigma})}\right)^{\frac{1}{\sum_{j\in R(\boldsymbol{\sigma})} k_j(\boldsymbol{\sigma})}}.$$

Let $\mathsf{SP}(\mathsf{LB})$ be the set of strategy profiles of an atomic load-balancing game LB. An optimal strategy profile $\boldsymbol{\sigma}^*(\mathsf{LB})$ of a load-balancing game LB is a strategy profile $\boldsymbol{\sigma}^* \in \arg\min_{\boldsymbol{\sigma}\in\mathsf{SP}(\mathsf{LB})} \mathsf{NSW}(\boldsymbol{\sigma})$, i.e., a strategy profile minimizing the NSW. Analogously, for the non-atomic setting, we have

$$\mathsf{NSW}(\Delta) = \left(\prod_{j\in R(\Delta)} cost_j(\Delta)^{k_j(\Delta)}\right)^{\frac{1}{\sum_{j\in R(\Delta)} k_j(\Delta)}},$$

where $R(\Delta) := \{j \in R : k_j(\Delta) > 0\}$. Let $\mathsf{SP}(\mathsf{NLB})$ be the set of strategy profiles of a non-atomic load-balancing game NLB. An optimal strategy profile $\Delta^*(\mathsf{NLB})$ of a load-balancing game NLB is a strategy profile $\Delta^* \in \arg\min_{\Delta\in\mathsf{SP}(\mathsf{NLB})} \mathsf{NSW}(\Delta)$, i.e., a strategy profile minimizing the NSW.[3]

Pure Nash Equilibria and Their Efficiency. In the atomic setting, for a given strategy profile $\boldsymbol{\sigma}$, let $(\boldsymbol{\sigma}_{-i}, \sigma_i') := (\sigma_1, \sigma_2, \ldots, \sigma_{i-1}, \sigma_i', \sigma_{i+1}, \ldots, \sigma_n)$, i.e., a strategy profile equal to $\boldsymbol{\sigma}$, except for strategy σ_i'. A *pure Nash equilibrium* is a strategy profile $\boldsymbol{\sigma}$ such that $cost_i(\boldsymbol{\sigma}) \le cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i')$ for any $\sigma_i' \in \Sigma_i$ and $i \in N$, i.e., a strategy profile in which no player can improve her cost by unilateral deviations.[4] Let $\mathsf{PNE}(\mathsf{LB})$ be the set of pure Nash equilibria of a load-balancing game LB. The *Nash price of anarchy* of LB is defined as:

$$\mathsf{NPoA}(\mathsf{LB}) = \sup_{\boldsymbol{\sigma}\in\mathsf{PNE}(\mathsf{LB})} \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*(\mathsf{LB}))}.$$

Given a class $\mathcal{G}$ of load-balancing games, the *Nash price of anarchy* of $\mathcal{G}$ is defined as $\mathsf{NPoA}(\mathcal{G}) = \sup_{\mathsf{LB}\in\mathcal{G}} \mathsf{NPoA}(\mathsf{LB})$.

[3]We observe that, for any non-atomic game, there always exists a strategy profile minimizing the Nash social welfare. To show this, we first recall that the set of pure strategy profiles can be represented as the set of vectors $\Delta := (\Delta_i(j))_{i\in N, j\in\Sigma_i}$ satisfying the linear constraints $\sum_{j'\in\Sigma_i} \Delta_i(j') = r_i$ and $\Delta_i(j) \ge 0$, for any $i \in N$ and $j \in \Sigma_i$. This implies that the set of strategy profiles can be represented as a compact subset of the Euclidean space. Furthermore, we have that a generic function of type $g(x) := \ell(x)^x = e^{x\ln(\ell(x))}$ is continuous in $x \ge 0$ if the function ℓ is non-negative and non-decreasing; thus, since the Nash social welfare $\mathsf{NSW}(\Delta)$ can be seen as continuous composition of several functions of type $g(x) := \ell(x)^x$, we have that $\mathsf{NSW}(\Delta)$ is a continuous function in the variables $\Delta_i(j)$'s. We conclude that the problem of minimizing the Nash social welfare is equivalent to that of minimizing a continuous function over a compact set, and by the Weierstrass Theorem, such a problem always admits a minimum.

[4]As shown in Reference [52], a pure Nash equilibrium always exists in weighted load-balancing games. Instead, when moving to the more general class of weighted congestion games, pure Nash equilibria may not exist (see Reference [60]), but they continue to exist in the subclass of unweighted congestion games (see Reference [83]).

In the non-atomic setting, a *pure Nash equilibrium* is a strategy profile $\boldsymbol{\Delta}$ such that, for any player type $i \in N$, resources $j, j' \in \Sigma_i$ such that $\Delta_i(j) > 0$, $cost_j(\boldsymbol{\Delta}) \leq cost_{j'}(\boldsymbol{\Delta})$ holds, that is, an outcome of the game in which no player can improve her situation by unilaterally deviating to another strategy.[5] The *Nash price of anarchy* of a non-atomic game NLB (denoted as NPoA(NLB)) is defined as in the atomic setting, and again, given a class $\mathcal{G}$ of non-atomic load-balancing games, the *Nash price of anarchy* of $\mathcal{G}$ is defined as $\mathsf{NPoA}(\mathcal{G}) = \sup_{\mathsf{NLB} \in \mathcal{G}} \mathsf{NPoA}(\mathsf{NLB})$.

2.2 Online Load Balancing

We now introduce online load balancing. There is a natural correspondence between a load-balancing game and an instance of the online load-balancing problem. When dealing with the online setting, as usual in the literature, we adopt a different nomenclature. In particular, an instance I of the online load-balancing problem is a tuple $\mathsf{I} = (N, R, (\ell_j)_{j \in R}, (w_i)_{i \in N}, (\Sigma_i)_{i \in N})$, where $N = [n]$ is a set of $n \geq 1$ *clients*, R is a finite set of resources, $\ell_j : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{>0}$ is the (non-decreasing and positive) latency function of resource $j \in R$, and, for each $i \in N$, $w_i > 0$ is the weight of client i and $\Sigma_i \subseteq R$ (with $\Sigma_i \neq \emptyset$) is her set of *admissible resources*. Furthermore, in the online setting an assignment of clients to resources is called state: A *state* is an n-tuple $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$, where $\sigma_i \in \Sigma_i \subseteq R$ is the resource assigned to player $i \in N$ in $\boldsymbol{\sigma}$. As in load-balancing games, given a class of latency latency functions C, let $\mathsf{WLB}(C)$ denote class of load-balancing instances with latency functions in C.

The NSW of a state and the optimal state are defined analogously to the selfish load-balancing setting.

The online setting. In *online load balancing*, clients appear in online fashion, in consecutive *steps*; when a client appears, an irrevocable decision has to be taken to assign it to a resource. We assume w.l.o.g. that clients appear in increasing order, i.e., client $i \in [n]$ appears before client $j \in [n]$ if and only if $i < j$. More formally, for any $i \in [n]$, an online algorithm has to assign client i to a resource being admissible for it without the knowledge of the future clients $i + 1, i + 2, \ldots$; the assignment of client i decided by the algorithm at step i cannot be modified at later steps.

Notice that at each step $i > 1$ a new instance is obtained by adding client i to the instance of step $i - 1$.

Competitive Ratio. Following the standard performance measure in competitive analysis, we evaluate the performance of an online algorithm in terms of its *competitiveness* (or *competitive ratio*).

An online algorithm A is *c-competitive* on instance I if the following holds: Let σ and σ^* be the state computed by algorithm A and the optimal state for I, respectively. Then, $\mathsf{NSW}(\sigma) \leq c \cdot \mathsf{NSW}(\sigma^*)$. The competitive ratio $\mathsf{CR}_\mathsf{A}(\mathsf{I})$ of algorithm A on instance I is the smallest c such that A is c-competitive on I [29].

Given a class $\mathcal{I}$ of load-balancing instances, the competitive ratio $\mathsf{CR}_\mathsf{A}(\mathcal{I})$ of Algorithm A on $\mathcal{I}$ is simply given by the supremum competitive ratio of A over all instances $\mathsf{I} \in \mathcal{I}$, i.e., $\mathsf{CR}_\mathsf{A}(\mathcal{I}) = \sup_{\mathsf{I} \in \mathcal{I}} \mathsf{CR}_\mathsf{A}(\mathsf{I})$.

Greedy algorithm. A natural algorithm proposed in Reference [8] for this problem is to assign each client to the resource yielding the minimum increase to the social welfare (ties are broken arbitrarily). This results to *greedy assignments*. Therefore, given an instance of online load balancing, an assignment of clients to resources is called a greedy assignment if the assignment of a client to a resource minimizes the total cost of the instance revealed up to the time of its appearance.

[5] As shown in Reference [11], a pure Nash equilibrium always exists in non-atomic load-balancing games and continues to exist in the more general class of non-atomic congestion games.

3 SELFISH LOAD BALANCING

In this section, we focus on selfish load balancing. In particular, in Section 3.1, we deal with the analysis of the price of anarchy in weighted load-balancing games; in Section 3.2, we consider the subclass of unweighted load-balancing games; while in Section 3.3, we analyze the price of anarchy of non-atomic load-balancing games.

3.1 The NPoA for Weighted Load-balancing Games

We first provide an upper bound to the Nash price of anarchy of weighted load-balancing games. Given a class of latency function C, define

$$\psi(C) := \sup_{\substack{f_1,f_2\in C,\\ k_1,k_2,o_1,o_2\in\mathbb{R}:k_1\geq o_1>0,\, o_2>k_2\geq 0}} \left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}} \left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}}.$$

Theorem 3.1 (Upper Bound). *Let C be a class of latency functions. The Nash price of anarchy of weighted load-balancing games with latency functions in C is* $\mathsf{NPoA}(\mathsf{WLB}(C)) \leq \psi(C)$.

Proof. Let $\mathsf{LB} \in \mathsf{WLB}(C)$ be a weighted load-balancing game with latency functions in C, and let $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ be a worst-case pure Nash equilibrium and an optimal strategy profile of LB, respectively. Let k_j denote $k_j(\boldsymbol{\sigma})$ and o_j denote $k_j(\boldsymbol{\sigma}^*)$. Since $\boldsymbol{\sigma}$ is a pure Nash equilibrium, we have that $cost_i(\boldsymbol{\sigma}) \leq cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*)$. Thus, we get $\prod_{i\in N} cost_i(\boldsymbol{\sigma})^{w_i} \leq \prod_{i\in N} cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*)^{w_i}$.

Since $cost_i(\boldsymbol{\sigma}) = \ell_{\sigma_i}(k_{\sigma_i})$ and $cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*) \leq \ell_{\sigma_i^*}(k_{\sigma_i^*} + w_i)$, it holds that

$$\prod_{i\in N} cost_i(\boldsymbol{\sigma})^{w_i} = \prod_{i\in N} \ell_{\sigma_i}(k_{\sigma_i})^{w_i} = \prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{\sum_{i:j=\sigma_i} w_i} = \prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j}$$

and

$$\begin{aligned}\prod_{i\in N} cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*)^{w_i} &\leq \prod_{i\in N} \ell_{\sigma_i^*}(k_{\sigma_i^*} + w_i)^{w_i} \leq \prod_{i\in N} \ell_{\sigma_i^*}(k_{\sigma_i^*} + o_{\sigma_i^*})^{w_i}\\ &= \prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+o_j)^{\sum_{i:j=\sigma_i^*} w_i} = \prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+o_j)^{o_j}.\end{aligned}$$

By putting together the above inequalities, we get

$$\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j} = \prod_{i\in N} cost_i(\boldsymbol{\sigma})^{w_i} \leq \prod_{i\in N} cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*)^{w_i} \leq \prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+o_j)^{o_j}. \tag{1}$$

By exploiting the properties of the logarithmic function and by using (1), we obtain

$$\begin{aligned}\ln(\mathsf{NPoA}(\mathsf{LB})) &= \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}\right)\\ &\leq \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+o_j)^{o_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}\right)\\ &= \frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} o_j\left(\ln(\ell_j(k_j+o_j)) - \ln(\ell_j(o_j))\right)}{\sum_{i\in N} w_i}.\end{aligned} \tag{2}$$

Since $\sum_{i \in N} w_i = \sum_{j \in R} k_j = \sum_{j \in R} o_j$, we have that (2) is upper bounded by the optimal solution of the following optimization problem OP on some linear variables $(\alpha_j)_{j \in R}$:

$$\text{OP}: \max_{(\alpha_j)_{j \in R}} \quad \frac{\sum_{j \in R(\sigma^*)} \alpha_j o_j \left(\ln(\ell_j(k_j + o_j)) - \ln(\ell_j(o_j))\right)}{\sum_{j \in R} \alpha_j k_j} \tag{3}$$
$$\text{s.t.} \quad \sum_{j \in R} \alpha_j k_j = \sum_{j \in R} \alpha_j o_j, \quad \alpha_j \geq 0 \ \forall j \in R.$$

Indeed, (2) is the solution of OP obtained by setting $\alpha_j = 1$ for each $j \in R$. We have the following lemma:

Lemma 3.2. *The optimal value of* OP *is at most*

$$\sup_{\substack{k_1 \geq o_1 > 0, \\ o_2 > k_2 \geq 0, \\ f_1, f_2 \in C}} \frac{(o_2 - k_2) o_1 \left(\ln(f_1(k_1 + o_1)) - \ln(f_1(o_1))\right) + (k_1 - o_1) o_2 \left(\ln(f_2(k_2 + o_2)) - \ln(f_2(o_2))\right)}{k_1 o_2 - k_2 o_1}.$$

Proof of Lemma 3.2. First, by exploiting the structure of OP, we can introduce the normalization constraint $\sum_{j \in R} \alpha_j k_j = \sum_{j \in R} \alpha_j o_j = 1$ without affecting the optimal value of OP. By introducing such normalization constraint, OP becomes the following linear program LP:

$$\text{LP}: \max_{(\alpha_j)_{j \in R}} \quad \sum_{j \in R(\sigma^*)} \alpha_j o_j \left(\ln(\ell_j(k_j + o_j)) - \ln(\ell_j(o_j))\right) \tag{4}$$
$$\text{s.t.} \quad \sum_{j \in R} \alpha_j k_j = 1, \quad \sum_{j \in R} \alpha_j o_j = 1, \quad \alpha_j \geq 0 \ \forall j \in R.$$

By standard arguments of linear programming, we have that an optimal solution of LP is given by a vertex of the polyhedral region defined by the linear constraints of LP, and such vertex can be obtained by nullifying at least $|R| - 2$ variables. Thus, we can assume w.l.o.g. that in an optimal solution of LP there are at most two variables, say, α_1 and α_2, such that $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$. If both variables α_1 and α_2 are (strictly) positive, then we have that they are univocally determined by the constraints $\alpha_1 k_1 + \alpha_2 k_2 = 1$ and $\alpha_1 o_1 + \alpha_2 o_2 = 1$, so

$$\alpha_1 = \frac{o_2 - k_2}{k_1 o_2 - k_2 o_1}, \quad \alpha_2 = \frac{k_1 - o_1}{k_1 o_2 - k_2 o_1}, \quad \alpha_j = 0 \ \forall j \geq 3. \tag{5}$$

By symmetry, we can assume w.l.o.g. that $k_1 o_2 - k_2 o_1 > 0$, so $k_1 > o_1 \geq 0$ and $o_2 > k_2 \geq 0$.

Now, assume that one variable among α_1 and α_2 is null, and assume w.l.o.g. that $\alpha_2 = 0$. In this case, we necessarily get $k_1 = o_1 > 0$ and $\alpha_1 = 1/o_1$, and the value of the objective function of LP becomes $\ln(f_1(2o_1)) - \ln(f_1(o_1))$. Anyway, we obtain the same value of the objective function by using in (4) the values of α_1 and α_2 considered in (5), and by setting $k_1 = o_1 > 0$ and $o_2 > k_2 \geq 0$. We also observe that, if $o_1 = 0$ and $\alpha_1, \alpha_2 > 0$, then the value of the objective function is $\ln(f_2(k_2 + o_2)) - \ln(f_2(o_2)) \leq \ln(f_2(2o_2)) - \ln(f_2(o_2))$, i.e., at most equal to the value of the objective function in which one of the two variables among α_1 and α_2 is null. Thus, we may omit the case $o_1 = 0$.

We conclude that, by considering the objective function of LP with the values α_1 and α_2 defined in (5), and by considering the supremum of the objective function over $k_1 \geq o_1 > 0$ and $o_2 > k_2 \geq 0$, we obtain the upper bound of the claim. □

By Lemma 3.2, and by continuing from (2), we have that the upper bound provided in Lemma 3.2 is higher or equal than $\ln(\text{NPoA}(\text{LB}))$. Thus, by exponentiating this inequality, we get $\text{NPoA}(\text{LB}) \leq \psi(C)$. Hence, by the arbitrariness of $\text{LB} \in \text{WLB}(C)$, the claim follows. □

In the following theorem, we show that the upper bound derived in Theorem 3.1 is tight under mild assumptions on the latency functions.

Theorem 3.3 (Lower Bound). *Let $\mathcal{C}$ be a class of latency functions. (i) If $\mathcal{C}$ is abscissa-scaling and ordinate-scaling, then* $\mathsf{NPoA}(\mathsf{WLB}(\mathcal{C})) \geq \psi(\mathcal{C})$. *(ii) If $\mathcal{C}$ is abscissa-scaling, ordinate-scaling, and unbounded-including, then the previous inequality holds even for symmetric weighted load-balancing games.*

Sketch of the proof. Here, we only describe the lower bounding instance used to show part (i) of the claim. The Appendix contains the full proof of part (ii), in which we analyze a more complex lower bounding instance than that of part (i); furthermore, the full proof of part (i) is presented in the Appendix as particular case of part (ii).

In the following, we will consider a lower bounding instance $\mathsf{LB}(m,k,h,f,g)$ parametrized by an integer $m \geq 4$, two real numbers $k \geq 1$ and $h \in (0,1)$, and two latency functions $f,g \in \mathcal{C}$. Then, we will show that, for any fixed $M < \psi(\mathcal{C})$, we can opportunely choose the above parameters in such a way that the Nash price of anarchy of $\mathsf{LB}(m,k,h,f,g)$ is at least M, thus showing the claim. The considered lower bound instance is similar to those used in References [23, 33] to provide lower bounds on the price of anarchy w.r.t. the USW.

