Dynamic Games and Applications
https://doi.org/10.1007/s13235-022-00470-y

Robust Utility Design in Distributed Resource Allocation Problems with Defective Agents

Bryce L. Ferguson[1] · **Jason R. Marden**[1]

Accepted: 4 August 2022

Abstract
The use of multi-agent systems to solve large-scale problems can be an effective method to reduce physical and computational burdens; however, these systems should be robust to sub-system failures. In this work, we consider the problem of designing utility functions, which agents seek to maximize, as a method of distributed optimization in resource allocation problems. Though recent work has shown that optimal utility design can bring system operation into a reasonable approximation of optimal, our results extend the existing literature by investigating how robust the system's operation is to defective agents and by quantifying the achievable performance guarantees in this setting. Interestingly, we find that there is a trade-off between improving the robustness of the utility design and offering good nominal performance. We characterize this trade-off in the set of resource covering problems and find that there are considerable gains in robustness that can be made by sacrificing some nominal performance.

Keywords Price of anarchy · Utility design · Potential games · Multi-agent systems · Robust learning

1 Introduction

Multi-agents systems have emerged as a viable method of implementing distributed system operation. In teams of robots [11, 15], resource allocation problems [3, 13], autonomous mobility and delivery [1, 4], and many other large-scale systems, distributing certain decision

This work is supported in part by ONR Grant #N00014-20-1-2359, AFOSR Grants #FA9550-20-1-0054 and #FA9550-21-1-0203, the Army Research Lab through ARL DCIST CRA W911NF-17-2-0181.

This article is part of the topical collection "Multi-agent Dynamic Decision Making and Learning" edited by Konstantin Avrachenkov, Vivek S. Borkar and U. Jayakrishnan Nair.

✉ Bryce L. Ferguson
blferguson@ece.ucsb.edu

Jason R. Marden
jrmarden@ece.ucsb.edu

1 Electrical & Computer Engineering, University of California, Santa Barbara, Santa Barbara, CA, USA

making processes to individual agents can help reduce computational and communication burdens. Designing local control laws for agents that guarantee good system performance overall can be difficult; a promising method of solving this problem is by using tools from game theory to describe the system operation [21]. A system designer can assign each agent a local objective or utility function which they seek to maximize. By carefully designing these utility functions, the system designer can guarantee that the emergent system behavior is within a good approximation of the system optimal [5, 12]. Though the use of game theoretic techniques in distributed control is encouraging, the robustness of these utility designs is not well understood, and certain sub-system failures may cause significant degradation in the system performance.

In this work, we consider the problem of utility design in resource allocation problems in which each agent selects a set of resources (or tasks) with the objective of maximizing the system welfare through the agents' collective actions. To determine a preference over actions, each agent is assigned a local objective (or utility) function which they seek to maximize by repeatedly updating their action; the equilibria of this process are the Nash equilibria of the game formed between the agents with their payoffs described by their local objectives. By designing the utility rules of the agents, a system operator can alter the Nash equilibria and improve the equilibrium performance guarantees. The problem of utility design has been studied in many settings of resource allocation problem [6, 17, 22] and has been proven to be effective at offering reasonable approximations of the optimal system welfare. In this work, we seek to understand the robustness of utility rules to sub-system failures by investigating the impact of defective agents on performance guarantees.

Previous work has considered how sub-system failures can affect the operation of a distributed control system. In [8], the authors look at how performance of greedy submodular-maximization degrades as communication channels between agents are removed and find that certain design modifications can be beneficial. The authors of [10] introduce a framework to discuss the robustness properties of log-linear learning: a process in which users' action updates are noisy best responses. They find that the presence of a single heterogeneous agent can alter the long-term group behavior but do not discuss the impact this agent has on system welfare. In [23], the authors consider the effect of heterogeneous agents on opinion dynamics in networked systems and their impact on group consensus. In this work, we not only consider how these types of agents affect the system operation and equilibria but also how they impact the system welfare and achievable performance guarantees of a utility design. We further consider how existing design techniques can be modified to offer improved robustness to defective agents.