Given an integer $m \geq 4$, two real numbers $k \geq 1$ and $h \in (0,1)$, and two latency functions $f,g \in \mathcal{C}$, let $\mathsf{LB}(m) := \mathsf{LB}(m,k,h,f,g)$ be a weighted load-balancing game with $2m$ resources $r_1,\ldots,r_{2m}$, where the latency function ℓ_{r_j} of resource r_j is defined as $\ell_{r_j}(x) := \alpha_j \hat{f}_j\left(\beta_j x\right)$ with:

$$\hat{f}_j := \begin{cases} f & \text{if } j \leq m-1 \\ g & \text{if } j \geq m \end{cases}, \quad \beta_j := \begin{cases} \left(\frac{1}{k}\right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{1}{h}\right)^{j-m}\left(\frac{1}{k}\right)^{m-1} & \text{if } m \leq j \leq 2m \end{cases}, \tag{6}$$

$$\alpha_j := \begin{cases} \left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{g(h)}{g(h+1)}\right)^{j-m}\left(\frac{f(k)}{g(h+1)}\right)\left(\frac{f(k)}{f(k+1)}\right)^{m-2} & \text{if } m \leq j \leq 2m-1 \\ \frac{g(h)}{g(1)}\left(\frac{g(h)}{g(h+1)}\right)^{m-1}\left(\frac{f(k)}{g(h+1)}\right)\left(\frac{f(k)}{f(k+1)}\right)^{m-2} & \text{if } j = 2m \end{cases}. \tag{7}$$

We have $2m-1$ players, and each player $j \in [2m-1]$ has weight $w_j := 1/\beta_{j+1}$ and can select resource r_j or r_{j+1} only. We observe that all the latency functions of $\mathsf{LB}(m)$ belong to $\mathcal{C}$, as $\mathcal{C}$ is abscissa-scaling and ordinate-scaling.

Let $\boldsymbol{\sigma}$ be the strategy profile in which each player $j \in [2m-1]$ selects resource r_j. Observe that, by construction of α_j, β_j, w_j, the following properties hold:

$$\begin{cases} \alpha_j f(k) = \alpha_{j+1} f(k+1) & \text{if } j \leq m-2 \\ \alpha_j f(k) = \alpha_{j+1} g(h+1) & \text{if } j = m-1 \\ \alpha_j g(h) = \alpha_{j+1} g(h+1) & \text{if } m \leq j \leq 2m-2 \\ \alpha_j g(h) = \alpha_{j+1} g(1) & \text{if } j = 2m-1 \end{cases},$$

$$\begin{cases} \beta_j w_j = k,\ w_j = k^j & \text{if } j \leq m-1 \\ \beta_j w_j = h,\ w_j = h^{j+1-m}k^{m-1} & \text{if } m \leq j \leq 2m-1 \\ \beta_{j+1} w_j = 1 & \text{if } j \leq 2m-1 \end{cases}. \tag{8}$$

By exploiting (8), one can easily show that $\boldsymbol{\sigma}$ is a pure Nash equilibrium. To this aim, we fix an arbitrary player j, and we show that her cost does not change when she deviates from resource r_j to resource r_{j+1} in $\boldsymbol{\sigma}$. If $j \leq m-2$, then by using (8), we get $cost_i(\boldsymbol{\sigma}) = \ell_{r_j}(k_{r_j}(\boldsymbol{\sigma})) = \alpha_j \hat{f}_j(\beta_j w_j) = \alpha_j f(k) = \alpha_{j+1} f(k+1) = \alpha_{j+1} f(\beta_{j+1} w_{j+1} + \beta_{j+1} w_j) = \alpha_{j+1} \hat{f}_{j+1} (\beta_{j+1}(w_{j+1}+w_j)) = \ell_{r_h}(k_{r_h}(\boldsymbol{\sigma}_{-i}, \{r_{j+1}\}))$. The cases $j = m-1$, $m \leq j \leq 2m-2$, and $j = 2m-1$ can be separately considered by exploiting (8), so one can analogously show that $cost_i(\boldsymbol{\sigma}) = \alpha_j \hat{f}_j(\beta_j w_j) = \alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}(w_{j+1}+w_j)) = cost_i(\boldsymbol{\sigma}_{-i},\{r_{j+1}\})$, where we set $w_{2m} := 0$.

Let $\boldsymbol{\sigma}^*$ be the strategy profile in which each player j selects resource r_{j+1}. By standard calculations, one can show that, for any $\epsilon > 0$, there exists a sufficiently large m such that the following inequalities hold:

$$
\begin{aligned}
&\mathsf{NPoA}(\mathsf{LB}(m)) && (9)\\
\geq &\frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)} && (10)\\
= &\left(\frac{\prod_{j=1}^{2m-1}\left(\alpha_j \hat{f}_j\left(\beta_j w_j\right)\right)^{w_j}}{\prod_{j=2}^{2m}\left(\alpha_j \hat{f}_j\left(\beta_j w_{j-1}\right)\right)^{w_{j-1}}}\right)^{\frac{1}{\sum_{j=1}^{2m-1} w_j}}\\
= &\left(\left(\frac{f(k+1)}{f(1)}\right)^{\sum_{j=1}^{m-2} k^j}\left(\frac{g(h+1)}{g(1)}\right)^{\sum_{j=m-1}^{2m-2} h^{j+1-m}k^{m-1}}\right)^{\frac{1}{\sum_{j=1}^{m-2}k^j+\sum_{j=m-1}^{2m-1}h^{j+1-m}k^{m-1}}}\\
\geq &\lim_{m\to\infty}\left(\left(\frac{f(k+1)}{f(1)}\right)^{\sum_{j=1}^{m-2} k^j}\left(\frac{g(h+1)}{g(1)}\right)^{\sum_{j=m-1}^{2m-2} h^{j+1-m}k^{m-1}}\right)^{\frac{1}{\sum_{j=1}^{m-2}k^j+\sum_{j=m-1}^{2m-1}h^{j+1-m}k^{m-1}}} - \epsilon && (11)\\
= &\left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}}\left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}} - \epsilon, && (12)
\end{aligned}
$$

where (10) holds, since $\boldsymbol{\sigma}$ is a pure Nash equilibrium, and (11) holds by choosing a sufficiently large m. Thus, (12) provides a parametric lower bound on the Nash price of anarchy.

Now, we fix an arbitrary $M < \psi(\mathcal{C})$, and we choose the parameters m, k, h, f, g in such a way that (12) is at least equal to M. By definition of $\psi(\mathcal{C})$, we have that there exist $f_1, f_2 \in \mathcal{C}$ and $k_1, k_2, o_1, o_2 \geq 0$ such that $k_1 \geq o_1 > 0, o_2 > k_2 \geq 0$, and a sufficiently small $\epsilon > 0$ such that

$$\left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}}\left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}} > M+\epsilon.$$

Let f, g be two latency functions such that $f(x) := f_1(o_1 x)$ and $g(x) := f_2(o_2 x)$, and let $k := k_1/o_1$ and $h := k_2/o_2$; we observe that f, g belong to $\mathcal{C}$, as $\mathcal{C}$ is abscissa-scaling and ordinate-scaling. Since

$$\left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}}\left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}} = \left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}}\left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}},$$

we have that

$$\left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}}\left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}} > M+\epsilon. \tag{13}$$

We first assume that $h > 0$. In such case, as $k \geq 1$ and $h \in (0,1)$, we can use inequality (12) with the considered parameters k, h, f, g. Thus, by putting (12) and (13) together, we have that the above lower bounding instance $\mathsf{LB}(m)$, for a sufficiently large m, guarantees a Nash price of anarchy of at least M, and this shows the claim of part (i).

If $h = 0$, we can consider a load-balancing game defined as $\mathsf{LB}(m)$, but restricted to the first m resources and to the first $m-1$ players. By using the same proof arguments as those used for $h > 0$, one can show the claim as well. □

When considering functions belonging to the class $\mathcal{P}(p)$ of polynomials of maximum degree p, the following technical lemma holds:

Lemma 3.4. $\psi(\mathcal{P}(p)) = 2^p$.

Proof. We have that

$$\psi(\mathcal{P}(p)) = \sup_{k_1 \geq o_1 > 0, o_2 > k_2 \geq 0, f_1, f_2 \in \mathcal{P}(p)} \left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}} \left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}}$$

$$= \sup_{\substack{k_1 \geq o_1 > 0,\\ o_2 > k_2 \geq 0,\\ \alpha_0,\dots,\alpha_p, \geq 0\\ \beta_0,\dots,\beta_p \geq 0}} \left(\frac{\sum_{d=0}^p \alpha_d (k_1+o_1)^d}{\sum_{d=0}^p \alpha_d o_1^d}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}} \left(\frac{\sum_{d=0}^p \beta_d (k_2+o_2)^d}{\sum_{d=0}^p \beta_d o_2^d}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}}$$

$$= \sup_{\substack{k_1 \geq o_1 > 0,\\ o_2 > k_2 \geq 0}} \left(\max_{d \in [p] \cup \{0\}} \frac{(k_1+o_1)^d}{o_1^d}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}} \left(\max_{d \in [p] \cup \{0\}} \frac{(k_2+o_2)^d}{o_2^d}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}}$$

$$= \sup_{k_1 \geq o_1 > 0, o_2 > k_2 \geq 0} \left(\left(\frac{k_1+o_1}{o_1}\right)^p\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}} \left(\left(\frac{k_2+o_2}{o_2}\right)^p\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}}$$

$$= \sup_{k \geq 1, 0 \leq h < 1} \left((k+1)^{\frac{1-h}{k-h}} (h+1)^{\frac{k-1}{k-h}}\right)^p, \tag{14}$$

where (14) can be obtained by setting the real values $k := k_1/o_1$ and $h := k_2/o_2$. Now, we show that the maximum value of function $F(k,h) := (k+1)^{\frac{1-h}{k-h}} (h+1)^{\frac{k-1}{k-h}}$ over $k \geq 1$ and $0 \leq h < 1$ is equal to 2. Observe that $\ln(F(k,h)) = \frac{1-h}{k-h} \ln(k+1) + \frac{k-1}{k-h} \ln(h+1) \leq \ln(\frac{1-h}{k-h}(k+1) + \frac{k-1}{k-h}(h+1))$, where the last inequality holds, since $\ln(F(k,h))$ is defined as convex combination of $\ln(k+1)$ and $\ln(h+1)$, and because of the concavity of the natural logarithm. Thus, we get

$$F(k,h) \leq \frac{1-h}{k-h}(k+1) + \frac{k-1}{k-h}(h+1) = \frac{(k-h)+(k-h)}{k-h} = 2. \tag{15}$$

Finally, since $F(k,h) = 2$ for $k = 1$ and $h = 0$, and because of (15), we have that the maximum of $F(k,h)$ over $k \geq 1$ and $0 \leq h < 1$ is 2. Thus, we get that (14) is at most 2^p. □

Given Lemma 3.4, and, since the class of polynomial latency functions is ordinate-scaling, abscissa-scaling, and unbounded-including, the following corollary of Theorems 3.1 and 3.3 establishes the exact Nash price of anarchy for polynomial latency functions:

Corollary 3.5 (Polynomial Latencies). *The Nash price of anarchy of weighted load-balancing games with polynomial latency functions (even for symmetric games) of maximum degree p is* $\mathsf{NPoA}(\mathsf{WLB}(\mathcal{P}(p))) = \mathsf{NPoA}(\mathsf{SWLB}(\mathcal{P}(p))) = \psi(\mathcal{P}(p)) = 2^p$.

When considering games with identical resources and polynomial latency functions, the price of anarchy does not decrease, as shown in the following corollary of Theorem 3.3, whose proof is deferred to the Appendix.

Corollary 3.6 (Polynomial Latencies + Identical Resources). *The Nash price of anarchy of weighted load-balancing games with polynomial latency functions of maximum degree p and identical resources is at least* 2^p.

3.2 The NPoA for Unweighted Load-balancing Games

We first provide an upper bound to the Nash price of anarchy of unweighted load-balancing games. Given a class of latency function C, define

$$\xi(C) := \sup_{f\in C, k\in\mathbb{N}, o\in[k]} \left(\frac{f(k+1)}{f(o)}\right)^{\frac{o}{k}}.$$

Theorem 3.7 (Upper Bound). *Let C be a class of latency functions. The Nash price of anarchy of unweighted load-balancing games with latency functions in C is* $\mathsf{NPoA}(\mathsf{ULB}(C)) \leq \xi(C)$.

Proof. Let $\mathsf{LB} \in \mathsf{ULB}(C)$ be an unweighted load-balancing game with latency functions in C, and let $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ be a worst-case pure Nash equilibrium and an optimal strategy profile of LB, respectively. Let k_j denote $k_j(\boldsymbol{\sigma})$ and o_j denote $k_j(\boldsymbol{\sigma}^*)$. As in Theorem 3.1, we get

$$\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j} \leq \prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+1)^{o_j}. \tag{16}$$

By exploiting the properties of the logarithmic function, we get

$$\begin{aligned}
\ln(\mathsf{NPoA}(\mathsf{LB})) &= \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j}\right)^{\frac{1}{n}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{n}}}\right)\\
&\leq \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(k_j+1)^{o_j}\right)^{\frac{1}{n}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{n}}}\right) \qquad (17)\\
&= \frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j)))}{\sum_{j\in R} k_j},
\end{aligned}$$

where (17) comes from (16). Now, let $R_+ := \{j \in R(\boldsymbol{\sigma}^*) : k_j \geq o_j\}$. We have that

$$\begin{aligned}
&\frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j)))}{\sum_{j\in R} k_j}\\
\leq &\frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j)))}{\sum_{j\in R(\boldsymbol{\sigma}^*)} k_j}\\
\leq &\frac{\sum_{j\in R_+} o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j)))}{\sum_{j\in R_+} k_j} \qquad (18)\\
\leq &\max_{j\in R_+} \frac{o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j)))}{k_j}\\
\leq &\sup_{f\in C, k\in\mathbb{N}, o\in[k]} \frac{o(\ln(f(k+1)) - \ln(f(o)))}{k},
\end{aligned}$$

where (18) holds, because for any $j \in R(\boldsymbol{\sigma}^*) \setminus R_+$, it holds that $o_j(\ln(\ell_j(k_j+1)) - \ln(\ell_j(o_j))) \leq 0$. Therefore, we conclude that

$$\ln(\mathsf{NPoA}(\mathsf{LB})) \leq \sup_{f\in C, k\in\mathbb{N}, o\in[k]} \frac{o(\ln(f(k+1)) - \ln(f(o)))}{k},$$

and by exponentiating the previous inequality, we get the claim. □

In the following theorem, we show that the upper bound derived in Theorem 3.7 is tight under mild assumptions on the considered latency functions:

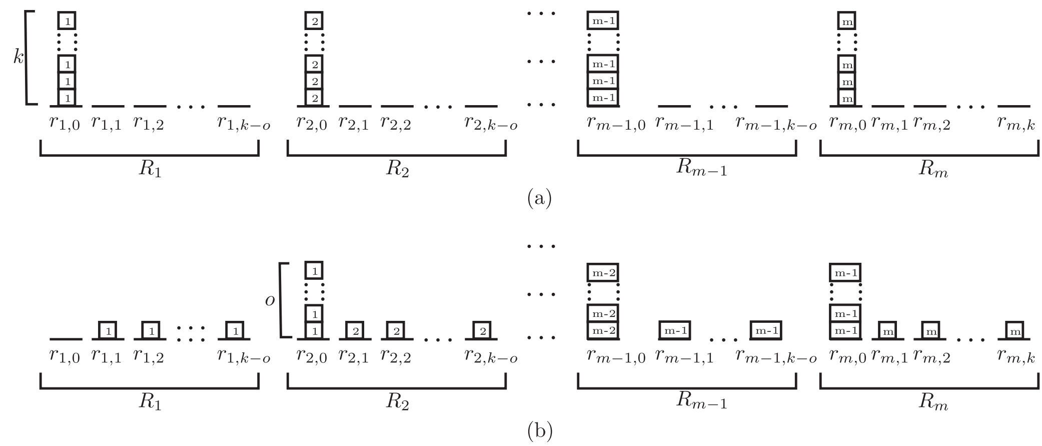

Fig. 1. The LB used in the proof of Theorem 3.8. Columns represent resources and squares represent players (number j inside a square means that the player belongs to group N_j). (a): The Nash equilibrium $\boldsymbol{\sigma}$; (b): The strategy profile $\boldsymbol{\sigma}^*$.

Theorem 3.8 (Lower Bound). *Let C be a class of latency functions. If C is ordinate-scaling, then* $\mathsf{NPoA}(\mathsf{ULB}(C)) \geq \xi(C)$.

Sketch of the proof. Here, we only provide the structure of the lower bounding instance and the main steps on how to show the claim (the full proof of the theorem is deferred to the Appendix). Similarly as in Theorem 3.3, the lower bounding instance we consider is parametrized by some integers and a latency function of C, and we will show that an opportune choice of such parameters guarantees a Nash price of anarchy arbitrarily close to $\xi(C)$. Given an integer $m > 0$, two integers k, o with $k \geq o > 0$, and a latency function $f \in C$, let $\mathsf{LB}(m) := \mathsf{LB}(m, k, o, f)$ be an unweighted load-balancing game with $(k-o+1)m+o$ resources, partitioned into m groups $R_1, R_2, \ldots, R_m$ such that $R_j := \{r_{j,0}, r_{j,1}, \ldots, r_{j,k-o}\}$ for any $j \in [m-1]$, and $R_m := \{r_{m,0}, r_{m,1}, \ldots, r_{m,k}\}$. Each resource $r_{j,h}$ has latency function $\ell_{r_{j,h}}(x) := \alpha_{j,h} f(x)$, with

$$\alpha_{j,h} := \begin{cases} \left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{if } h = 0 \\ \frac{f(k)}{f(1)} \left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{otherwise.} \end{cases}$$

We have $n := mk$ players split into m groups $N_1, N_2, \ldots, N_m$ of k players each. For $j \in [m-1]$, the set of strategies Σ_j of players of group N_j is $R_j \cup \{r_{j+1,0}\}$, and the set of strategies Σ_m of players in N_m is R_m. We observe that all the latency functions of $\mathsf{LB}(m)$ belong to C, as C is ordinate-scaling.