The results of this work give insight on the robustness of game theoretic techniques of distributed control in resource allocation problems with defective agents. We consider two types of defective agents: (1) stubborn agents which do not update their actions and do not contribute to the system welfare but do alter the agents' perceived utilities, and (2) failure-prone agents that have a probability of failing to contribute to the system objective. In either setting, the presence of these defective agents will alter the intended system operation and worsen system performance guarantees. In Sect. 3, we first leverage existing results on utility design for resource allocation to provide a tractable linear program which can be used to compute the optimal, robust utility design in the presence of either type of defective agent. This program not only tells us the structure of the optimal utility design, but also the associated performance guarantees. In Sect. 4, to better understand the affect these defective agents have on system performance, we focus on the class of covering problems. We find that significant performance improvements are attainable by designing utility rules more robustly; however, these design modifications necessarily reduce the system performance in the nominal setting. In Theorem 2, we characterize the trade-off frontier between offering good nominal and robust performance in the presence of a finite number of stubborn agents. Similarly, in Sect. 4.2 we offer a numerical analysis to highlight a similar trade-off in the setting of failure-prone agents.

2 Preliminaries

Consider a multi-agent system comprised of a finite set of agents $N = \{1, \ldots, n\}$ whose interactions comprise the operation of the system. Each agent $i \in N$ has a set of allowable actions $\mathcal{A}_i$ that they can employ. When each agent $i \in N$ has selected an action $a_i \in \mathcal{A}_i$, we denote the joint action or allocation of users decisions by the tuple $a = (a_1, \ldots, a_n) \in \mathcal{A} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$. To quantify the efficacy of system operation, each joint action is mapped to a global objective value via a system welfare function $W : \mathcal{A} \to \mathbb{R}$.

The objective of a system operator is to design a distributed decision-making process that enables system operation at a joint action $a \in \mathcal{A}$ that maximizes the system welfare $a^{\text{opt}} \in \arg\max_{a \in \mathcal{A}} W(a)$. One popular approach to the design of such processes is to design a utility function $U_i : \mathcal{A} \to \mathbb{R}$ for each agent $i \in N$ in the system that will influence and guide the agent's local decision making [21]. By employing suitable distributed learning algorithms, e.g., fictitious play [2] or regret matching [9], a system operator can ensure that the resulting distributed process converges to an equilibrium of the designed games. This work will primarily focus on the notion of pure Nash equilibrium, which is characterized by a joint action profile $a^{\text{Ne}} \in \mathcal{A}$ that satisfies

$$U_i(a_i^{\text{Ne}}, a_{-i}^{\text{Ne}}) \geq U_i(a_i', a_{-i}^{\text{Ne}}), \quad \forall a_i' \in \mathcal{A}_i,\ i \in N, \tag{1}$$

where a_{-i} denotes the action of all players excluding player i. Here, it is important to highlight that the design of utility function shapes the underlying equilibria of the systems which in turn influences the performance of the underlying system at such equilibria.

One of the fundamental goals of a system operator is to design utility function that lead to highly efficient equilibria as measured with regard to the system welfare. We measure the efficiency of a Nash equilibrium by the ratio between the equilibrium system welfare and the optimal system welfare. This performance guarantee is termed the *price of anarchy* and can formally be defined as

$$\text{PoA}(G) = \frac{\min_{a \in \text{NE}(G)} W(a)}{\max_{a \in \mathcal{A}} W(a)}, \tag{2}$$

where $G = (N, \mathcal{A}, \{U_i\}_{i \in N}, W)$ is a given game equipped with a performance metric and $\text{NE}(G)$ is the set of Nash equilibria in the game. The effectiveness of utility design has been studied in many settings and has been shown to be effective at providing a reasonable approximation for the system optimal [6, 13]. In fact, in many cases such an approach provides the same approximation ratio as the best centralized algorithms [16].

The focus of this paper is on class of multi-agent systems pertaining to resource allocation problems. Here, there is ground set of resources (or tasks) $\mathcal{R}$ and each agent $i \in N$ is associated with an admissible choice set $\mathcal{A}_i \subset 2^{\mathcal{R}}$. Further, each resource r is associated with a non-decreasing welfare function $w_r : \{0, 1, \ldots, n\} \to \mathbb{R}_{\geq}$, where $w_r(k)$ encodes the welfare accrued at resource r when $k \geq 0$ agents are at resource r. The system welfare can thus be written as