Let $\boldsymbol{\sigma}$ be the strategy profile such that, for any $j \in [m]$, all k players of group N_j select resource $r_{j,0}$, so each resource $r_{j,0}$ has congestion k, and all the remaining resources have null congestion (see Figure 1(a)). Similarly to Theorem 3.3, one can show that $\boldsymbol{\sigma}$ is a pure Nash equilibrium.

Let $\boldsymbol{\sigma}^*$ be a strategy profile defined as follows: (i) for any $j \in [m-1]$, o players of group N_j select resource $r_{j+1,0}$, and each of the $k-o$ remaining players of N_j selects a distinct resource of $R_j \setminus \{r_{j,0}\}$; (ii) all the k players of group N_m select a distinct resource of $R_m \setminus \{r_{m,0}\}$. Thus, in $\boldsymbol{\sigma}^*$, any resource of type $r_{j,0}$ with $j > 1$ has congestion o, resource $r_{1,0}$ has null congestion, and the remaining resources have unitary congestion (see Figure 1(b)).

By standard calculations, and similarly to Theorem 3.3, one can show that, for any $\epsilon > 0$, there exists a sufficiently large m such that

$$\mathsf{NPoA}(\mathsf{LB}(m)) \geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)} \geq \left(\frac{f(k+1)}{f(o)}\right)^{\frac{o}{k}} - \epsilon. \tag{19}$$

Now, we fix an arbitrary $M < \xi(C)$, and we choose $f \in C$, $k \in \mathbb{N}$, $o \in [k]$, and $\epsilon > 0$ in such a way that

$$\left(\frac{f(k+1)}{f(o)}\right)^{\frac{o}{k}} > M + \epsilon; \tag{20}$$

we observe that, by definition of $\xi(C)$, the above parameters exist. By putting inequalities (19) and (20) together, we have that $\mathsf{NPoA}(\mathsf{LB}(m)) > M$ for a sufficiently large m, and this shows the claim. □

The following result for polynomial latency functions holds and can be shown analogously to Corollary 3.5 (the proof is deferred to the Appendix).

Corollary 3.9 (Polynomial Latencies). *The Nash price of anarchy of unweighted load-balancing games with polynomial latency functions of maximum degree p is $\mathsf{NPoA}(\mathsf{ULB}(\mathcal{P}(p))) = \xi(\mathcal{P}(p)) = 2^p$.*

3.3 The NPoA for Non-atomic Load-balancing Games

In this subsection, we consider Non-atomic Load balancing Games, in which there is a finite number of player types (belonging to set N), and an amount $r_i \in \mathbb{R}_{\geq 0}$ is associated to type i (for every $i \in N$). It is worth noticing that these games represent a good approximation for atomic ones when players become infinitely many and the contribution of each player to social welfare becomes infinitesimally small.

We first provide an upper bound to the Nash price of anarchy of non-atomic load-balancing games. Given a class of latency function C, define

$$\eta(C) := \sup_{f \in C, k \geq o > 0} \left(\frac{f(k)}{f(o)}\right)^{\frac{o}{k}},$$

where k and o are two positive real numbers.

Theorem 3.10 (Upper Bound). *Let C be a class of latency functions. The Nash price of anarchy of non-atomic load-balancing games with latency functions in C is $\mathsf{NPoA}(\mathsf{NLB}(C)) \leq \eta(C)$.*

Proof. Let $\mathsf{NLB} \in \mathsf{NLB}(C)$ be a non-atomic load-balancing game with latency functions in C, and let Δ and Δ^* be a worst-case pure Nash equilibrium and an optimal strategy profile of NLB, respectively. Let k_j denote $k_j(\Delta)$ and o_j denote $k_j(\Delta^*)$.

For any player type i and pair (j, j^*) of resources, let α^i_{j,j^*} be the amount of players of type i selecting resource j in Δ and resource j^* in Δ^*. Clearly, it holds that, for any $i \in N$, $\sum_{j,j^* \in R} \alpha^i_{j,j^*} = r_i$.

Since Δ is a pure Nash equilibrium, if there exists $i \in N$ such that $\alpha^i_{j,j^*} > 0$, then we have that $cost_j(\Delta) \leq cost_{j^*}(\Delta)$. For any $j, j^* \in R$, let $A_{j,j^*} = \sum_{i \in N} \alpha^i_{j,j^*}$. Clearly, it holds that

$$cost_j(\Delta)^{A_{j,j^*}} \leq cost_{j^*}(\Delta)^{A_{j,j^*}}. \tag{21}$$

Since, for any $j \in R(\Delta)$, $\sum_{j^* \in R} A_{j,j^*} = k_j$ and, symmetrically, for any $j^* \in R(\Delta^*)$, $\sum_{j \in R} A_{j,j^*} = o_j$, it follows that

$$\prod_{j,j^* \in R} cost_j(\Delta)^{A_{j,j^*}} = \prod_{j \in R(\Delta)} cost_j(\Delta)^{k_j} \tag{22}$$

and

$$\prod_{j,j^* \in R} cost_{j^*}(\Delta^*)^{A_{j,j^*}} = \prod_{j \in R(\Delta^*)} cost_j(\Delta)^{o_j}. \tag{23}$$

By multiplying (21) over all pairs of resources in R and by exploiting (22) and (23), we obtain

$$\prod_{j \in R(\Delta)} \ell_j(k_j)^{k_j} = \prod_{j \in R(\Delta)} cost_j(\Delta)^{k_j} = \prod_{j,j^* \in R} cost_j(\Delta)^{A_{j,j^*}}$$
$$\leq \prod_{j,j^* \in R} cost_{j^*}(\Delta)^{A_{j,j^*}} = \prod_{j \in R(\Delta^*)} cost_j(\Delta)^{o_j} = \prod_{j \in R(\Delta^*)} \ell_j(k_j)^{o_j}. \tag{24}$$

By exploiting the properties of the logarithmic function, we get

$$\begin{aligned}
\ln(\text{NPoA}(\text{LB})) &= \ln\left(\frac{\left(\prod_{j \in R(\Delta)} \ell_j(k_j)^{k_j}\right)^{\frac{1}{\sum_{i \in N} r_i}}}{\left(\prod_{j \in R(\Delta^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i \in N} r_i}}}\right) \\
&\leq \ln\left(\frac{\left(\prod_{j \in R(\Delta^*)} \ell_j(k_j)^{o_j}\right)^{\frac{1}{\sum_{i \in N} r_i}}}{\left(\prod_{j \in R(\Delta^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i \in N} r_i}}}\right) \qquad (25) \\
&= \frac{\sum_{j \in R(\Delta^*)} o_j \ln(\ell_j(k_j)) - \sum_{j \in R(\Delta^*)} o_j \ln(\ell_j(o_j))}{\sum_{i \in N} r_i} \\
&= \frac{\sum_{j \in R(\Delta^*)} o_j(\ln(\ell_j(k_j)) - \ln(\ell_j(o_j)))}{\sum_{j \in R} k_j}, \\
&\leq \frac{\sum_{j \in R_+} o_j(\ln(\ell_j(k_j)) - \ln(\ell_j(o_j)))}{\sum_{j \in R_+} k_j} \qquad (26) \\
&\leq \max_{j \in R_+} \frac{o_j(\ln(\ell_j(k_j)) - \ln(\ell_j(o_j)))}{k_j} \\
&\leq \sup_{f \in C, k \geq o > 0} \frac{o(\ln(f(k)) - \ln(f(o)))}{k},
\end{aligned}$$

where (25) comes from (24), and (26) is obtained by using similar arguments as in Theorem 3.7 (in particular, see inequalities (18)). Therefore, we conclude that

$$\ln(\text{NPoA}(\text{NLB})) \leq \sup_{f \in C, k \geq o > 0} \frac{o(\ln(f(k)) - \ln(f(o)))}{k},$$

and by exponentiating the previous inequality, we get the claim. □

In the following theorem, we show that the upper bound derived in Theorem 3.10 is tight under mild assumptions on the considered latency functions, even for symmetric games. In particular, the considered tight lower-bound is a simple Pigou-like network [82], which is a symmetric load-balancing game with two resources only; thus, the worst-case price of anarchy is attained by the simplest possible combinatorial structure, as well as this fact holds for the utilitarian social welfare [84].

Theorem 3.11 (Lower Bound). *Let C be a class of latency functions. If C is all-constant-including, then* NPoA(SNLB(C)) $\geq \eta(C)$.

Proof. To show the theorem, we equivalently show that, for any $M < \eta(C)$, there exists a symmetric non-atomic load-balancing game NLB $\in$ SNLB(C) such that NPoA(NLB) $> M$. Fix an arbitrary $M < \eta(C)$. Let $f \in C$ and $k \geq o > 0$ two positive real numbers such that $(\frac{f(k)}{f(o)})^{\frac{o}{k}} > M$. Let NLB be a symmetric non-atomic load-balancing game with a unique player type, say, 1, and two resources having latency defined as $\ell_1(x) := f(x)$ and $\ell_2(x) := f(k)$. Assume that the amount of players of type 1 is $r_1 = k$. Let Δ be the strategy profile in which all players select resource 1, and let Δ^* be the strategy profile in which an amount o of players selects resource 1 and the remaining one (i.e., $k - o$) selects resource 2. We trivially have that Δ is a pure Nash equilibrium. Thus, we obtain

$$\mathsf{NPoA}(\mathsf{NLB}) \geq \frac{\mathsf{NSW}(\Delta)}{\mathsf{NSW}(\Delta^*)} = \left(\frac{\ell_1(k)^k}{\ell_1(o)^o \ell_2(k-o)^{k-o}}\right)^{\frac{1}{k}} = \left(\frac{f(k)^k}{f(o)^o f(k)^{k-o}}\right)^{\frac{1}{k}} = \left(\frac{f(k)}{f(o)}\right)^{\frac{o}{k}} > M,$$

and the claim follows. □

The following result for polynomial latency functions holds.

Corollary 3.12 (Polynomial Latencies). *The Nash price of anarchy of non-atomic load-balancing games with polynomial latency functions of maximum degree p (even for symmetric games) is* $\mathsf{NPoA}(\mathsf{NLB}(\mathcal{P}(p))) = \mathsf{NPoA}(\mathsf{SNLB}(\mathcal{P}(p))) = \eta(\mathcal{P}(p)) = (e^{\frac{1}{e}})^p \simeq (1.44)^p$.

Proof. By applying Theorems 3.10 and 3.11, we get that

$$\mathsf{NPoA}(\mathsf{NLB}(\mathcal{P}(p))) = \eta(\mathcal{P}(p))) = \sup_{\alpha_0,\alpha_1,\ldots,\alpha_p \geq 0, k\geq o>0} \left(\frac{\sum_{d=0}^p \alpha_d k^d}{\sum_{d=0}^p \alpha_d o^d}\right)^{\frac{o}{k}} = \sup_{k\geq o>0}\left(\max_{d\in[p]\cup\{0\}} \frac{k^d}{o^d}\right)^{\frac{o}{k}}$$

$$= \max_{d\in[p]\cup\{0\}} \sup_{k\geq o>0}\left(\frac{k^d}{o^d}\right)^{\frac{o}{k}} = \max_{d\in[p]\cup\{0\}}\left(\sup_{k\geq o>0}\left(\frac{k}{o}\right)^{\frac{o}{k}}\right)^d = \left(\sup_{k\geq o>0}\left(\frac{k}{o}\right)^{\frac{o}{k}}\right)^p = \left(\sup_{x>0} x^{\frac{1}{x}}\right)^p = \left(e^{\frac{1}{e}}\right)^p.$$

□

4 ONLINE LOAD BALANCING

Recall that, in the setting considered in this section, clients appear in online fashion, in consecutive steps; when a client appears, an irrevocable decision has to be taken to assign it to a resource. We assume that clients appear in increasing order, i.e., client $i \in [n]$ appears before client $j \in [n]$ if and only if $i < j$.

We first provide an upper bound on the competitive ratio of the greedy algorithm. Given a class of latency functions C, define

$$\zeta(C) := \sup_{\substack{f_1,f_2\in C,\\ k_1,k_2,o_1,o_2\in\mathbb{R}: k_1\geq o_1>0, o_2>k_2\geq 0}} \left(\frac{f_1(k_1+o_1)^{k_1+o_1}}{f_1(k_1)^{k_1} f_1(o_1)^{o_1}}\right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}} \left(\frac{f_2(k_2+o_2)^{k_2+o_2}}{f_2(k_2)^{k_2} f_2(o_2)^{o_2}}\right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}},$$

where we set $f_2(0)^0 := 1$.

Theorem 4.1 (Upper Bound). *Let C be a class of quasi-log-convex functions. The competitive ratio of the greedy algorithm* G *applied to load-balancing instances with latency functions in C is* $\mathsf{CR}_{\mathsf{G}}(\mathsf{WLB}(C)) \leq \zeta(C)$.

Proof. The high-level structure of the proof is similar to that of Theorem 3.1, but here we resort to more sophisticated calculations and we use the further hypothesis of quasi-log-convex latency functions.

Let $\mathsf{I} \in \mathsf{WLB}(C)$ be a load-balancing instance with latency functions in C, and let $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ be the states returned by the greedy algorithm and an optimal strategy profile of LB, respectively.

Let k_j denote $k_j(\boldsymbol{\sigma})$ and o_j denote $k_j(\boldsymbol{\sigma}^*)$. For any $i \in N$ and resource j, let $(\boldsymbol{\sigma}^i)$ be the partial state in which the first i clients have been assigned according to $\boldsymbol{\sigma}$, and let $(\boldsymbol{\sigma}^{i-1}, j)$ be the state in which the first $i-1$ clients have been assigned according to $\boldsymbol{\sigma}$ and client i is assigned to resource j. By definition of greedy algorithm, we have that $\sigma_i \in \arg\min_{j \in R} \mathsf{NSW}(\boldsymbol{\sigma}^{i-1}, j) = \arg\min_{j \in R} \frac{\prod_{l \leq i} cost_l(\boldsymbol{\sigma}^{i-1}, j)}{\prod_{l \leq i-1} cost_l(\boldsymbol{\sigma}^{i-1})} = \arg\min_{j \in R} \frac{\ell_j(k_j(\boldsymbol{\sigma}^{i-1}, j))^{k_j(\boldsymbol{\sigma}^{i-1}, j)}}{\ell_j(k_j(\boldsymbol{\sigma}^{i-1}))^{k_j(\boldsymbol{\sigma}^{i-1})}}$, where we set $\ell_j(0)^0 := 1$. Thus, we can equivalently define the greedy assignment by saying that each client i is assigned to the resource j minimizing $\frac{\ell_j(k_j(\boldsymbol{\sigma}^{i-1}, j))^{k_j(\boldsymbol{\sigma}^{i-1}, j)}}{\ell_j(k_j(\boldsymbol{\sigma}^{i-1}))^{k_j(\boldsymbol{\sigma}^{i-1})}}$, so

$$\frac{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^i))^{k_{\sigma_i}(\boldsymbol{\sigma}^i)}}{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i}(\boldsymbol{\sigma}^{i-1})}} \leq \frac{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*)}}{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1})}}. \tag{27}$$

We have that:

$$\begin{aligned}
\prod_{i \in N} \frac{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^i))^{k_{\sigma_i}(\boldsymbol{\sigma}^i)}}{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i}(\boldsymbol{\sigma}^{i-1})}} &= \prod_{j \in R(\boldsymbol{\sigma})} \prod_{i \in N: \sigma_i = j} \frac{\ell_j(k_j(\boldsymbol{\sigma}^i))^{k_j(\boldsymbol{\sigma}^i)}}{\ell_j(k_j(\boldsymbol{\sigma}^{i-1}))^{k_j(\boldsymbol{\sigma}^{i-1})}} \\
&= \prod_{j \in R(\boldsymbol{\sigma})} \ell_j(k_j(\boldsymbol{\sigma}^n))^{k_j(\boldsymbol{\sigma}^n)} && (28) \\
&= \prod_{j \in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j}, && (29)
\end{aligned}$$

where (28) is obtained by exploiting telescoping properties. Furthermore, we get

$$\begin{aligned}
&\prod_{i \in N} \frac{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*)}}{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1})}} \\
&= \prod_{i \in N} \frac{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}) + w_i)^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}) + w_i}}{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1})}} \\
&\leq \prod_{i \in N} \frac{\ell_{\sigma_i^*}(k_{\sigma_i^*} + w_i)^{k_{\sigma_i^*} + w_i}}{\ell_{\sigma_i^*}(k_{\sigma_i^*})^{k_{\sigma_i^*}}} && (30) \\
&= \prod_{j \in R(\boldsymbol{\sigma}^*)} \prod_{i \in N: \sigma_i^* = j} \frac{\ell_j(k_j + w_i)^{k_j + w_i}}{\ell_j(k_j)^{k_j}} \\
&\leq \prod_{j \in R(\boldsymbol{\sigma}^*)} \prod_{i \in N: \sigma_i^* = j} \frac{\ell_j(k_j + \sum_{t \leq i: \sigma_t^* = j} w_t)^{k_j + \sum_{t \leq i: \sigma_t^* = j} w_t}}{\ell_j(k_j + \sum_{t < i: \sigma_t^* = j} w_t)^{k_j + \sum_{t < i: \sigma_t^* = j} w_t}} && (31) \\
&= \prod_{j \in R(\boldsymbol{\sigma}^*)} \frac{\ell_j(k_j + \sum_{t: \sigma_t^* = j} w_t)^{k_j + \sum_{t: \sigma_t^* = j} w_t}}{\ell_j(k_j)^{k_j}} && (32) \\
&= \prod_{j \in R(\boldsymbol{\sigma}^*)} \frac{\ell_j(k_j + o_j)^{k_j + o_j}}{\ell_j(k_j)^{k_j}}, && (33)
\end{aligned}$$

where (32) is obtained by exploiting telescoping properties, and (30) and (31) easily come from the following fact:

Fact 1. *Given a quasi-log-convex latency function* f, *we have that* $\frac{f(x+z)^{x+z}}{f(x)} \leq \frac{f(x+y+z)^{x+y+z}}{f(x+y)^{x+y}}$ *for any* $x, y, z \geq 0$.

Proof. Since the function g such that $g(t) = t\ln(f(t))$ is convex, we have that $g(x+z) - g(x) \leq g(x+y+z) - g(x+y)$ for any $x, y, z \geq 0$, thus, by exponentiating the previous inequality, the claim follows. □

By putting together (27), (29), and (33), we get

$$\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j} = \prod_{i\in N} \frac{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^i))^{k_{\sigma_i}(\boldsymbol{\sigma}^i)}}{\ell_{\sigma_i}(k_{\sigma_i}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i}(\boldsymbol{\sigma}^{i-1})}} \leq \prod_{i\in N} \frac{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}, \sigma_i^*)}}{\ell_{\sigma_i^*}(k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1}))^{k_{\sigma_i^*}(\boldsymbol{\sigma}^{i-1})}} \leq \prod_{j\in R(\boldsymbol{\sigma}^*)} \frac{\ell_j(k_j+o_j)^{k_j+o_j}}{\ell_j(k_j)^{k_j}}. \tag{34}$$

By exploiting the properties of the logarithmic function, we obtain

$$\ln(\mathrm{CR}_\mathrm{G}(\mathrm{I})) = \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma})} \ell_j(k_j)^{k_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}\right) \leq \ln\left(\frac{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \frac{\ell_j(k_j+o_j)^{k_j+o_j}}{\ell_j(k_j)^{k_j}}\right)^{\frac{1}{\sum_{i\in N} w_i}}}{\left(\prod_{j\in R(\boldsymbol{\sigma}^*)} \ell_j(o_j)^{o_j}\right)^{\frac{1}{\sum_{i\in N} w_i}}}\right) \tag{35}$$

$$= \frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} \left((k_j+o_j)\ln(\ell_j(k_j+o_j)) - k_j\ln(\ell_j(k_j)) - o_j\ln(\ell_j(o_j))\right)}{\sum_{i\in N} w_i}, \tag{36}$$

where (35) comes from (34). Since $\sum_{i\in N} w_i = \sum_{j\in R} k_j = \sum_{j\in R} o_j$, we have that (36) is upper bounded by the optimal solution of the following optimization problem OP on some linear variables $(\alpha_j)_{j\in R}$:

$$\mathrm{OP}: \quad \max \quad \frac{\sum_{j\in R(\boldsymbol{\sigma}^*)} \alpha_j\left((k_j+o_j)\ln(\ell_j(k_j+o_j)) - k_j\ln(\ell_j(k_j)) - o_j\ln(\ell_j(o_j))\right)}{\sum_{j\in R}\alpha_j k_j}$$
$$\text{s.t.} \quad \sum_{j\in R}\alpha_j k_j = \sum_{j\in R}\alpha_j o_j, \quad \alpha_j \geq 0\ \forall j\in R.$$

We have the following lemma, whose proof is omitted, since it is similar to that of Lemma 3.2.

Lemma 4.2. *The optimal value of* OP *is at most*

$$\sup_{k_1\geq o_1>0,\, o_2>k_2\geq 0,\, f_1,f_2\in C} \frac{(o_2-k_2)F(f_1,o_1,k_1) + (k_1-o_1)F(f_2,o_2,k_2)}{k_1 o_2 - k_2 o_1},$$

where $F(f,o,k) := (k+o)\ln(f(k+o)) - k\ln(f(k)) - o\ln(f(o))$.

By continuing from (36) and by using Lemma 4.2, we get

$$\ln(\mathsf{CR}_{\mathsf{G}}(\mathsf{I})) \leq \sup_{k_1 \geq o_1 > 0,\, o_2 > k_2 \geq 0,\, f_1, f_2 \in C} \frac{(o_2 - k_2)F(f_1, o_1, k_1) + (k_1 - o_1)F(f_2, o_2, k_2)}{k_1 o_2 - k_2 o_1}.$$

By exponentiating the previous inequality, we get the claim. □

We have that the analysis derived in Theorem 4.1 is tight if the considered latency functions are abscissa-scaling and ordinate-scaling.

Theorem 4.3 (Lower Bound). *Let C be a class of latency functions and let* G *be the greedy algorithm. If C is abscissa-scaling and ordinate-scaling, then* $\mathsf{CR}_{\mathsf{G}}(\mathsf{WLB}(C)) \geq \zeta(C)$.

The proof of Theorem 4.3 is technically similar to that of Theorem 3.3 and is deferred to the Appendix.

The following result for polynomial latency functions holds.

Corollary 4.4 (Polynomial Latencies). *The competitive ratio of the greedy algorithm applied to weighted load-balancing instances with polynomial latency functions of maximum degree p is* $\mathsf{CR}_{\mathsf{G}}(\mathsf{WLB}(\mathcal{P}(p))) = \zeta(\mathcal{P}(p)) = 4^p$.

The proof of Corollary 4.4 is analogue to that of Corollary 3.5 and is deferred to the Appendix.

We show that, when considering polynomial latency functions, the upper bound of Corollary 4.4 is tight for any online algorithm, i.e., we are able to provide a matching lower bound to the online load-balancing problem.

Theorem 4.5 (Polynomial Latencies - Lower Bound w.r.t. Any Online Algorithm). *The competitive ratio of any online algorithm* A *applied to load-balancing instances with polynomial latencies of maximum degree p is at least* $\mathsf{CR}_{\mathsf{A}}(\mathcal{P}(p)) \geq 4^p$*, even for instances with identical resources.*

Sketch of the proof. We equivalently show that, for any online algorithm A and $\epsilon > 0$, there exists a load-balancing instance I such that $\mathsf{CR}_{\mathsf{A}}(\mathsf{I}) \geq 4^p - \epsilon$. We construct an instance similar to that defined in Theorem 17 of Reference [33]. Given an integer $m \geq 0$ and a real number $w > 0$, let $\mathsf{I}(m)$ be a load-balancing instance with identical polynomial latency functions of type $\ell(x) = x^p$, and recursively defined as follows:

- If $m = 0$, then $\mathsf{I}(m)$ has no clients and there is a unique resource denoted as *fundamental resource* of $\mathsf{I}(0)$.
- If $m \geq 1$, then: (i) $\mathsf{I}(m)$ contains a sub-instance equivalent to $\mathsf{I}(i-1)$ for any $i \in [m]$; (ii) $\mathsf{I}(m)$ has a further resource r denoted as *fundamental resource* of $\mathsf{I}(m)$; (iii) there are further m clients such that, for any $i \in [m]$, the ith client has weight $w_i := 2^{i-1}$ and can select among r and the fundamental resource $r(i)$ of the sub-instance of type $\mathsf{I}(i-1)$ included in $\mathsf{I}(m)$; (iv) for any client $i \in [m]$, r and $r(i)$ are, respectively, denoted as first and second resource of the ith client included in $\mathsf{I}(m)$.

One can show that, for any $\epsilon > 0$, there exists a sufficiently large m such that $\mathsf{CR}_{\mathsf{A}}(\mathsf{I}(m)) \geq 4^p - \epsilon$, and this shows the claim. The proof of this last fact is deferred to the Appendix. □

5 CONCLUDING REMARKS AND OPEN PROBLEMS

To the best of our knowledge, this is the first work that adopts the NSW as a benchmarking quality measure in load-balancing problems. Several open problems deserve further investigation.

Our article mostly focuses on evaluating the performance of selfish and online load balancing. Concerning complexity issues, it is worth noticing that, on the one hand, when considering unweighted players, an optimal configuration with respect to the NSW can be easily computed in polynomial time by exploiting the same techniques developed in References [38, 77] for the USW; on the other hand, when considering weighted players, a simple reduction from the NP-complete problem PARTITION shows that the problem becomes NP-hard. Therefore, an interesting open problem is that of providing better polynomial time approximation algorithms for the weighted case and polynomial latency functions (we notice that, as shown in Corollary 4.4, the greedy algorithm provides a constant approximation factor).

Finally, other interesting research directions connected with our work are the following: (i) considering other quality metrics to evaluate the performance under the NSW (e.g., the price of stability [5, 16, 20, 40, 41, 55], the efficiency of one-round walks starting from the empty-state [18, 23, 44, 73]); (ii) analyzing other sub-classes of load-balancing games and online load-balancing problems (e.g., symmetric unweighted games, unweighted instances for the online setting); (iii) applying the NSW to other resource selection games (e.g., congestion games [83] and their variants [5, 17, 19, 22, 26, 27, 49, 51, 59]) and to more general cost-minimization settings; (iv) designing mechanisms to improve the performance under the NSW (e.g., taxing mechanisms [24, 36, 47], Stackelberg strategies [25, 54, 58, 86], coordination and cost-sharing mechanisms [32, 34, 43, 45, 65, 70]).

A APPENDIX

A.1 Lower Bound for Linear Congestion Games

Unweighted congestion games are a further generalization of unweighted load-balancing games. The difference is that the strategy set of each player $i \in N$ is a collection $\Sigma_i \subseteq 2^R \setminus \{\emptyset\}$, i.e., a strategy is a non-empty subset of R. Furthermore, given a strategy profile $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_n)$ (with $\sigma_i \in \Sigma_i$), the cost of each player $i \in N$ is $cost_i(\boldsymbol{\sigma}) := \sum_{j \in \sigma_i} \ell_j(k_j(\boldsymbol{\sigma}))$, where $k_j(\boldsymbol{\sigma}) := |i \in N : j \in \sigma_i|$ is the congestion of resource j in strategy profile $\boldsymbol{\sigma}$. In the following theorem, we show that, even for linear latency functions, the Nash price of anarchy of unweighted congestion games with linear latency functions is non-constant in the number of players, differently from the case of load-balancing games. This fact exhibits a substantial difference with respect to the case of the price of anarchy when the considered social welfare function is the sum of the players' costs. Indeed, in such case, the price of anarchy for linear congestion games is finite, and the price of anarchy of load-balancing games is as high as that of general linear congestion games.

Theorem A.1. *The Nash price of anarchy of linear congestion games is at least $n^{1-o(1)}$, where n is the number of players (and $o(1)$ is an infinitesimal w.r.t. to n).*

Proof. We show that, for any $\epsilon \in (0, 1/2)$, there exists a congestion game CG with linear latency functions and $n \geq 2$ players such that:

$$\mathrm{NPoA(CG)} \geq \lceil n\epsilon \rceil^{1-\frac{\lceil n\epsilon \rceil}{n}}, \tag{37}$$

and this fact will imply the claim, as $\lceil n\epsilon \rceil^{1-\frac{\lceil n\epsilon \rceil}{n}} \in \Theta(n^{1-\epsilon})$ for any fixed $\epsilon \in (0, 1/2)$. Let $\epsilon \in (0, 1/2)$, $n \geq 2$, and $m := \lceil n\epsilon \rceil$. Let $\mathrm{CG}(n, \epsilon)$ be an unweighted congestion game with n players defined as follows: The set of resources is organized into three groups R_1, R_2, R_3, with $R_j := \{r_{j,1}, \dots, r_{j,n-m}\}$ for any $j \in [2]$, and $R_3 := \{r_{3,1}, \dots, r_{3,m}\}$. The latency function of each resource $r_{j,h}$ is $\ell_{r_{j,h}}(x) := \alpha_j x$, where $\alpha_1 = m + 1$, $\alpha_2 = 1$, and $\alpha_3 = m$. There are two groups of players N_1, N_2, with $N_1 := \{i_{1,1}, \dots, i_{1,n-m}\}$ and $N_2 := \{i_{2,1}, \dots, i_{2,m}\}$. Each player $i_{1,h} \in N_1$ has two strategies $S_{1,h}$ and $S^*_{1,h}$ defined as $S_{1,h} := \{r_{1,h}\}$ and $S^*_{1,h} := \{r_{2,h}\}$, and each player $i_{2,h} \in N_2$ has two strategies $S_{2,h}$ and $S^*_{2,h}$ defined as $S_{2,h} := R_2$ and $S^*_{3,h} := \{r_{3,h}\}$. Let $\boldsymbol{\sigma}$ (respectively, $\boldsymbol{\sigma}^*$) be the strategy profile such that each player $i_{t,h}$ plays strategy $S_{t,h}$ (respectively, $S^*_{t,h}$), for any $t \in [2]$. One can easily show that $cost_i(\boldsymbol{\sigma}) = cost_i(\boldsymbol{\sigma}_{-i}, \sigma_i^*)$ for any player i, thus $\boldsymbol{\sigma}$ is a pure Nash equilibrium. We have that:

$$\begin{aligned}
&\mathsf{NPoA}(\mathsf{CG}(n,\epsilon))\\
&\geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)}\\
&= \left(\left(\prod_{i\in N_1}\frac{cost_i(\boldsymbol{\sigma})}{cost_i(\boldsymbol{\sigma}^*)}\right)\left(\prod_{i\in N_2}\frac{cost_i(\boldsymbol{\sigma})}{cost_i(\boldsymbol{\sigma}^*)}\right)\right)^{\frac{1}{n}}\\
&= \left(\left(\prod_{i\in N_1}\frac{cost_i(\boldsymbol{\sigma}_{-i},\sigma_i^*)}{cost_i(\boldsymbol{\sigma}^*)}\right)\left(\prod_{i\in N_2}\frac{cost_i(\boldsymbol{\sigma}_{-i},\sigma_i^*)}{cost_i(\boldsymbol{\sigma}^*)}\right)\right)^{\frac{1}{n}}\\
&= \left(\left(\frac{\alpha_2(m+1)}{\alpha_2}\right)^{n-m}\left(\frac{\alpha_3}{\alpha_3}\right)^{m}\right)^{\frac{1}{n}}\\
&= (m+1)^{\frac{n-m}{n}}\\
&\geq \lceil n\epsilon\rceil^{1-\frac{\lceil n\epsilon\rceil}{n}},
\end{aligned} \tag{38}$$

thus (37) holds, and the claim follows. □

A.2 Proof of Theorem 3.3 (full version)

First, we deal with part (ii) of the claim: Let us assume that C is abscissa-scaling, ordinate-scaling, and unbounded-including. To prove part (ii), we equivalently show that for any $M < \psi(C)$ there exists a game $\mathsf{LB} \in \mathsf{WLB}(C)$ such that $\mathsf{NPoA}(\mathsf{LB}) > M$.

We first assume without loss of generality that C contains at least a non-constant latency function. Indeed, if it is not case, then we have that $\psi(C) = 1 \leq \mathsf{NPoA}(C)$, and the claim immediately follows.[6] Let $f_1, f_2 \in C, k_1, k_2, o_1, o_2 \geq 0$ such that $k_1 \geq o_1 > 0, o_2 > k_2 \geq 0$, and a sufficiently small $\epsilon > 0$ such that

$$\left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}}\left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}} > M+\epsilon.$$

Let f, g be two latency functions such that $f(x) := f_1(o_1x)$ and $g(x) := f_2(o_2x)$, and let $k := k_1/o_1$ and $h := k_2/o_2$ two positive real numbers; we observe that f, g belong to C, since C is abscissa-scaling and ordinate-scaling. Since

$$\left(\frac{f_1(k_1+o_1)}{f_1(o_1)}\right)^{\frac{(o_2-k_2)o_1}{k_1o_2-k_2o_1}}\left(\frac{f_2(k_2+o_2)}{f_2(o_2)}\right)^{\frac{(k_1-o_1)o_2}{k_1o_2-k_2o_1}} = \left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}}\left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}},$$

we have that

$$\left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}}\left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}} > M+\epsilon, \text{ for some } f,g\in C,\ k\geq 1, \text{ and } h<1. \tag{39}$$

Observe that f and g can be chosen in such a way that they are non-constant functions. Indeed, if one function among f and g is constant, then it is sufficient replacing it with an arbitrary non-constant function of C, so (39) holds as well (we recall that we have initially assumed without

[6]If all the latency functions of C are constant, then we can alternatively show that $\mathsf{NPoA}(C) = 1$ without invoking Theorem 3.1. Indeed, given a game in which all the latency functions are constant, the equilibrium strategy of each player is to select the cheapest resource (whose cost does not depend on the players' actions) within her strategy set, thus any equilibrium is also an optimal strategy profile.

loss of generality that C contains at least a non-constant latency function). Since C is unbounded-including and f, g are non-constant, we have that $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$.

First, we assume that $h > 0$. Given two integers $m \geq 3$ and $s \geq 1$, let $\mathsf{LB}(m,s)$ be a symmetric weighted load-balancing game where the resources are partitioned into $2m$ groups $R_1, R_2, R_3 \ldots, R_{2m}$. Each group R_j has s^{j-1} resources and the latency function of each resource $r \in R_j$ is defined as $\ell_r(x) := \alpha_j \hat{f}_j(\beta_j x)$ with

$$\hat{f}_j := \begin{cases} f & \text{if } j \leq m-1 \\ g & \text{if } j \geq m \end{cases}, \quad \beta_j := \begin{cases} \left(\frac{s}{k}\right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{s}{h}\right)^{j-m}\left(\frac{s}{k}\right)^{m-1} & \text{if } m \leq j \leq 2m \end{cases}, \tag{40}$$

$$\alpha_j := \begin{cases} \left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{g(h)}{g(h+1)}\right)^{j-m}\left(\frac{f(k)}{g(h+1)}\right)\left(\frac{f(k)}{f(k+1)}\right)^{m-2} & \text{if } m \leq j \leq 2m-1 \\ \frac{g(h)}{g(1)}\left(\frac{g(h)}{g(h+1)}\right)^{m-1}\left(\frac{f(k)}{g(h+1)}\right)\left(\frac{f(k)}{f(k+1)}\right)^{m-2} & \text{if } j = 2m \end{cases}. \tag{41}$$

The set of players N is partitioned into $2m-1$ sets $N_1, N_2, \ldots, N_{2m-1}$, and each group N_j has s^j players having weight $w_j := 1/\beta_{j+1}$. We observe that all the latency functions of $\mathsf{LB}(m,s)$ belong to C, as C is abscissa-scaling and ordinate-scaling.