$$W(a) = \sum_{r \in \mathcal{R}} w_r(|a|_r), \tag{3}$$

where $|a|_r$ denotes the number of agents utilizing resource r in allocation a. As above, the goal of a system operator is to design agent utility functions to optimize the efficiency of the resulting equilibria. To that end, here we focus on the design of agent utility functions where each resource r is associated with a *local utility rule* $f_r : \{0, \ldots, n\} \to \mathbb{R}_{\geq 0}$ that describes how agents should assess the benefit of selecting resource r given the behavior of the other agents in the system. For these utility designs to be robust to different problem instances, we assume that the local utility function is the same for each agent.[1] When a system designer chooses a set of local utility rules $f = \{f_r\}_{r \in \mathcal{R}}$, the utility of agent i given an action profile $a \in \mathcal{A}$ is of the form

$$U_i(a) = \sum_{r \in a_i} f_r(|a|_r). \tag{4}$$

We will express a resource allocation game by the tuple $G = (N, \mathcal{A}, \mathcal{R}, \{w_r, f_r\}_{r \in \mathcal{R}})$, as the utility functions and welfare functions are derived from the pairs $\{w_r, f_r\}_{r \in \mathcal{R}}$. A resource allocation game, with non-decreasing local utility functions, permits a potential function and thus at least one Nash equilibrium exists [17].

When designing the local utility rules f, a system operator may have minimal information about the specific game instance G, such as uncertainty about the agents available actions or the full set of resources. For example, in the context of content distribution networks and data caching, the system operator has little knowledge of the available servers and the paths over the internet that connect them [7], in ride-sharing, the requests of passengers are not known until they are placed [20], or in team formations, the tasks may be changing over time [14]. A common assumption in the literature is that the system operator has knowledge of the type of different resource welfare functions, which we express by $\mathcal{W}$, but is unsure of the specific game instance or the specific welfare functions employed. The goal of a system operator is to design a local utility rule f_r for each welfare function $w_r \in \mathcal{W}$. We define this association by the map $\mathcal{F} : \mathcal{W} \to \mathbb{R}^n_{\geq 0}$, where $f_r = \mathcal{F}(w_r)$. Accordingly, given a set of potential welfare functions $\mathcal{W}$ and a local utility rule $\mathcal{F}$, we denote the set of possible games as

$$\mathcal{G}_{\mathcal{W},\mathcal{F}} = \{(N, \mathcal{A}, \mathcal{R}, \{w_r, \mathcal{F}(w_r)\}_{r \in \mathcal{R}}) : w_r \in \mathcal{W}, \mathcal{A}_1, \ldots, \mathcal{A}_n \subseteq 2^{\mathcal{R}}\}.$$

Note that $\mathcal{G}_{\mathcal{W},\mathcal{F}}$ consists of any resource allocation game with n agents where each resource $r \in \mathcal{R}$ employs the pair $\{w_r, f_r = \mathcal{F}(w_r)\}$ where $w_r \in \mathcal{W}$. Given the set of possible welfare functions $\mathcal{W}$, the goal of a system operator is to design the local utility rules $\mathcal{F}$ to optimize the price of anarchy over the family of games $\mathcal{G}_{\mathcal{W},\mathcal{F}}$, i.e.,

$$\mathrm{PoA}(\mathcal{G}_{\mathcal{W},\mathcal{F}}) = \inf_{G \in \mathcal{G}_{\mathcal{W},\mathcal{F}}} \mathrm{PoA}(G). \tag{5}$$

There are several recent works that highlight how to compute the local utility rule $\mathcal{F}$ that optimizes the price of anarchy given in (5) [5, 17].

[1] Because we consider a worst-case analysis, we cannot differentiate between the role of two agents, as if we designed the utility rule of agent i and j differently, a problem may be realized where their roles are reversed. If the system designer had full knowledge of the problem structure, designing agents' utilities heterogeneously can certainly help; however, it is currently unknown as to whether player specific utility functions can help in improving worst-case performance guarantees across a class of problem instances. Additionally, adopting a local utility rule that is consistent for each player lets us maintain the potential game structure.

3 Robust Utility Design

This work focuses on understanding the robustness of these game theoretic methods of distributed control to sub-system failures in the form of defective agents. As such, we seek to understand the impact of defective agents on the price of anarchy and what improvements are possible when agents' utilities are designed with the knowledge that these hazards may exist. Specifically, we consider two different modes of agent failures, stubborn agents and failure prone agents, and illustrate how one can design robust utility function to optimize performance guarantees in these non-ideal settings.