Let $\boldsymbol{\sigma}$ be the strategy profile in which, for any $j \in [2m-1]$, each resource of group R_j is selected by exactly s players of group N_j (see Figure 2(a)). Observe that, by construction of α_j, β_j, w_j, the following properties hold:

$$\begin{cases} \alpha_j f(k) = \alpha_{j+1} f(k+1) & \text{if } j \leq m-2 \\ \alpha_j f(k) = \alpha_{j+1} g(h+1) & \text{if } j = m-1 \\ \alpha_j g(h) = \alpha_{j+1} g(h+1) & \text{if } m \leq j \leq 2m-2 \\ \alpha_j g(h) = \alpha_{j+1} g(1) & \text{if } j = 2m-1 \end{cases},$$

$$\begin{cases} \beta_j w_j s = k,\ w_j|N_j| = k^j & \text{if } j \leq m-1 \\ \beta_j w_j s = h,\ w_j|N_j| = h^{j+1-m}k^{m-1} & \text{if } m \leq j \leq 2m-1 \\ \beta_{j+1} w_j = 1 & \text{if } j \leq 2m-1 \end{cases}. \tag{42}$$

By exploiting (42) and the fact that $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$, one can show the following lemma:

Lemma A.2. *By taking a sufficiently large s, the strategy profile $\boldsymbol{\sigma}$ is a pure Nash equilibrium.*

Proof. Let $j \in [2m-1]$, $t \in [2m]$, and i be an arbitrary player selecting a resource r_j of group R_j in the strategy profile $\boldsymbol{\sigma}$, and assume that she deviates to a resource r_t of group R_t. We have three cases:

- **t = j + 1:** First, assume that $j \leq m-2$. By using (42), we get $cost_i(\boldsymbol{\sigma}) = \ell_{r_j}(k_{r_j}(\boldsymbol{\sigma})) = \alpha_j \hat{f}_j(\beta_j s w_j) = \alpha_j f(k) = \alpha_{j+1} f(k+1) = \alpha_{j+1} f(\beta_{j+1} s w_{j+1} + \beta_{j+1} w_j) = \alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}(s w_{j+1} + w_j)) = \ell_{r_h}(k_{r_h}(\boldsymbol{\sigma}_{-i}, \{r_t\})) = cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$. The cases $j = m-1$, $m \leq j \leq 2m-2$, and $j = 2m-1$ can be separately considered by exploiting (42), so one can analogously show $cost_i(\boldsymbol{\sigma}) = \alpha_j \hat{f}_j(\beta_j s w_j) = \alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}(s w_{j+1} + w_j)) = cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$, where we set $w_{2m} := 0$.
- **t ≤ j :** From the previous case, we have that if one player is playing a resource at some level l, and deviates to some resource at level $l+1$, then her cost does not change. Thus, we necessarily have that the cost of each resource in strategy profile $\boldsymbol{\sigma}$ is a non-increasing

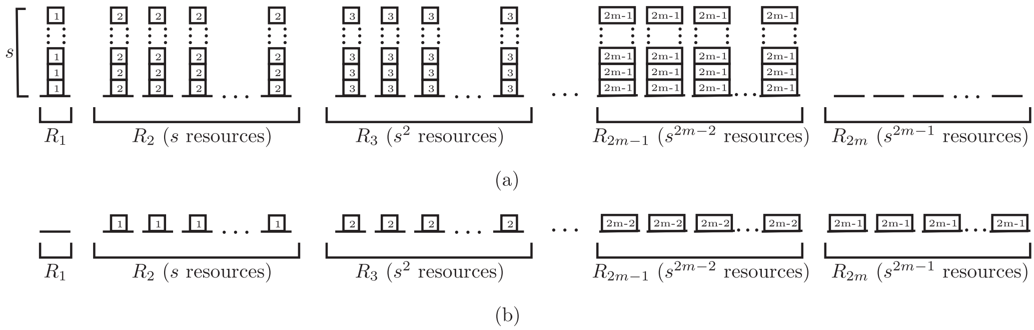

Fig. 2. The LB used in the proof of Theorem 3.3. Columns represent resources and squares represent players (number j inside a square means that the player belongs to group N_j). (a): a Nash equilibrium $\boldsymbol{\sigma}$; (b): the strategy profile $\boldsymbol{\sigma}^*$.

function of the level $l \in [2m]$ that it belongs to. Thus, since $t \leq j$, we necessarily have that $cost_i(\boldsymbol{\sigma}) \leq cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$.

- **t > j + 1** : If we consider the asymptotic behavior of $cost_i(\boldsymbol{\sigma})$ and $cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$ with respect to parameter s, then we get $cost_i(\boldsymbol{\sigma}) = \alpha_j \hat{f}_j(\beta_j s w_j) = \alpha_j \hat{f}_j(\Theta(s^{j-1} \cdot s \cdot s^{-j})) = \Theta(1)$, thus $cost_i(\boldsymbol{\sigma})$ does not depend on s; furthermore, we get $cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\}) \geq \alpha_j \hat{f}_j(\beta_t w_{j+1}) = \alpha_j \hat{f}_j(\Theta(s^{t-1} s^{-j})) \geq \alpha_j \hat{f}_j(\Theta(s))$, thus, since $\lim_{x\to\infty} \hat{f}(x) = \infty$, we have that $cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$ can be arbitrarily large as s increases. We conclude that, by taking a sufficiently large s, we get $cost_i(\boldsymbol{\sigma}) \leq cost_i(\boldsymbol{\sigma}_{-i}, \{r_t\})$ for any j and $t > j + 1$.[7]

The previous case-analysis shows that player i does not improve her cost after deviating in favor of any resource r_t at level t, for any $t \in [2m]$, and thus that $\boldsymbol{\sigma}$ is a pure Nash equilibrium of $\mathsf{LB}(m, s)$. □

For any integer $m \geq 3$, let s_m be a sufficiently large integer such that (according to Lemma A.2) $\boldsymbol{\sigma}$ is a pure Nash equilibrium of the game $\mathsf{LB}(m, s_m)$. Let $\boldsymbol{\sigma}^*$ be the strategy profile of $\mathsf{LB}(m, s_m)$ in which, for any $j \in [2m-1]$, each resource of group R_{j+1} is selected by exactly one player of group N_j (see Figure 2(b)). By exploiting the definitions of α_j,β_j, $\hat{f}_j$, w_j, and N_j, we have that:

$$
\begin{aligned}
&\mathsf{NPoA}(\mathsf{LB}(m, s_m)) \\
&\geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)} \\
&= \left(\frac{\prod_{j=1}^{2m-1} \left(\alpha_j \hat{f}_j \left(\beta_j s_m w_j \right) \right)^{|N_j| w_j}}{\prod_{j=2}^{2m} \left(\alpha_j \hat{f}_j \left(\beta_j w_{j-1} \right) \right)^{|N_{j-1}| w_{j-1}}} \right)^{\frac{1}{\sum_{j=1}^{2m-1} |N_j| w_j}} \\
&= \left(\frac{\left(\prod_{j=1}^{m-1} \left(\alpha_j f(k) \right)^{|N_j| w_j} \right) \left(\prod_{j=m}^{2m-1} \left(\alpha_j g(h) \right)^{|N_j| w_j} \right)}{\left(\prod_{j=2}^{m-1} \left(\alpha_j f(1) \right)^{|N_{j-1}| w_{j-1}} \right) \left(\prod_{j=m}^{2m} \left(\alpha_j g(1) \right)^{|N_{j-1}| w_{j-1}} \right)} \right)^{\frac{1}{\sum_{j=1}^{2m-1} |N_j| w_j}} \qquad (43)
\end{aligned}
$$

[7]We point out that the analysis of the case $t > j + 1$ (in Lemma A.2) is the unique part of the proof of Theorem 3.3 using the hypothesis $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$. Furthermore, as we can observe from the proof, if s is not sufficiently large, then $\boldsymbol{\sigma}$ is not guaranteed to be a pure Nash equilibrium. Then, if we had simplified the lower bounding instance by setting $s = 1$ (thus obtaining the symmetric variant of the lower bounding instance provided in the proof sketch of Section 3.1), then the resulting instance would have not worked for the symmetric case.

$$= \left(\frac{\left(\prod_{j=1}^{m-1} \left(\alpha_j f(k)\right)^{k^j}\right) \left(\prod_{j=m}^{2m-1} \left(\alpha_j g(h)\right)^{h^{j+1-m} k^{m-1}}\right)}{\left(\prod_{j=2}^{m-1} \left(\alpha_j f(1)\right)^{k^{j-1}}\right) \left(\prod_{j=m}^{2m} \left(\alpha_j g(1)\right)^{h^{j-m} k^{m-1}}\right)} \right)^{\frac{1}{\sum_{j=1}^{2m-1} |N_j| w_j}}$$

$$= \left(\frac{\left(\prod_{j=1}^{m-2} \left(\alpha_{j+1} f(k+1)\right)^{k^j}\right) \left(\prod_{j=m-1}^{2m-2} \left(\alpha_{j+1} g(h+1)\right)^{h^{j+1-m} k^{m-1}}\right) \left(\alpha_{2m} g(1)\right)^{h^m k^{m-1}}}{\left(\prod_{j=2}^{m-1} \left(\alpha_j f(1)\right)^{k^{j-1}}\right) \left(\prod_{j=m}^{2m} \left(\alpha_j g(1)\right)^{h^{j-m} k^{m-1}}\right)} \right)^{\frac{1}{\sum_{j=1}^{2m-1} |N_j| w_j}} \tag{44}$$

$$= \left(\frac{\left(\prod_{j=1}^{m-2} \left(\alpha_{j+1} f(k+1)\right)^{k^j}\right) \left(\prod_{j=m-1}^{2m-2} \left(\alpha_{j+1} g(h+1)\right)^{h^{j+1-m} k^{m-1}}\right) \left(\alpha_{2m} g(1)\right)^{h^m k^{m-1}}}{\left(\prod_{j=1}^{m-2} \left(\alpha_{j+1} f(1)\right)^{k^j}\right) \left(\prod_{j=m-1}^{2m-1} \left(\alpha_{j+1} g(1)\right)^{h^{j+1-m} k^{m-1}}\right)} \right)^{\frac{1}{\sum_{j=1}^{2m-1} |N_j| w_j}}$$

$$= \left(\frac{\left(\prod_{j=1}^{m-2} \left(\alpha_{j+1} f(k+1)\right)^{k^j}\right) \left(\prod_{j=m-1}^{2m-2} \left(\alpha_{j+1} g(h+1)\right)^{h^{j+1-m} k^{m-1}}\right)}{\left(\prod_{j=1}^{m-2} \left(\alpha_{j+1} f(1)\right)^{k^j}\right) \left(\prod_{j=m-1}^{2m-2} \left(\alpha_{j+1} g(1)\right)^{h^{j+1-m} k^{m-1}}\right)} \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m} k^{m-1}}}$$

$$= \left(\left(\prod_{j=1}^{m-2} \left(\frac{f(k+1)}{f(1)} \right)^{k^j} \right) \left(\prod_{j=m-1}^{2m-2} \left(\frac{g(h+1)}{g(1)} \right)^{h^{j+1-m} k^{m-1}} \right) \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m} k^{m-1}}}$$

$$= \left(\left(\frac{f(k+1)}{f(1)} \right)^{\sum_{j=1}^{m-2} k^j} \left(\frac{g(h+1)}{g(1)} \right)^{\sum_{j=m-1}^{2m-2} h^{j+1-m} k^{m-1}} \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m} k^{m-1}}}, \tag{45}$$

where (43) and (44) come from (42). We have two cases: $k > 1$ and $k = 1$. If $k > 1$, then by continuing from (45) and by considering a sufficiently large m, we get

$$\left(\left(\frac{f(k+1)}{f(1)} \right)^{\sum_{j=1}^{m-2} k^j} \left(\frac{g(h+1)}{g(1)} \right)^{\sum_{j=m-1}^{2m-2} h^{j+1-m} k^{m-1}} \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m} k^{m-1}}}$$

$$= \left(\left(\frac{f(k+1)}{f(1)} \right)^{\frac{k^{m-1}-k}{k-1}} \left(\frac{g(h+1)}{g(1)} \right)^{k^{m-1}\left(\frac{1-h^m}{1-h}\right)} \right)^{\frac{1}{\frac{k^{m-1}-k}{k-1} + k^{m-1}\left(\frac{1-h^{m+1}}{1-h}\right)}}$$

$$= \left(\frac{f(k+1)}{f(1)} \right)^{\frac{\frac{k^{m-1}-k}{k-1}}{\frac{k^{m-1}-k}{k-1} + k^{m-1}\left(\frac{1-h^{m+1}}{1-h}\right)}} \left(\frac{g(h+1)}{g(1)} \right)^{\frac{k^{m-1}\left(\frac{1-h^m}{1-h}\right)}{\frac{k^{m-1}-k}{k-1} + k^{m-1}\left(\frac{1-h^{m+1}}{1-h}\right)}}$$

$$= \left(\frac{f(k+1)}{f(1)} \right)^{\frac{\frac{1-h}{1-h^{m+1}}}{\frac{1-h}{1-h^{m+1}} + k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)}} \left(\frac{g(h+1)}{g(1)} \right)^{\frac{k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)\left(\frac{1-h^m}{1-h^{m+1}}\right)}{\frac{1-h}{1-h^{m+1}} + k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)}}$$

$$\geq \lim_{m \to \infty} \left(\frac{f(k+1)}{f(1)} \right)^{\frac{\frac{1-h}{1-h^{m+1}}}{\frac{1-h}{1-h^{m+1}} + k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)}} \left(\frac{g(h+1)}{g(1)} \right)^{\frac{k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)\left(\frac{1-h^m}{1-h^{m+1}}\right)}{\frac{1-h}{1-h^{m+1}} + k^{m-1}\left(\frac{k-1}{k^{m-1}-k}\right)}} - \epsilon \tag{46}$$

$$= \left(\frac{f(k+1)}{f(1)} \right)^{\frac{1-h}{(1-h)+(k-1)}} \left(\frac{g(h+1)}{g(1)} \right)^{\frac{k-1}{(1-h)+(k-1)}} - \epsilon \tag{47}$$

$$
\begin{aligned}
&= \left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}} \left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}} - \epsilon \\
&> M + \epsilon - \epsilon && (48)\\
&= M, && (49)
\end{aligned}
$$

where (46) holds if m is sufficiently large, (47) comes from the fact that $k > 1$ and $h < 1$, and (48) comes from (39).

If $k = 1$, then by continuing from (45), we get:

$$
\begin{aligned}
&\left(\left(\frac{f(k+1)}{f(1)}\right)^{\sum_{j=1}^{m-2} k^j} \left(\frac{g(h+1)}{g(1)}\right)^{\sum_{j=m-1}^{2m-2} h^{j+1-m} k^{m-1}}\right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m} k^{m-1}}} \\
&= \left(\frac{f(k+1)}{f(1)}\right)^{\frac{m-2}{m-2+\frac{1-h^{m+1}}{1-h}}} \left(\frac{g(h+1)}{g(1)}\right)^{\frac{\frac{1-h^m}{1-h}}{m-2+\frac{1-h^{m+1}}{1-h}}} \\
&\geq \lim_{m\to\infty} \left(\frac{f(k+1)}{f(1)}\right)^{\frac{m-2}{m-2+\frac{1-h^{m+1}}{1-h}}} \left(\frac{g(h+1)}{g(1)}\right)^{\frac{\frac{1-h^m}{1-h}}{m-2+\frac{1-h^{m+1}}{1-h}}} - \epsilon && (50)\\
&= \left(\frac{f(k+1)}{f(1)}\right)^{1} \left(\frac{g(h+1)}{g(1)}\right)^{0} - \epsilon \\
&= \left(\frac{f(k+1)}{f(1)}\right)^{\frac{1-h}{k-h}} \left(\frac{g(h+1)}{g(1)}\right)^{\frac{k-1}{k-h}} - \epsilon && (51)\\
&> M + \epsilon - \epsilon && (52)\\
&= M, && (53)
\end{aligned}
$$

where (50) holds if m is sufficiently large, (51) comes from the fact that $k = 1$ and $h < 1$, and (52) comes from (39). By (49) and (53), we have that, for a sufficiently large m, $\mathsf{NPoA}(\mathsf{LB}(m, s_m)) \geq M$, thus showing part (ii) of the claim.

If $h = 0$, then we consider a load-balancing game defined as $\mathsf{LB}(m, s_m)$, but restricted to the resources of groups $R_1, \ldots, R_m$ and to the players of groups $N_1, \ldots, N_{m-1}$. By using the same proof arguments as those used for $h > 0$, one can show the claim as well.

We now show part (i). Assume that C is abscissa-scaling and ordinate-scaling. Analogously to the proof of part (ii), we have that (39) holds. Moreover, let $\mathsf{LB}'(m, s)$ be a weighted load-balancing game equal to game $\mathsf{LB}(m, s)$ defined in the proof of part (ii), except for the strategy set of each player: For any $j \in [2m-1]$, the strategy set of each player of group N_j is $\Sigma_j := R_j \cup R_{j+1}$.

Let $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ be the strategy profiles defined as in game $\mathsf{LB}(m, s)$. By considering the same proof arguments of Lemma A.2, one can show that, for any $j \in [2m-1]$, each player of group N_j does not reduce her cost when deviating unilaterally from $\boldsymbol{\sigma}$ to any resource $r \in R_{j+1}$; we observe that the analysis of these restricted deviations does not require that $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$, differently from the symmetric instance of part (ii) in which all the deviations are possible. Thus, $\boldsymbol{\sigma}$ is a pure Nash equilibrium of $\mathsf{LB}'(m, s)$ for any $s \geq 1$.

To conclude, if we take a sufficiently large m, an arbitrary $s \geq 1$, and by applying to game $\mathsf{LB}'(m, s)$ the same inequalities as in (49) and (53), then part (i) of the claim follows.

We observe that, for $s = 1$, the obtained instance is equivalent to the load-balancing game $\mathsf{LB}(m)$ analyzed in the sketch of the proof given in Section 3.1 to show part (i) of this theorem.

A.3 Proof of Corollary 3.6

Let $\epsilon > 0$. Let $\mathsf{LB}'(m)$ be the load-balancing game defined as the game $\mathsf{LB}'(m,s)$ considered in the proof of part (i) of Theorem 3.3 (last paragraph of Section A.2), with $s = 2$, $k = 1$, $h = 0$, and f, g defined as $f(x) = g(x) = x^p$. One can easily observe that $\mathsf{LB}'(m)$ is a game with identical resources. Furthermore, because of the proof of Theorem 3.3, there exists a sufficiently large integer m such that $\mathsf{NPoA}(\mathsf{LB}'(m)) > 2^p - \epsilon$, and the claim follows by the arbitrariness of $\epsilon > 0$.

A.4 Proof of Theorem 3.8 (Full Version)

To prove the theorem, we equivalently show that, for any $M < \xi(C)$, there exists a game $\mathsf{LB} \in \mathsf{ULB}(C)$ such that $\mathsf{NPoA}(\mathsf{LB}) > M$.

Fix an arbitrary $M < \xi(C)$. Let $f \in C$, $k \in \mathbb{N}$, $o \in [k]$, and a sufficiently small $\epsilon > 0$ such that

$$\left(\frac{f(k+1)}{f(o)}\right)^{\frac{o}{k}} > M + \epsilon. \tag{54}$$

Given an integer $m > 0$, let $\mathsf{LB}(m)$ be an unweighted load-balancing game with $(k-o+1)m+o$ resources, partitioned into m groups $R_1, R_2, \ldots, R_m$ such that $R_j := \{r_{j,0}, r_{j,1}, \ldots, r_{j,k-o}\}$ for any $j \in [m-1]$, and $R_m := \{r_{m,0}, r_{m,1}, \ldots, r_{m,k}\}$. Each resource $r_{j,h}$ has latency function $\ell_{r_{j,h}}(x) := \alpha_{j,h} f(x)$, with

$$\alpha_{j,h} := \begin{cases} \left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{if } h = 0 \\ \frac{f(k)}{f(1)}\left(\frac{f(k)}{f(k+1)}\right)^{j-1} & \text{otherwise.} \end{cases}$$

We have $n := mk$ players split into m groups $N_1, N_2, \ldots, N_m$ of k players each. For $j \in [m-1]$, the set of strategies Σ_j of players of group N_j is $R_j \cup \{r_{j+1,0}\}$, and the set of strategies Σ_m of players in N_m is R_m. We observe that all the latency functions of $\mathsf{LB}(m)$ belong to C, as C is ordinate scaling.

Let $\boldsymbol{\sigma}$ be the strategy profile such that, for any $j \in [m]$, all k players of group N_j select resource $r_{j,0}$, so each resource $r_{j,0}$ has congestion k, and all the remaining resources have null congestion (see Figure 1(a)). We show that $\boldsymbol{\sigma}$ is a pure Nash equilibrium. Given an arbitrary player i of group N_j with $j \in [m]$, such player has a cost equal to $\ell_{r_{j,0}}(k) = \alpha_{j,0} f(k) = (\frac{f(k)}{f(k+1)})^{j-1} f(k)$ when playing strategy σ_i. If $j \in [m-1]$, and player i unilaterally deviates to strategy $r_{j+1,0}$, then her cost is $\ell_{r_{j+1,0}}(k+1) = \alpha_{j+1,0} f(k+1) = (\frac{f(k)}{f(k+1)})^j f(k+1) = (\frac{f(k)}{f(k+1)})^{j-1} f(k) = \ell_{r_{j,0}}(k)$, thus her cost does not improve. Analogously, if $j \in [m]$, and player i unilaterally deviates to any strategy $r_{j,h}$ with $h \neq 0$, then her cost is $\ell_{r_{j,h}}(1) = \alpha_{j,h} f(1) = \frac{f(k)}{f(1)}(\frac{f(k)}{f(k+1)})^{j-1} f(1) = \ell_{r_{j,0}}(k)$, thus her cost does not improve as well. We conclude that $\boldsymbol{\sigma}$ is a pure Nash equilibrium.

Now, let $\boldsymbol{\sigma}^*$ be a strategy profile defined as follows: (i) for any $j \in [m-1]$, o players of group N_j select resource $r_{j+1,0}$, and each of the $k-o$ remaining players of N_j selects a distinct resource of $R_j \setminus \{r_{j,0}\}$, (ii) all the k players of group N_m select a distinct resource of $R_m \setminus \{r_{m,0}\}$. Thus, in $\boldsymbol{\sigma}^*$, any resource of type $r_{j,0}$ with $j > 1$ has congestion o, resource $r_{1,0}$ has null congestion, and the remaining resources have unitary congestion (see Figure 1(b)).

By some algebraic manipulation, it holds that

$$\frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)}$$

$$= \left(\frac{\prod_{j=1}^{m} \ell_{j,0}(k)^k}{\prod_{j=1}^{m-1}\left(\ell_{r_{j+1,0}}(o)^o \prod_{r \in R_j \setminus \{r_{j,0}\}} \ell_r(1)\right) \prod_{r \in R_m \setminus \{r_{m,0}\}} \ell_r(1)}\right)^{\frac{1}{km}}$$

$$= \left(\frac{\prod_{j=1}^{m} \left(\left(\frac{f(k)}{f(k+1)} \right)^{j-1} f(k) \right)^{k}}{\prod_{j=1}^{m-1} \left[\left(\left(\frac{f(k)}{f(k+1)} \right)^{j} f(o) \right)^{o} \left(\frac{f(k)}{f(1)} \left(\frac{f(k)}{f(k+1)} \right)^{j-1} f(1) \right)^{k-o} \right] \left(\frac{f(k)}{f(1)} \left(\frac{f(k)}{f(k+1)} \right)^{m-1} f(1) \right)^{k}} \right)^{\frac{1}{km}}$$

$$= \left(\frac{\prod_{j=1}^{m} \left(\left(\frac{f(k)}{f(k+1)} \right)^{j} f(k+1) \right)^{k}}{\prod_{j=1}^{m-1} \left[\left(\left(\frac{f(k)}{f(k+1)} \right)^{j} f(o) \right)^{o} \left(\left(\frac{f(k)}{f(k+1)} \right)^{j} f(k+1) \right)^{k-o} \right] \left(\left(\frac{f(k)}{f(k+1)} \right)^{m} f(k+1) \right)^{k}} \right)^{\frac{1}{km}}$$

$$= \left(\frac{\left(\prod_{j=1}^{m} \left(\frac{f(k)}{f(k+1)} \right)^{kj} \right) f(k+1)^{km}}{\left(\prod_{j=1}^{m-1} \left(\frac{f(k)}{f(k+1)} \right)^{kj} \right) f(o)^{o(m-1)} f(k+1)^{(k-o)(m-1)} \left(\frac{f(k)}{f(k+1)} \right)^{km} f(k+1)^{k}} \right)^{\frac{1}{km}}$$

$$= \left(\frac{f(k+1)^{km}}{f(o)^{o(m-1)} f(k+1)^{(k-o)(m-1)} f(k+1)^{k}} \right)^{\frac{1}{km}}$$

$$= \left(\frac{f(k+1)^{o(m-1)}}{f(o)^{o(m-1)}} \right)^{\frac{1}{km}}$$

$$= \left(\frac{f(k+1)}{f(o)} \right)^{\frac{o(m-1)}{km}} . \tag{55}$$

By using (54) and (55), and by choosing a sufficiently large m, we get

$$\mathsf{NPoA}(\mathsf{LB}(m)) \geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)} = \left(\frac{f(k+1)}{f(o)} \right)^{\frac{o(m-1)}{km}} \geq \lim_{m \to \infty} \left(\frac{f(k+1)}{f(o)} \right)^{\frac{o(m-1)}{km}} - \epsilon$$
$$= \left(\frac{f(k+1)}{f(o)} \right)^{\frac{o}{k}} - \epsilon > M + \epsilon - \epsilon = M,$$

thus showing the claim.

A.5 Proof of Corollary 3.9

The claim follows from the following lemma:

Lemma A.3. $\xi(\mathcal{P}(p)) = 2^p$.

Proof. We have that

$$\xi(\mathcal{P}(p)) = \sup_{f \in \mathcal{P}(p), k \in \mathbb{N}, o \in [k]} \left(\frac{f(k+1)}{f(o)} \right)^{\frac{o}{k}}$$
$$= \sup_{\alpha_0, \alpha_1, \ldots, \alpha_p \geq 0, k \in \mathbb{N}, o \in [k]} \left(\frac{\sum_{d=0}^{p} \alpha_d (k+1)^d}{\sum_{d=0}^{p} \alpha_d o^d} \right)^{\frac{o}{k}}$$
$$= \sup_{k \in \mathbb{N}, o \in [k]} \left(\max_{d \in [p] \cup \{0\}} \frac{(k+1)^d}{o^d} \right)^{\frac{o}{k}}$$
$$= \sup_{k \in \mathbb{N}, o \in [k]} \left(\left(\frac{k+1}{o} \right)^{p} \right)^{\frac{o}{k}}$$

$$= \left(\sup_{k \in \mathbb{N}, o \in [k]} \left(\frac{k+1}{o} \right)^{\frac{o}{k}} \right)^p$$
$$= 2^p, \tag{56}$$

where (56) holds for the following reasons: First, we have that $(\frac{k+1}{o})^{\frac{o}{k}} = 2$ if $o = k = 1$, thus showing that $2 \leq \sup_{k \in \mathbb{N}, o \in [k]} (\frac{k+1}{o})^{\frac{o}{k}}$; furthermore, by setting $x := k/o$, we obtain $(\frac{k+1}{o})^{\frac{o}{k}} = (x + \frac{1}{o})^{\frac{1}{x}} \leq (x+1)^{\frac{1}{x}} \leq 2$, where the last inequality is equivalent to the well-known inequality $2^x \geq x + 1$, which holds for any $x \geq 1$. □

A.6 Proof of Theorem 4.3

Let us assume that C is abscissa-scaling and ordinate-scaling. We equivalently show that for any $M < \zeta(C)$ there exists an instance $\mathsf{I} \in \mathsf{WLB}(C)$ such that $\mathsf{NPoA}(\mathsf{I}) > M$.

Let $f_1, f_2 \in C, k_1, k_2, o_1, o_2 \geq 0$ such that $k_1 \geq o_1 > 0, o_2 > k_2 \geq 0$, and let $\epsilon > 0$ be a sufficiently small number such that $(\frac{f_1(k_1+o_1)^{k_1+o_1}}{f_1(k_1)^{k_1} f_1(o_1)^{o_1}})^{\frac{o_2-k_2}{o_2k_1-o_1k_2}} (\frac{f_2(k_2+o_2)^{k_2+o_2}}{f_2(k_2)^{k_2} f_2(o_2)^{o_2}})^{\frac{k_1-o_1}{o_2k_1-o_1k_2}} > M + \epsilon$. Let $f, g \in C$ be such that $f(x) := f_1(o_1 x)$ and $g(x) := f_2(o_2 x)$, and let $k := k_1/o_1$ and $h := k_2/o_2$ two positive real numbers. Since

$$\left(\frac{f_1(k_1+o_1)^{k_1+o_1}}{f_1(k_1)^{k_1} f_1(o_1)^{o_1}} \right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}} \left(\frac{f_2(k_2+o_2)^{k_2+o_2}}{f_2(k_2)^{k_2} f_2(o_2)^{o_2}} \right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}} = \left(\frac{f(k+1)^{k+1}}{f(k)^k f(1)} \right)^{\frac{1-h}{k-h}} \left(\frac{g(h+1)^{h+1}}{g(h)^h g(1)} \right)^{\frac{k-1}{k-h}},$$

we have that

$$\left(\frac{f(k+1)^{k+1}}{f(k)^k f(1)} \right)^{\frac{1-h}{k-h}} \left(\frac{g(h+1)^{h+1}}{g(h)^h g(1)} \right)^{\frac{k-1}{k-h}} > M + \epsilon, \text{ for some } f, g \in C,\ k \geq 1, \text{ and } h < 1. \tag{57}$$

First, we assume that $h > 0$. Given an integer $m \geq 3$, let $\mathsf{I}(m)$ be a load-balancing instance having $2m$ resources $r_1, r_2, r_3 \ldots, r_{2m}$ and $2m - 1$ clients such that the set of strategies of each client j is $\{r_j, r_{j+1}\}$. Each resource r_j has a latency function defined as $\ell_j(x) := \alpha_j \hat{f}_j(\beta_j x)$, and the weight of each client j is defined as $w_j := 1/\beta_{j+1}$, where α_j, $\hat{f}_j$, and β_j are defined as follows:

$$\hat{f}_j := \begin{cases} f & \text{if } j \leq m-1 \\ g & \text{if } j \geq m \end{cases}, \quad \beta_j := \begin{cases} \left(\frac{1}{k}\right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{1}{h}\right)^{j-m} \left(\frac{1}{k}\right)^{m-1} & \text{if } m \leq j \leq 2m \end{cases}, \tag{58}$$

$$\alpha_j := \begin{cases} \left(\frac{f(k)^{k+1}}{f(k+1)^{k+1}} \right)^{j-1} & \text{if } j \leq m-1 \\ \left(\frac{g(h)^{h+1}}{g(h+1)^{h+1}} \right)^{j-m} \left(\frac{f(k)g(h)^h}{g(h+1)^{h+1}} \right) \left(\frac{f(k)^{k+1}}{f(k+1)^{k+1}} \right)^{m-2} & \text{if } m \leq j \leq 2m-1 \\ \frac{g(h)}{g(1)} \left(\frac{g(h)^{h+1}}{g(h+1)^{h+1}} \right)^{m-1} \left(\frac{f(k)g(h)^h}{g(h+1)^{h+1}} \right) \left(\frac{f(k)^{k+1}}{f(k+1)^{k+1}} \right)^{m-2} & \text{if } j = 2m \end{cases}. \tag{59}$$

We observe that all the latency functions of $\mathsf{I}(m)$ belong to C, as C is abscissa-scaling and ordinate-scaling. By construction of α_j, β_j, w_j, the following properties hold:

$$\begin{cases} \alpha_j f(k) = \alpha_{j+1} \frac{f(k+1)^{k+1}}{f(k)^k} & \text{if } j \leq m-2 \\ \alpha_j f(k) = \alpha_{j+1} \frac{g(h+1)^{h+1}}{g(h)^h} & \text{if } j = m-1 \\ \alpha_j g(h) = \alpha_{j+1} \frac{g(h+1)^{h+1}}{g(h)^h} & \text{if } m \leq j \leq 2m-2 \\ \alpha_j g(h) = \alpha_{j+1} g(1) & \text{if } j = 2m-1 \end{cases}, \begin{cases} \beta_j w_j = k,\ w_j = k^j & \text{if } j \leq m-1 \\ \beta_j w_j = h,\ w_j = h^{j+1-m} k^{m-1} & \text{if } m \leq j \leq 2m-1 \\ \beta_{j+1} w_j = 1 & \text{if } j \leq 2m-1 \end{cases}. \tag{60}$$

Let $\boldsymbol{\sigma}$ be the strategy profile in which each client j is assigned to resource r_j. We show that $\boldsymbol{\sigma}$ is a state that can be possibly returned by the greedy algorithm when clients are processed in reverse order w.r.t. index j. We equivalently show that $\frac{\mathsf{NSW}(\boldsymbol{\sigma}^j)}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})} \leq \frac{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1}, r_{j+1})}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})}$ for any $j \leq 2m-1$, where $\boldsymbol{\sigma}^j$ denotes the partial assignment in which each client $t \geq j$ is assigned to resource r_t, and $(\boldsymbol{\sigma}^{j+1}, r_{j+1})$ denotes the partial assignment in which each client $t \geq j+1$ is assigned to resource r_t and client j is assigned to resource r_{j+1}. Let $j \in [2m-1]$. First, assume that $j \leq m-2$. By using (60), we get

$$\begin{aligned}
\frac{\mathsf{NSW}(\boldsymbol{\sigma}^j)}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})} &= \ell_{r_j}(k_{r_j}(\boldsymbol{\sigma}))^{k_{r_j}(\boldsymbol{\sigma})} = \left(\alpha_j \hat{f}_j\left(\beta_j w_j\right)\right)^{w_j} = (\alpha_j f(k))^{w_j} = \left(\alpha_{j+1}\frac{f(k+1)^{k+1}}{f(k)^k}\right)^{w_j} \\
&= \alpha_{j+1}^{w_j}\frac{f(k+1)^{kw_j+w_j}}{f(k)^{kw_j}} = \alpha_{j+1}^{w_j}\frac{f(k+1)^{w_{j+1}+w_j}}{f(k)^{w_{j+1}}} = \frac{\left(\alpha_{j+1}f(k+1)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}f(k)\right)^{w_{j+1}}} \\
&= \frac{\left(\alpha_{j+1}f\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}f(\beta_{j+1}w_{j+1})\right)^{w_{j+1}}} = \frac{\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}w_{j+1})\right)^{w_{j+1}}} \\
&= \frac{\ell_{r_{j+1}}(k_{r_{j+1}}(\boldsymbol{\sigma}^{j+1}, r_{j+1}))^{k_{r_{j+1}}(\boldsymbol{\sigma}^{j+1}, r_{j+1})}}{\ell_{r_{j+1}}(k_{r_{j+1}}(\boldsymbol{\sigma}^{j+1}))^{k_{r_{j+1}}(\boldsymbol{\sigma}^{j+1})}} = \frac{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1}, r_{j+1})}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})}.
\end{aligned}$$