3.1 Stubborn Agents

We consider a setting where there exist a finite number of stubborn agents in the system. While these stubborn agents do not contribute to the system welfare, they do impact the behavior of the agents in the systems and can potentially diminish the quality of the resulting equilibria.

To that end, we consider a scenario where there are at most m defective agents that can each occupy a subset of the resources in $\mathcal{R}$. Because each stubborn agent can occupy any subset of the resources, we instead represent the allocation of stubborn agents by the number of stubborn agents on each resource. Accordingly, we will express the behavior of the stubborn agents by the tuple $d = \{d_r\}_{r\in\mathcal{R}}$, where $d_r \in \{0, 1, \dots, m\}$ for each resource $r \in \mathcal{R}$. Further, we will express the collective behavior in our system by the tuple $(a; d)$ to reflect the behavior of both the nominal and stubborn agents. Here, it is important to highlight that the stubborn agents do not impact the system welfare, which stays of the form

$$W(a; d) = \sum_{r\in\mathcal{R}} w_r(|a|_r).$$

However, the presence of stubborn agents does affect the utility functions of the nominal agents who are unable to differentiate between the behavior of stubborn agents and other nominal agents. Here, the utility function for any agent $i \in N$, action profile $a \in \mathcal{A}$, and stubborn agent profile d is of the form

$$U_i(a; d) = \sum_{r\in a_i} f_r(|a|_r + d_r), \tag{6}$$

where $f_r \in \mathbb{R}^{n+m}_{\geq 0}$. We will now measure the price of anarchy associated with a given form of stubborn behavior $d \in \{0, \dots, m\}^{|\mathcal{R}|}$ as

$$\mathrm{PoA}(G, d) = \frac{\min_{a\in\mathrm{NE}(G,d)} W(a)}{\max_{a^{\mathrm{opt}}\in\mathcal{A}} W(a^{\mathrm{opt}})}, \tag{7}$$

where $\mathrm{NE}(G, d)$ defines the equilibria associated with the game defined by d given in (6). As stated before, the goal of a system operator is to design a local utility rule $\mathcal{F}$ that optimizes the worst-case performance measure over both game instances $G \in \mathcal{G}_{\mathcal{W},\mathcal{F}}$ and stubborn-agent behavior $d \in \{0, \dots, m\}^{|\mathcal{R}|}$, i.e.,

$$\mathrm{PoA}(\mathcal{G}^m_{\mathcal{W},\mathcal{F}}) = \inf_{G\in\mathcal{G}_{\mathcal{W},\mathcal{F}}} \min_{d\in\{0,\dots,m\}^{|\mathcal{R}|}} \mathrm{PoA}(G, d). \tag{8}$$

The following proposition demonstrates that one can find the local utility rule $\mathcal{F}$ that optimizes the price of anarchy via the solution of a tractable linear program.

Proposition 1 *Let* $\mathcal{W} = \{\sum_{b=1}^{B} \alpha_b w_b | \alpha_b \geq 0 \, \forall b \in [B]\}$ *be a set of resource value functions where* $w_b \in \mathbb{R}^n_{\geq 0}$ *is a basis welfare function. For each* $b \in [B]$*, let* $(f_b^\star, \mu_b^\star)$ *be the solution of the following linear program*

$$\begin{aligned}(f_b^\star, \mu_b^\star) \in \underset{f\in\mathbb{R}^{n+m},\ \mu\in\mathbb{R}}{\arg\max} \quad & \mu \\ \text{s.t. } & w_b(z+y) - \mu w_b(x+y) \\ & \quad + xf(x+y+d) - zf(x+y+d+1) \leq 0 \\ & \forall (x,y,z) \in \mathcal{I}_n,\ d \in \{0,\dots,m\}, \end{aligned} \tag{9}$$

where $\mathcal{I}_n = \{(x,y,z) \in \mathbb{N}^3_{\geq 0} \mid 1 \leq x+y+z \leq n\}$*. The following statements hold true for the family of resource allocation games with at most n agents and m stubborn agents:*

(i) *There exists a linear local utility rule* $\mathcal{F}^*$ *that optimizes the price of anarchy given in* (8)*. Furthermore, if* $w = \sum_{b=1}^{B} \alpha_b w_b$*, then* $\mathcal{F}^*(w) = \sum_{b=1}^{B} \alpha_b f_b^*$.
(ii) *The optimal price of anarchy satisfies* $\mathrm{PoA}(\mathcal{G}^m_{\mathcal{W},\mathcal{F}^*}) = \max_{b\in[B]} 1/\mu_b^\star$.

The linear program (9) is a generalization of [5, Theorem 3.7] which solves the local utility design problem in the nominal setting and is recovered when $m = 0$. The proof appears in Appendix A. Here, the local utility function f is treated as a vector in $\mathbb{R}^{n+m}$ where $f(i)$ denotes what the utility rule evaluates to when i agents utilize a resource; the linear program thus has $n + m + 1$ decision variables and $|\mathcal{I}_n|$ constraints. It is interesting to note that the optimal local utility design in a class of games of the described form decomposes into finding an optimal basis local utility function for each basis value function (a phenomenon first observed in [18]), making the computation of these utility rules more efficient. The parameterization using tuples of the form (x, y, z, d) is described in Appendix along with the proof. This approach not only solves for the optimal local utility rule, but also gives the associated price of anarchy guarantee. By comparing the solution of (9) when $m = 0$ and when $m > 0$, we can observe the impact stubborn agents have on the capabilities of a system designer. The magnitude of this performance degradation is discussed in Sect. 4.1, along with an investigation of the impact the design modifications that promote robustness have on the nominal performance of the system.

3.2 Failure Prone Agents

The second type of defective agent we consider are failure prone agents, where each agent operates normally but has a probability of failing and not contributing to the global objective. Every agent will follow the designed utility rule but has a chance of failing and no longer contributing to the system welfare. In contrast to the defective agents in the previous section which were ineffective and stubborn in their action selection, here an agent that fails can be thought of as ineffective but still updating their actions as a normal agent would. Additionally, each agent independently fails to contribute to the welfare with probability p. In a resource allocation problem G, each agent (failed or not) will follow their best response dynamic until the system reaches a Nash equilibrium $a^{\mathrm{Ne}} \in \mathrm{NE}(G)$. In an allocation a, a resource $r \in \mathcal{R}$, utilized by $|a|_r$ agents, has $X_r \leq |a|_r$ non-failed agents remaining with probability

$$\mathbb{P}[X_r = x] = \binom{|a|_r}{x}(1-p)^x \, p^{|a|_r - x}.$$

In a game G, the expected system welfare in an allocation a is

$$W(a) = \mathbb{E}\left[\sum_{\{r \in \mathcal{R}: X_r > 0\}} w_r(X_r)\right]. \tag{10}$$

The price of anarchy when agents are failure-prone $\mathrm{PoA}(G; p)$ is thus the worst-case ratio between the expected system welfare in a Nash equilibrium and the optimal expected system welfare. The worst-case performance guarantee is a lower bound on the price of anarchy over resource covering games with a probability of failure p, i.e., $\mathrm{PoA}(\mathcal{G}_{\mathcal{W},\mathcal{F}}; p)$. As described in Sect. 3.1, the optimal local utility design problem can be described as finding a mapping $\mathcal{F}$ that maximizes $\mathrm{PoA}(\mathcal{G}_{\mathcal{W},\mathcal{F}}; p)$.

Proposition 1 provides a tool for computing local utility rules that are robust to stubborn agents for general local welfare functions. Additionally, we can amend the local welfare function and use the same linear program to compute the optimal local utility rule in the face of failure prone agents.

Corollary 1 *Let $\mathcal{W} = \{\sum_{b=1}^{B} \alpha_b w_b | \alpha_b \geq 0\ \forall b \in [B]\}$ be a set of resource value functions where $w_b \in \mathbb{R}^n_{\geq 0}$ is a basis welfare function. If agents fail with probability p, then for each $b \in [B]$, let $(f_b^\star, \mu_b^\star)$ be the solution of the (9) with $m = 0$ and the amended value functions*

$$\overline{w}_b(x) = \sum_{k=0}^{x} w_b(k)\binom{x}{k}(1-p)^k\ p^{x-k} \quad \forall\, b \in [B]. \tag{11}$$

The following statements hold true for the family of resource allocation games with at most n agents and probability of failure p:

(i) *There exists a linear local utility rule $\mathcal{F}^*$ that optimizes the price of anarchy given in (8). Furthermore, if $w = \sum_{b=1}^{B} \alpha_b w_b$, then $\mathcal{F}^*(w) = \sum_{b=1}^{B} \alpha_b f_b^*$.*
(ii) *The optimal price of anarchy satisfies* $\mathrm{PoA}(\mathcal{G}_{\mathcal{W},\mathcal{F}^*}; p) = \max_{b \in [B]} 1/\mu_b^\star$.