The cases $j = m-1$, $m \leq j \leq 2m-2$, and $j = 2m-1$ can be separately considered by exploiting (60), so one can analogously get

$$\frac{\mathsf{NSW}(\boldsymbol{\sigma}^j)}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})} = \left(\alpha_j \hat{f}_j\left(\beta_j w_j\right)\right)^{w_j} = \frac{\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}w_{j+1})\right)^{w_{j+1}}} = \frac{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1}, r_{j+1})}{\mathsf{NSW}(\boldsymbol{\sigma}^{j+1})}, \quad (61)$$

where we set $(\alpha_{2m}\hat{f}_{2m}(\beta_{2m}w_{2m}))^{w_{2m}} := 1$ and $w_{2m} := 0$. Now, let $\boldsymbol{\sigma}^*$ be the strategy profile of $\mathsf{I}(m)$ in which each client $j \in [m-1]$ is assigned to resource r_{j+1}. By exploiting the definitions of α_j,β_j, $\hat{f}_j$, and w_j, and by considering a sufficiently large m, we have that:

$$\begin{aligned}
&\mathsf{NPoA}(\mathsf{I}(m)) \\
&\geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)} \\
&= \left(\frac{\prod_{j=1}^{2m-1}\left(\alpha_j\hat{f}_j\left(\beta_j w_j\right)\right)^{w_j}}{\prod_{j=2}^{2m}\left(\alpha_j\hat{f}_j\left(\beta_j w_{j-1}\right)\right)^{w_{j-1}}}\right)^{\frac{1}{\sum_{j=1}^{2m-1} w_j}} \\
&= \left(\frac{\prod_{j=1}^{2m-1}\left(\frac{\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}w_{j+1})\right)^{w_{j+1}}}\right)}{\prod_{j=2}^{2m}\left(\alpha_j\hat{f}_j\left(\beta_j w_{j-1}\right)\right)^{w_{j-1}}}\right)^{\frac{1}{\sum_{j=1}^{2m-1} w_j}} \quad (62) \\
&= \left(\frac{\prod_{j=1}^{2m-1}\left(\frac{\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}w_{j+1})\right)^{w_{j+1}}}\right)}{\prod_{j=1}^{2m-1}\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}w_j\right)\right)^{w_j}}\right)^{\frac{1}{\sum_{j=1}^{2m-1} w_j}}
\end{aligned}$$

$$= \left(\prod_{j=1}^{2m-1} \left(\frac{\left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}(w_{j+1}+w_j)\right)\right)^{w_{j+1}+w_j}}{\left(\alpha_{j+1}\hat{f}_{j+1}(\beta_{j+1}w_{j+1})\right)^{w_{j+1}} \left(\alpha_{j+1}\hat{f}_{j+1}\left(\beta_{j+1}w_j\right)\right)^{w_j}} \right) \right)^{\frac{1}{\sum_{j=1}^{2m-1} w_j}}$$

$$= \left(\prod_{j=1}^{m-2} \left(\frac{f(k+1)^{k^{j+1}+k^j}}{f(k)^{k^{j+1}} f(1)^{k^j}} \right) \prod_{j=m-1}^{2m-2} \left(\frac{g(h+1)^{h^{j+2-m}k^{m-1}+h^{j+1-m}k^{m-1}}}{g(h)^{h^{j+2-m}k^{m-1}} g(1)^{h^{j+1-m}k^{m-1}}} \right) \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m}k^{m-1}}}$$

$$= \left(\prod_{j=1}^{m-2} \left(\frac{f(k+1)^{k+1}}{f(k)^k f(1)} \right)^{k^j} \prod_{j=m-1}^{2m-2} \left(\frac{g(h+1)^{h+1}}{g(h)^h g(1)} \right)^{h^{j+1-m}k^{m-1}} \right)^{\frac{1}{\sum_{j=1}^{m-2} k^j + \sum_{j=m-1}^{2m-1} h^{j+1-m}k^{m-1}}}$$

$$\geq \left(\frac{f(k+1)^{k+1}}{f(k)^k f(1)} \right)^{\frac{1-h}{k-h}} \left(\frac{g(h+1)^{h+1}}{g(h)^h g(1)} \right)^{\frac{k-1}{k-h}} - \epsilon \tag{63}$$

$$> M + \epsilon - \epsilon \tag{64}$$

$$= M, \tag{65}$$

where (62) comes from (61), (63) can be shown by using similar arguments as in the proof of Theorem 3.3 (see steps (47) and (51)), and (64) comes from (57). By (65), the claim follows.

If $h = 0$, then we consider a load-balancing instance defined as $\mathsf{I}(m)$, but restricted to resources $r_1, r_2, \ldots, r_m$ and to players in $[m-1]$. By using the same proof arguments as those used for $h > 0$, one can show the claim as well.

A.7 Proof of Corollary 4.4

The proof follows from the following lemma:

Lemma A.4. $\zeta(\mathcal{P}(p)) = 4^p$.

Proof. We have that

$$\zeta(\mathcal{P}(p)) \tag{66}$$

$$= \sup_{k_1 \geq o_1 > 0, o_2 > k_2 \geq 0, f_1, f_2 \in C} \left(\frac{f_1(k_1+o_1)^{k_1+o_1}}{f_1(k_1)^{k_1} f_1(o_1)^{o_1}} \right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}} \left(\frac{f_2(k_2+o_2)^{k_2+o_2}}{f_2(k_2)^{k_2} f_2(o_2)^{o_2}} \right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}}$$

$$= \sup_{\substack{k_1 \geq o_1 > 0,\\ o_2 > k_2 \geq 0,\\ \alpha_0,\ldots,\alpha_p \geq 0,\\ \beta_0,\ldots,\beta_p \geq 0}} \left(\frac{\left(\sum_{d=0}^p \alpha_d (k_1+o_1)^d\right)^{k_1+o_1}}{\left(\sum_{d=0}^p \alpha_d k_1^d\right)^{k_1} \left(\sum_{d=0}^p \alpha_d o_1^d\right)^{o_1}} \right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}} \left(\frac{\left(\sum_{d=0}^p \beta_d (k_2+o_2)^d\right)^{k_2+o_2}}{\left(\sum_{d=0}^p \beta_d k_2^d\right)^{k_2} \left(\sum_{d=0}^p \beta_d o_2^d\right)^{o_2}} \right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}}$$

$$= \sup_{\substack{k_1 \geq o_1 > 0, o_2 > k_2 \geq 0,\\ \alpha_0,\ldots,\alpha_p, \beta_0,\ldots,\beta_p \geq 0}} \left(\left(\frac{\sum_{d=0}^p \alpha_d (k_1+o_1)^d}{\sum_{d=0}^p \alpha_d k_1^d} \right)^{k_1} \left(\frac{\sum_{d=0}^p \alpha_d (k_1+o_1)^d}{\sum_{d=0}^p \alpha_d o_1^d} \right)^{o_1} \right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}}$$

$$\cdot \left(\left(\frac{\sum_{d=0}^p \beta_d (k_2+o_2)^d}{\sum_{d=0}^p \beta_d k_2^d} \right)^{k_2} \left(\frac{\sum_{d=0}^p \beta_d (k_2+o_2)^d}{\sum_{d=0}^p \beta_d o_2^d} \right)^{o_2} \right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}}$$

$$= \sup_{k_1 \geq o_1 > 0, o_2 > k_2 \geq 0} \left(\left(\max_{d \in [p] \cup \{0\}} \frac{(k_1+o_1)^d}{k_1^d} \right)^{k_1} \left(\max_{d \in [p] \cup \{0\}} \frac{(k_1+o_1)^d}{o_1^d} \right)^{o_1} \right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}}$$

$$\cdot\left(\left(\max_{d\in[p]\cup\{0\}}\frac{(k_2+o_2)^d}{k_2^d}\right)^{k_2}\left(\max_{d\in[p]\cup\{0\}}\frac{(k_2+o_2)^d}{o_2^d}\right)^{o_2}\right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}}$$

$$=\sup_{\substack{k_1\geq o_1>0,\\ o_2>k_2\geq 0}}\left(\left(\frac{(k_1+o_1)^p}{k_1^p}\right)^{k_1}\left(\frac{(k_1+o_1)^p}{o_1^p}\right)^{o_1}\right)^{\frac{o_2-k_2}{o_2k_1-o_1k_2}}\left(\left(\frac{(k_2+o_2)^p}{k_2^p}\right)^{k_2}\left(\frac{(k_2+o_2)^p}{o_2^p}\right)^{o_2}\right)^{\frac{k_1-o_1}{o_2k_1-o_1k_2}}$$

$$=\sup_{k\geq 1,0\leq h<1}\left(\left(\frac{(k+1)^{k+1}}{k^k}\right)^{\frac{1-h}{k-h}}\left(\frac{(h+1)^{h+1}}{h^h}\right)^{\frac{k-1}{k-h}}\right)^p, \tag{67}$$

where (67) can be obtained by setting $k := k_1/o_1$ and $h := k_2/o_2$ (k and h are two non-negative real numbers). Now, we show that the maximum value of function $F(k,h) := (\frac{(k+1)^{k+1}}{k^k})^{\frac{1-h}{k-h}}(\frac{(h+1)^{h+1}}{h^h})^{\frac{k-1}{k-h}}$ over $k\geq 1$ and $0\leq h<1$ is equal to 4. Observe that $\ln(F(k,h)) = \frac{1-h}{k-h}((k+1)\ln(k+1) - k\ln(k)) + \frac{k-1}{k-h}((h+1)\ln(h+1) - h\ln(h)) \leq (\frac{1-h}{k-h}(k+1) + \frac{k-1}{k-h}(h+1))\ln(\frac{1-h}{k-h}(k+1) + \frac{k-1}{k-h}(h+1))$, where the second last inequality holds because of the concavity of the function g defined as $g(x) := (x+1)\ln(x+1) - x\ln(x)$ and since $\ln(F(k,h))$ is defined as convex combination of $g(k)$ and $g(h)$. Thus, we get

$$\begin{aligned}F(k,h) &\leq \left(\frac{1-h}{k-h}(k+1)+\frac{k-1}{k-h}(h+1)\right)^{\frac{1-h}{k-h}(k+1)+\frac{k-1}{k-h}(h+1)}\\ &= \left(\frac{(k-h)+(k-h)}{k-h}\right)^{\frac{(k-h)+(k-h)}{k-h}} = 2^2 = 4.\end{aligned} \tag{68}$$

Finally, since $F(k,h) = 4$ for $k = 1$ and $h = 0$, and because of (68), we have that the maximum of $F(k,h)$ over $k\geq 1$ and $0\leq h<1$ is 4. Thus, we get that (67) is at most 4^p. □

A.8 Proof of Theorem 4.5 (Last Part)

To complete the proof of Theorem 4.5, it is sufficient showing that, for any $\epsilon > 0$, there exists a sufficiently large $m\geq 1$ such that $\mathsf{CR}_{\mathsf{A}}(\mathsf{I}(m)) \geq 4^p - \epsilon$.

Given an arbitrary integer $m \geq 1$, let $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ be the states of $\mathsf{I}(m)$ in which each client is assigned to her first and second resource, respectively. We have that $\boldsymbol{\sigma}$ is a state that can be returned by any online algorithm if clients are processed according to the following partial ordering: (i) given two clients i_1 and i_2 having their first resource in sub-instances of type $\mathsf{I}(m_1)$ and $\mathsf{I}(m_2)$, respectively, if $m_1 < m_2$, then client i_1 is processed before client i_2; (ii) the clients defined in the same sub-instance are processed in increasing order with respect to their weights. This fact is true, since each time the greedy algorithm processes some client i according to the partial ordering defined above, the congestions of the first and the second resource of that client are equal. Thus, since the latency functions are equal too, any online algorithm cannot distinguish between the two resources selectable by each client, and by symmetry, both choices can potentially lead to the same worst-case competitive ratio.

We have the following fact:

Fact 2. *Given two integers $m\geq 1$ and $i\in[m-1]\cup\{0\}$ such that $j\geq i$, the number $N(m,i)$ of sub-instances of $\mathsf{I}(m)$ equivalent to $\mathsf{I}(j)$ for some $j\geq i$ is $N(m,i) = 2^{m-i}$.*

Proof. We show the claim by induction on $h(i) := m - i \geq 0$. If $h(i) = 0$, then the unique sub-instance equivalent to $\mathsf{I}(j)$ for some $j\geq i$ is the entire instance $\mathsf{I}(m)$, thus $N(m,i) = 1 = 2^{h(i)} = 2^{m-i}$ and the base step holds. Now, assume that the claim holds for any $h(i)\geq 0$. Observe that we can associate in a one-to-one correspondence each sub-instance that is equivalent to $\mathsf{I}(j)$ for some $j \geq i$, with a sub-instance equivalent to $\mathsf{I}(i-1)$, that is $N(m,i) = N(m,i-1) - N(m,i) \Rightarrow N(m,i-1) = 2N(m,i)$. Thus, we have that $N(m,i-1) = 2N(m,i) = 2 \cdot 2^{h(i)} = 2^{m-i+1} = 2^{h(i)+1}$, and the inductive step holds. □

Let $N(m,i)$ be defined as in Fact 2 and let $R(i)$ be the set of fundamental resources for sub-instances of type $\mathsf{I}(i)$. Observe that, for any $i \in [m]$ and resource r such that i clients select r as first resource, r is the fundamental resource of a sub-instance of type $\mathsf{I}(i)$, i.e., $r \in R(i)$. Thus, by exploiting Fact 2, we get

$$\begin{aligned}
\mathsf{NSW}(\boldsymbol{\sigma}) &= \left(\prod_{i\in[m]}\prod_{r\in R(i)} \ell(k_r(\boldsymbol{\sigma}))^{k_r(\boldsymbol{\sigma})}\right)^{\frac{1}{\sum_{r\in R} k_r(\boldsymbol{\sigma})}}\\
&= \left(\prod_{i\in[m]} \ell\left(\sum_{j=1}^{i} w_j\right)^{\left(\sum_{j=1}^{i} w_j\right)|R(i)|}\right)^{\frac{1}{\sum_{i\in[m]}\left(\sum_{j=1}^{i} w_j\right)|R(i)|}}\\
&= \left(\prod_{i\in[m]} \ell\left(\sum_{j=1}^{i} 2^{j-1}\right)^{\left(\sum_{j=1}^{i} 2^{j-1}\right)(N(m,i)-N(m,i+1))}\right)^{\frac{1}{\sum_{i\in[m]}\left(\sum_{j=1}^{i} 2^{j-1}\right)(N(m,i)-N(m,i+1))}}\\
&= \left(\prod_{i\in[m]} \left(2^i-1\right)^{p\left(2^i-1\right)2^{m-i-1}}\right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}} \qquad (69)
\end{aligned}$$

and

$$\begin{aligned}
\mathsf{NSW}(\boldsymbol{\sigma}^*) &= \left(\prod_{i\in[m]}\prod_{r\in R(i)}\prod_{j\in[i]} \ell(k_{r(j)}(\boldsymbol{\sigma}^*))^{k_{r(j)}(\boldsymbol{\sigma}^*)}\right)^{\frac{1}{\sum_{r\in R} k_r(\boldsymbol{\sigma})}}\\
&= \left(\prod_{i\in[m]}\prod_{j\in[i]} \ell\left(w_j\right)^{w_j(N(m,i)-N(m,i+1))}\right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}}\\
&= \left(\prod_{i\in[m]}\prod_{j\in[i]} \left(2^{j-1}\right)^{p2^{j-1}2^{m-i-1}}\right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}}\\
&= \left(\prod_{i\in[m]} 2^{p\left(\sum_{j=0}^{i-1} j2^j\right)2^{m-i-1}}\right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}}\\
&= \left(\prod_{i\in[m]} 2^{p\left(i2^i-2\left(2^i-1\right)\right)2^{m-i-1}}\right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}}. \qquad (70)
\end{aligned}$$

Let $\epsilon > 0$. By (69) and (70), and by taking a sufficiently large integer $m > 1$, we get

$$\mathsf{CR}_{\mathsf{A}}(\mathsf{I}(m)) \geq \frac{\mathsf{NSW}(\boldsymbol{\sigma})}{\mathsf{NSW}(\boldsymbol{\sigma}^*)}$$

$$= \left(\frac{\prod_{i\in[m]} \left(2^i - 1\right)^{p\left(2^i-1\right)2^{m-i-1}}}{\prod_{i\in[m]} 2^{p(i2^i-2(2^i-1))2^{m-i-1}}} \right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{m-i-1}}}$$

$$= \left(\frac{\prod_{i\in[m]} \left(2^i - 1\right)^{p\left(2^i-1\right)2^{-i-1}}}{\prod_{i\in[m]} 2^{p(i2^i-2(2^i-1))2^{-i-1}}} \right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{-i-1}}}$$

$$= \left(\frac{\prod_{i\in[m]} \left(2^i\right)^{p\left(2^i-1\right)2^{-i-1}}}{\prod_{i\in[m]} 2^{p(i2^i-2(2^i-1))2^{-i-1}}} \right)^{\frac{1}{\sum_{i\in[m]}(2^i-1)2^{-i-1}}} \prod_{i\in[m]} \left(\frac{2^i-1}{2^i}\right)^{\frac{p\left(2^i-1\right)2^{-i-1}}{\sum_{i\in[m]}(2^i-1)2^{-i-1}}}$$

$$= \left(2^p\right)^{\frac{\sum_{i\in[m]}\left(-i2^{-i-1}+1-2^{-i}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} \prod_{i\in[m]} \left(\frac{2^i-1}{2^i}\right)^{\frac{p\left(1/2-2^{-i-1}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}}. \tag{71}$$

We have the following fact:

Fact 3.

$$\lim_{m\to\infty} \prod_{i\in[m]} \left(\frac{2^i-1}{2^i}\right)^{\frac{p\left(1/2-2^{-i-1}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} = 1.$$

Proof. Set $\alpha_i := p\ln(\frac{2^i-1}{2^i})$ and $\beta_i := (1/2 - 2^{-i-1})$. We will equivalently show that $\lim_{m\to\infty} \frac{\sum_{i=1}^m \alpha_i\beta_i}{\sum_{i=1}^m \beta_i} = 0$, since, by exponentiating this equality, we get the claim. Set $a_m := \sum_{i=1}^m \alpha_i\beta_i$ and $b_m := \sum_{i=1}^m \beta_i$. We have that sequence $(b_m)_{m\geq 1}$ is positive, increasing, and unbounded. Thus, by the Stolz-Cesaro Theorem, we have that $\lim_{m\to\infty} \frac{a_m}{b_m} = \lim_{m\to\infty} \frac{a_{m+1}-a_m}{b_{m+1}-b_m}$. We conclude that $\lim_{m\to\infty} \frac{\sum_{i=1}^m \alpha_i\beta_i}{\sum_{i=1}^m \beta_i} = \lim_{m\to\infty} \frac{a_m}{b_m} = \lim_{m\to\infty} \frac{a_{m+1}-a_m}{b_{m+1}-b_m} = \lim_{m\to\infty} \frac{\alpha_m\beta_m}{\beta_m} = \lim_{m\to\infty} p\ln(\frac{2^m-1}{2^m}) = 0$, and the claim follows. □

By continuing from (71), we get

$$= \left(2^p\right)^{\frac{\sum_{i\in[m]}\left(-i2^{-i-1}+1-2^{-i}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} \prod_{i\in[m]} \left(\frac{2^i-1}{2^i}\right)^{\frac{p\left(1/2-2^{-i-1}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}}$$

$$\geq \lim_{m\to\infty} \left(2^p\right)^{\frac{\sum_{i\in[m]}\left(-i2^{-i-1}+1-2^{-i}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} \prod_{i\in[m]} \left(\frac{2^i-1}{2^i}\right)^{\frac{p\left(1/2-2^{-i-1}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} - \epsilon$$

$$= \lim_{m\to\infty} \left(2^p\right)^{\frac{\sum_{i\in[m]}\left(-i2^{-i-1}+1-2^{-i}\right)}{\sum_{i\in[m]}(1/2-2^{-i-1})}} - \epsilon \tag{72}$$

$$= \lim_{m\to\infty} \left(2^p\right)^{\frac{2^{-m-1}m+m+2^{1-m}-2}{1/2\left(m+2^{-m}-1\right)}} - \epsilon$$

$$= \left(2^p\right)^{\left(\lim_{m\to\infty} \frac{2^{-m-1}m+m+2^{1-m}-2}{1/2\left(m+2^{-m}-1\right)}\right)} - \epsilon$$

$$= \left(2^p\right)^{\left(\lim_{m\to\infty} \frac{m}{1/2(m)}\right)} - \epsilon$$

$$= \left(2^p\right)^2 - \epsilon$$

$$= 4^p - \epsilon,$$

where (72) comes from Fact 3. We conclude that there exists a sufficiently large $m \geq 1$ such that $\mathsf{CR}_{\mathsf{A}}(\mathsf{I}) \geq 4^p - \epsilon$, thus the claim follows.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions.