4 Efficacy of Robust Design

In Sect. 3, a linear program was introduced that can be used to solve for the optimal local utility design and associated performance guarantee. In this section, we seek to better understand the impact of these defective agents and what opportunities are available by utilizing a more robust design. To answer this, we look at a specific setting of resource allocation problems called *covering problems*. Here, each resource r contributes a fixed value to the system if it is utilized (or covered) by at least one agent. The welfare function for a covering problem can thus be written as

$$W(a) = \sum_{r \in \mathcal{R}} v_r \cdot w(|a|_r),$$

where $w(x)$ is an indicator function that takes value 1 if $x > 0$ and 0 otherwise. These problems are a specification of the more general problem description in Sect. 3 where there is now only a single basis function and $\mathcal{W} = \{\alpha \cdot w \mid \alpha > 0\}$. The focus to this setting allows for a more detailed analysis of robust performance as closed form solutions to general resource allocation problems with defective agents is still an open problem.

As discussed in Proposition 1, the optimal local utility rules satisfy a linearity property, therefore in the set of covering problems, we need only find a single local utility function

$f : [n] \to \mathbb{R}_{\geq 0}$ that is scaled by the resources value v_r to determine the local utility, i.e.,

$$U_i(a) = \sum_{r \in a_i} v_r \cdot f(|a|_r).$$

As the welfare functions are described by a single basis, and we need only consider a single vector f to define a utility rule, the set of covering games can be denoted $\mathcal{G}_{w,f}$. For brevity, and because the welfare basis function will not change, we will simply denote this set of games $\mathcal{G}_f$ and note the covering welfare function is implied. Altering f will constitute changes to the local utility design. Finding the form of the optimal local utility rule f in closed form is not trivial. However, in [6] the authors find that the local utility rule

$$f^0(j) := (j-1)! \frac{\frac{1}{(n-1)(n-1)!} + \sum_{i=j}^{n-1} \frac{1}{i!}}{\frac{1}{(n-1)(n-1)!} + \sum_{i=1}^{n-1} \frac{1}{i!}}, \tag{12}$$

is optimal in the nominal covering setting and provides a price of anarchy guarantee of $1 - \frac{1}{e}$. In this work, we investigate how performance guarantees like this are impacted by the presence of defective agents. Additionally, we ask if appropriate design modifications to the local utility rule can make the system more robust to defective agents and what impact these design modifications have on the systems nominal performance.

4.1 Stubborn Agents in Covering

In Proposition 1, a linear program was introduced whose solution gives the optimal local utility rule for a class of games with stubborn agents. In this section, we seek to further understand the impact of stubborn agents on performance guarantees by finding the optimal, robust local utility rule and associated performance guarantee in covering problems. Interestingly, we find that though robust design modifications can improve robust performance, these changes necessarily reduce the utility rules performance in the nominal setting, thus highlighting a trade-off between guaranteeing good robust and nominal performance. We let $\mathcal{G}_f^m$ denote the set of covering games with local utility rule f and at most m stubborn agents. By comparing the price of anarchy guarantees in the nominal setting $\mathcal{G}_f^0$ and in the presence of stubborn agents $\mathcal{G}_f^m$, we can discuss not only the newly found robustness of a utility rule f, but also what performance guarantees it maintains in the nominal setting.

Theorem 2 quantifies the trade-off between utility rules that are robust to stubborn agents and utility rules with good nominal performance in terms of price of anarchy guarantees.

Theorem 2 *Let $t \in [0, 1]$ be a chosen tuning parameter. In the class of covering games, if a local utility rule f achieves a robust price of anarchy guarantee of*

$$\mathrm{PoA}(\mathcal{G}_f^m) \geq \frac{\Gamma_m + \frac{e}{e-1}}{1 + t\Gamma_m}, \tag{13}$$

where $\Gamma_m = m! \frac{e - \sum_{i=0}^{m-1} \frac{1}{i!}}{e-1} - 1$, then its nominal price of anarchy guarantee will be no better than

$$\mathrm{PoA}(\mathcal{G}_f^0) \leq \frac{(e-1)(1+t\Gamma_m)}{1 + (e-1)(1+t\Gamma_m)}. \tag{14}$$

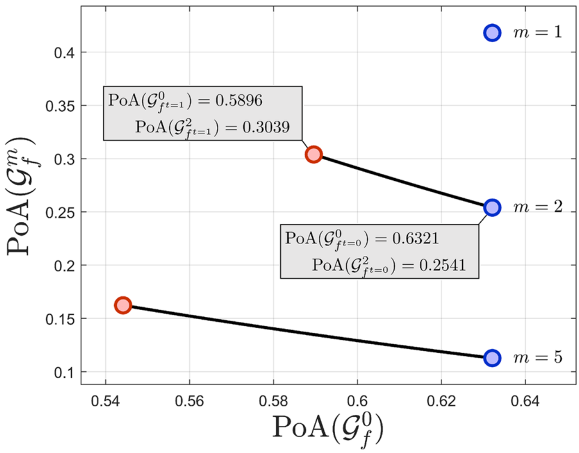


Fig. 1 Achievable price of anarchy guarantees in the nominal setting $\mathcal{G}^0_f$ and in the presence of m stubborn agents $\mathcal{G}^m_f$. Each line represents a Pareto-optimal frontier for the achievable performance guarantee in each setting for a specific m. The left (red) endpoints represent the price of anarchy guarantees of the optimal, robust utility rule $f^{t=1}$, and the right (blue) endpoints represent the price of anarchy guarantees of the optimal, nominal utility rule $f^{t=0}$. A system designer is only capable of offering joint performance guarantees that are on the line connecting the endpoints or lower (Color figure online)

Further, the price of anarchy guarantees in (13) *and* (14) *can be jointly realized by a local utility rule*

$$f^t(j) = f^0(j) - \max\left\{t\left(f^0(j) - \frac{m}{j}f^0(m)\right), 0\right\}. \tag{15}$$

The proof of Theorem 2 appears in "Appendix B."

The trade-off described in (13) and (14) is depicted in Fig. 1 for several values of m. The horizontal axis measures the nominal price of anarchy and the vertical axis measures the price of anarchy when there are at most m stubborn agents. By choosing $t = 0$ the local utility rule optimizes the nominal price of anarchy guarantee and choosing $t = 1$ optimizes the robust price of anarchy guarantee. The line drawn by varying the parameter $t \in [0, 1]$ constitutes a Pareto-optimal frontier on the multi-objective problem of maximizing the nominal and robust performance guarantees.

By letting $t = 0$, we can evaluate the performance of the optimal, nominal utility rule $f^{t=0} = f^0$ defined in (12). Clearly, the performance degrades as more stubborn agents are introduced into the problem: the presence of two stubborn agents reduces the performance of the nominal utility rule $f^{t=0}$ by almost 60% down to $\mathrm{PoA}(\mathcal{G}^{m=2}_{f^{t=0}}) = 0.2541$. By designing the utility rule more robustly, the price of anarchy guarantee in $\mathcal{G}^{m=2}_f$ can be improved by almost 20% by using $f^{t=1}$; however, this increase in robustness comes at the cost of nominal performance, as the local utility rule $f^{t=1}$ is approximately 7% less efficient than the optimal in the nominal setting. A system designer who would like to optimize both performance metrics can provide guarantees only up to the Pareto-optimal frontier described by (13) and (14) and shown in Fig. 1; these Pareto-optimal performance guarantees can be achieved by using f^t for $t \in [0, 1]$.

4.2 Failure-Prone in Covering

4.2.1 Optimal Utility Rules

Utilizing our results from Sect. 3.2, we perform a numerical analysis with (9) to understand the necessary design modifications and attainable performance guarantees with failure-prone