REFERENCES

[1] H. Ackermann, H. Röglin, and B. Vöcking. 2008. On the impact of combinatorial structure on congestion games. *J. ACM* 55, 6 (2008), 25:1–25:22.

[2] S. Aland, D. Dumrauf, M. Gairing, B. Monien, and F. Schoppmann. 2011. Exact price of anarchy for polynomial congestion games. *SIAM J. Comput.* 40, 5 (2011), 1211–1233.

[3] S. Albers. 1999. Better bounds for online scheduling. *SIAM J. Comput.* 29 (1999), 459–473.

[4] M. Aleksandrov and T. Walsh. 2019. Greedy algorithms for fair division of mixed manna. *CoRR* abs/1911.11005 (2019).

[5] E. Anshelevich, A. Dasgupta, J. M. Kleinberg, É. Tardos, T. Wexler, and T. Roughgarden. 2008. The price of stability for network design with fair cost allocation. *SIAM J. Comput.* 38, 4 (2008), 1602–1623.

[6] J. Aspnes, Y. Azar, A. Fiat, S. A. Plotkin, and O. Waarts. 1997. On-line routing of virtual circuits with applications to load balancing and machine scheduling. *J. ACM* 44, 3 (1997), 486–504.

[7] B. Awerbuch, Y. Azar, and A. Epstein. 2005. The price of routing unsplittable flow. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*. 57–66.

[8] B. Awerbuch, A. Yossi, E. F. Grove, M.-Y. Kao, P. Krishnan, and J. S. Vitter. 1995. Load balancing in the L_p norm. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS)*. 383–391.

[9] Y. Azar, J. Naor, and R. Rom. 1992. The competitiveness of on-line assignments. In *Proceedings of the 3rd Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms (SODA)*. 203–210.

[10] Y. Bartal, A. Fiat, H. Karloff, and R. Vohra. 1995. New algorithms for an ancient scheduling problem. *J. Comput. Syst. Sci.* 51 (1995), 359–366.

[11] M. J. Beckmann, C. B. McGuire, and C. B. Winsten. 1956. *Studies in the Economics of Transportation*. Yale University Press.

[12] X. Bei, J. Garg, M. Hoefer, and K. Mehlhorn. 2019. Earning and utility limits in fisher markets. *ACM Trans. Econ. Computat.* 7, 2 (2019), 10:1–10:35.

[13] F. Benita, V. Bilò, B. Monnot, G. Piliouras, and C. Vinci. 2020. Data-driven models of selfish routing: Why price of anarchy does depend on network topology. In *Proceedings of the 16th International Conference on Internet and Network Economics (WINE)*. 252–265.

[14] K. Bhawalkar, M. Gairing, and T. Roughgarden. 2014. Weighted congestion games: Price of anarchy, universal worst-case examples, and tightness. *ACM Trans. Econ. Computat.* 2, 4 (2014), 1–23.

[15] V. Bilò. 2018. A unifying tool for bounding the quality of non-cooperative solutions in weighted congestion games. *Theor. Comput. Syst.* 62, 5 (2018), 1288–1317.

[16] V. Bilò, I. Caragiannis, A. Fanelli, and G. Monaco. 2013. Improved lower bounds on the price of stability of undirected network design games. *Theor. Comput. Syst.* 52, 4 (2013), 668–686.

[17] V. Bilò, A. Fanelli, M. Flammini, and L. Moscardelli. 2011. Graphical congestion games. *Algorithmica* 61, 2 (2011), 274–297.

[18] V. Bilò, A. Fanelli, M. Flammini, and L. Moscardelli. 2011. Performances of one-round walks in linear congestion games. *Theor. Comput. Syst.* 49, 1 (2011), 24–45.

[19] V. Bilò, M. Flammini, V. Gallotti, and C. Vinci. 2020. On multidimensional congestion games. *Algorithms* 13, 10 (2020), 261.

[20] V. Bilò, M. Flammini, and L. Moscardelli. 2020. The price of stability for undirected broadcast network design with fair cost allocation is constant. *Games Econ. Behav.* 123 (2020), 359–376.

[21] V. Bilò, G. Monaco, L. Moscardelli, and C. Vinci. 2020. Nash social welfare in selfish and online load balancing. In *Proceedings of the 16th International Conference on Internet and Network Economics (WINE)*. 323–337.

[22] V. Bilò, L. Moscardelli, and C. Vinci. 2018. Uniform mixed equilibria in network congestion games with link failures. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*. 146:1–146:14.

[23] V. Bilò and C. Vinci. 2017. On the impact of singleton strategies in congestion games. In *Proceedings of the 25th Annual European Symposium on Algorithms, (ESA)*. 17:1–17:14.

[24] V. Bilò and C. Vinci. 2019. Dynamic taxes for polynomial congestion games. *ACM Trans. Econ. Comput.* 7, 3 (2019), 15:1–15:36.

[25] V. Bilò and C. Vinci. 2019. On Stackelberg strategies in affine congestion games. *Theor. Comput. Syst.* 63, 6 (2019), 1228–1249.

[26] V. Bilò and C. Vinci. 2020. Congestion games with priority-based scheduling. In *Proceedings of the 13th Symposium on Algorithmic Game Theory (SAGT)*. 67–82.

[27] V. Bilò and C. Vinci. 2020. The price of anarchy of affine congestion games with similar strategies. *Theor. Comput. Sci.* 806 (2020), 641–654.

[28] A. Bogomolnaia, H. Moulin, F. Sandomirskiy, and E. Yanovskaia. 2019. Dividing bads under additive utilities. *Soc. Choice Welf.* 52, 3 (2019), 395–417.

[29] A. Borodin and R. El-Yaniv. 1998. *Online Computation and Competitive Analysis*. Cambridge University Press.

[30] S. Brânzei, V. Gkatzelis, and R. Mehta. 2017. Nash social welfare approximation for strategic agents. In *Proceedings of the ACM Conference on Economics and Computation (EC)*. 611–628.

[31] I. Caragiannis. 2008. Better bounds for online load balancing on unrelated machines. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 972–981.

[32] I. Caragiannis. 2013. Efficient coordination mechanisms for unrelated machine scheduling. *Algorithmica* 66, 3 (2013), 512–540.

[33] I. Caragiannis, M. Flammini, C. Kaklamanis, P. Kanellopoulos, and L. Moscardelli. 2011. Tight bounds for selfish and greedy load balancing. *Algorithmica* 61, 3 (2011), 606–637.

[34] I. Caragiannis, V. Gkatzelis, and C. Vinci. 2017. Coordination mechanisms, cost-sharing, and approximation algorithms for scheduling. In *Proceedings of the 13th International Conference on Web and Internet Economics (WINE)*. 74–87.

[35] I. Caragiannis, N. Gravin, and X. Huang. 2019. Envy-freeness up to any item with high Nash welfare: The virtue of donating items. In *Proceedings of the ACM Conference on Economics and Computation (EC)*. 527–545.

[36] I. Caragiannis, C. Kaklamanis, and P. Kanellopoulos. 2010. Taxes for linear atomic congestion games. *ACM Trans. Algor.* 7, 1 (2010).

[37] I. Caragiannis, D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang. 2016. The unreasonable fairness of maximum Nash welfare. In *Proceedings of the ACM Conference on Economics and Computation (EC)*. 305–322.

[38] D. Chakrabarty, A. Mehta, V. Nagarajan, and V. Vazirani. 2005. Fairness and optimality in congestion games. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC)*. 52–57.

[39] C. Chekuri and S. Khanna. 2004. Approximation algorithms for minimizing average weighted completion time. In *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. Chapman & Hall/CRC.

[40] G. Christodoulou and M. Gairing. 2016. Price of stability in polynomial congestion games. *ACM Trans. Econ. Comput.* 4, 2 (2016), 10:1–10:17.

[41] G. Christodoulou, M. Gairing, Y. Giannakopoulos, and P. G. Spirakis. 2019. The price of stability of weighted congestion games. *SIAM J. Comput.* 48, 5 (2019), 1544–1582.

[42] G. Christodoulou and E. Koutsoupias. 2005. The price of anarchy of finite congestion games. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*. 67–73.

[43] G. Christodoulou, K. Mehlhorn, and E. Pyrga. 2011. Improving the price of anarchy for selfish routing via coordination mechanisms. In *Proceedings of the 19th European Conference on Algorithms (ESA)*. 119–130.

[44] G. Christodoulou, V. S. Mirrokni, and A. Sidiropoulos. 2012. Convergence and approximation in potential games. *Theor. Comput. Sci.* 438 (2012), 13–27.

[45] R. Cole, J. R. Correa, V. Gkatzelis, V. Mirrokni, and N. Olver. 2014. Decentralized utilitarian mechanisms for scheduling games. *Games Econ. Behav.* 92 (2014), 306–326.

[46] R. Cole, N. R. Devanur, V. Gkatzelis, K. Jain, T. Mai, V. V. Vazirani, and S. Yazdanbod. 2017. Convex program duality, Fisher markets, and Nash social welfare. In *Proceedings of theACM Conference on Economics and Computation (EC)*. 459–460.

[47] R. Cole, Y. Dodis, and T. Roughgarden. 2006. How much can taxes help selfish routing? *J. Comput. Syst. Sci.* 72, 3 (2006), 444–467.

[48] R. Cole and V. Gkatzelis. 2018. Approximating the Nash social welfare with indivisible items. *SIAM J. Comput.* 47, 3 (2018), 1211–1236.

[49] R. Cominetti, M. Scarsini, M. Schröder, and N. E. Stier Moses. 2019. Price of anarchy in stochastic atomic congestion games with affine costs. In *Proceedings of the ACM Conference on Economics and Computation (EC)*. ACM, 579–580.

[50] A. Czumaj and B. Vöcking. 2007. Tight bounds for worst-case equilibria. *ACM Trans. Algor.* 3, 1 (2007), 4:1–4:17.

[51] J. de Jong, M. Klimm, and M. Uetz. 2016. Efficiency of equilibria in uniform matroid congestion games. In *Proceedings of the 9th International Symposium on Algorithmic Game Theory (SAGT) (LNCS)*, Vol. 9928. 105–116.

[52] E. Even-Dar, A. Kesselman, and Y. Mansour. 2007. Convergence time to Nash equilibrium in load balancing. *ACM Trans. Algor.* 3, 3 (2007), 32.

[53] U. Faigle, W. Kern, and G. Turan. 1989. On the performance of on-line algorithms for particular problems. *Acta Cyber.* 9 (1989), 107–119.

[54] A. Fanelli, M. Flammini, and L. Moscardelli. 2010. Stackelberg strategies for network design games. In *Proceedings of the 6th International Conference on Internet and Network Economics (WINE)*. 222–233.

[55] A. Fanelli, D. Leniowski, G. Monaco, and P. Sankowski. 2015. The ring design game with fair cost allocation. *Theor. Comput. Sci.* 562 (2015), 90–100.

[56] R. Fleischer and M. Wahl. 2000. Online scheduling revisited. *J. Sched.* 3 (2000), 343–353.

[57] P. J. Fleming and J. J. Wallace. 1986. How not to lie with statistics: The correct way to summarize benchmark results. *Commun. ACM* 29, 3 (1986), 218–221.

[58] D. Fotakis. 2010. Stackelberg strategies for atomic congestion games. *Theor. Comput. Sys.* 47, 1 (2010), 218–249.

[59] D. Fotakis, V. Gkatzelis, A. C. Kaporis, and P. G. Spirakis. 2009. The impact of social ignorance on weighted congestion games. In *Proceedings of the 5th International Conference on Internet and Network Economics (WINE)*. 316–327.

[60] D. Fotakis, S. Kontogiannis, and P. Spirakis. 2005. Selfish unsplittable flows. *Theoret. Comput. Sci.* 348 (2005), 226–239.

[61] M. Gairing, T. Lücking, M. Mavronicolas, and M. Monien. 2006. The price of anarchy for restricted parallel links. *Parallel Process. Lett.* 16, 1 (2006), 117–132.

[62] M. Gairing and F. Schoppmann. 2007. Total latency in singleton congestion games. In *Proceedings of the 3rd International Workshop on Internet and Network Economics (WINE) (LNCS)*, Vol. 4858. 381–387.

[63] G. Galambos and G. Woeginger. 1993. An on-line scheduling heuristic with better worst case ratio than Graham's list scheduling. *SIAM J. Comput.* 22 (1993), 349–355.

[64] J. Garg, M. Hoefer, and K. Mehlhorn. 2018. Approximating the Nash social welfare with budget-additive valuations. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2326–2340.

[65] V. Gkatzelis, K. Kollias, and T. Roughgarden. 2016. Optimal cost-sharing in general resource selection games. *Oper. Res.* 64, 6 (2016), 1230–1238.

[66] R. L. Graham. 1966. Bounds for certain multiprocessing anomalies. *Bell Syst. Technic. J.* 45, 9 (1966), 1563–1581.

[67] D. S. Hochbaum and D. B. Shmoys. 1987. Using dual approximation algorithms for scheduling problems: Theoretical and practical results. *J. ACM* 34 (1987), 144–162.

[68] E. Horowitz and S. K. Sahni. 1976. Exact and approximate algorithms for scheduling nonidentical processors. *J. ACM* 23 (1976), 317–327.

[69] S. Ieong, R. McGrew, E. Nudelman, Y. Shoham, and Q. Sun. 2005. Fast and compact: A simple class of congestion games. In *Proceedings of the 20th AAAI Conference on Artificial Intelligence (AAAI)*. 489–494.

[70] N. Immorlica, L. E. Li, V. S. Mirrokni, and A. S. Schulz. 2009. Coordination mechanisms for selfish scheduling. *Theoret. Comput. Sci.* 410, 17 (2009), 1589–1598.

[71] M. Kaneko and K. Nakamura. 1979. The Nash social welfare function. *Econometrica* 47, 2 (1979), 423–435.

[72] D. R. Karger, S. J. Phillips, and E. Torng. 1996. A better algorithm for an ancient scheduling problem. *J. Algor.* 20 (1996), 400–430.

[73] M. Klimm, D. Schmand, and A. Tönnis. 2019. The online best reply algorithm for resource allocation problems. In *Proceedings of the 12th International Symposium on Algorithmic Game Theory (SAGT)*. 200–215.

[74] E. Koutsoupias and C. Papadimitriou. 1999. Worst-case equilibria. In *Proceedings of the 16th Annual Conference on Theoretical Aspects of Computer Science (STACS)*. 404–413.

[75] J. K. Lenstra, D. B. Shmoys, and E. Tardos. 1990. Approximation algorithms for scheduling unrelated parallel machines. *Math. Program.* 46 (1990), 259–271.

[76] T. Lücking, M. Mavronicolas, B. Monien, and M. Rode. 2008. A new model for selfish routing. *Theoret. Comput. Sci.* 406, 3 (2008), 187–2006.

[77] C. A. Meyers and A. S. Schulz. 2012. The complexity of welfare maximization in congestion games. *Networks* 59, 2 (2012), 252–260.

[78] H. Moulin. 1988. *Axioms of Cooperative Decision Making*. Cambridge University Press. DOI:https://doi.org/10.1017/CCOL0521360552

[79] H. Moulin. 2003. *Fair Division and Collective Welfare*. The MIT Press.

[80] J. Nash. 1950. The bargaining problem. *Econometrica* 18, 2 (1950), 155–162.

[81] T. T. Nguyen and J. Rothe. 2014. Minimizing envy and maximizing average Nash social welfare in the allocation of indivisible goods. *Discr. Appl. Math.* 179 (2014), 54–68. DOI:https://doi.org/10.1016/j.dam.2014.09.010

[82] A. C. Pigou. 1938. *The Economics of Welfare*. Macmillan and Co., London.

[83] R. W. Rosenthal. 1973. A class of games possessing pure-strategy Nash equilibria. *Int. J. Game Theor.* 2 (1973), 65–67.

[84] T. Roughgarden. 2003. The price of anarchy is independent of the network topology. *J. Comput. Syst. Sci.* 67, 2 (2003), 341–364.

[85] T. Roughgarden. 2004. The maximum latency of selfish routing. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 980–981.

[86] T. Roughgarden. 2004. Stackelberg scheduling strategies. *SIAM J. Comput.* 33, 2 (2004), 332–350.

[87] T. Roughgarden and É. Tardos. 2002. How bad is selfish routing? *J. ACM* 49, 2 (2002), 236–259.

[88] T. Roughgarden and É. Tardos. 2004. Bounding the inefficiency of equilibria in nonatomic congestion games. *Games Econ. Behav.* 47, 2 (2004), 389–403.
[89] J. F. Rudin III. 2001. *Improved Bounds for the On-line Scheduling Problem.* The University of Texas at Dallas.
[90] S. Suri, C. Tóth, and Y. Zhou. 2007. Selfish load balancing and atomic congestion games. *Algorithmica* 47, 1 (2007), 79–96.
[91] C. Vinci. 2019. Non-atomic one-round walks in congestion games. *Theor. Comput. Sci.* 764 (2019), 61–79.
[92] B. Vöcking. 2007. Selfish load balancing. In *Algorithmic Game Theory*. Cambridge, 517–542.
[93] J. G. Wardrop. 1952. Some theoretical aspects of road traffic research. *Proc. Instit. Civil Eng. Part II* 1, 36 (1952), 352–362.

Received January 2021; revised March 2022; accepted June 2022