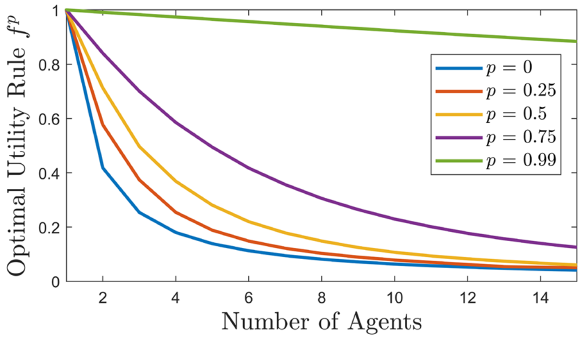


Fig. 2 Optimal local utility rules that are robust to failure-prone agents for several probabilities of failure p. As the probability of agent failure increases, the optimal utility rule is larger for values of $k > 1$, thus incentivizing more overlap in agents resource usage

agents in the setting of resource covering, i.e., w is the indicator function. From Corollary 1, we can compute optimal utility rules and price of anarchy guarantees using the augmented value function

$$\overline{w}(x) = \sum_{k=0}^{x} w(k) \binom{x}{k} (1-p)^k \, p^{x-k} = 1 - p^x,$$

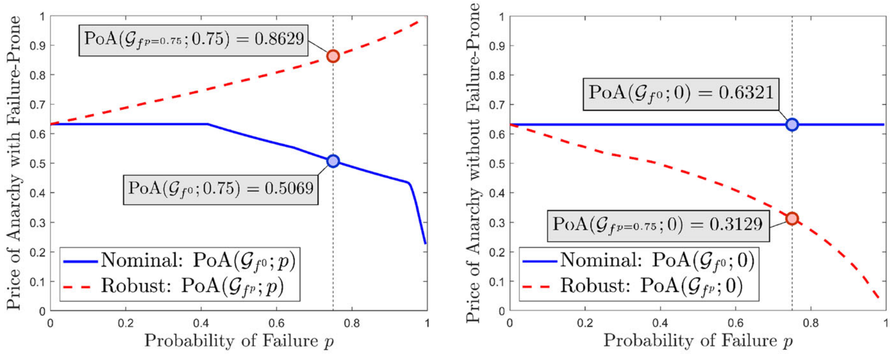
and the linear program provided in (9) with $m = 0$. Via the solution to the linear program, we not only investigate how the optimal utility rule and price of anarchy change with the probability of failure, we further investigate how the design modifications affect the nominal performance, had the agents not been failure prone. Additionally, we consider these performance metrics when the agents are not only failure-prone, but when there exist stubborn agents as well.

First, we utilize (9) to compute the optimal utility rule in the presence of failure-prone agents for several values of p, the probability failure. In Fig. 2, we see that the optimal local utility rule $f^p(k)$ increases with p for values of $k > 1$. Intuitively, this implies that it is optimal to design agents utilities as to promote more overlap in the resources they utilize. The larger the number of agents utilizing a resource will lead to a better chance that the resource will be covered by at least one non-failed agent. As p approaches 1, it is optimal for all agents to greedily, and without consideration of one another, to choose the most valuable set of resources.

Though these design modifications may make the system more robust to failure-prone agents, they may not be effective in the nominal setting.

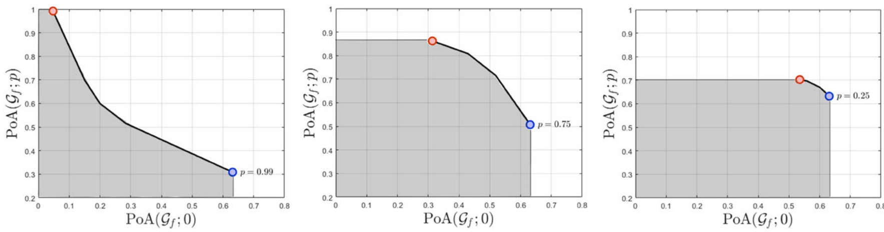
4.2.2 Nominal and Robust Performance

In Fig. 3, we analyze the performance trade-off between designing utility rules for the nominal and failure-prone settings. For $p \in [0, 1]$, we compute the optimal utility rule f^p that is robust to agent failure with probability p using (9) with $m = 0$. We then compare the performance of this new, robust utility rule with the performance of the nominal utility rule f^0, defined in (12), when agents are failure-prone and when agents are not failure-prone. Figure 3 shows the price of anarchy guarantees of the nominal and robust utility rules in presence of failure-prone agents. When the probability of failure is large, the robust utility rule offers large improvements to the expected system welfare; when $p = 0.75$, the robust utility rule offers a price of anarchy guarantee of $\text{PoA}(\mathcal{G}_{f^{p=0.75}}; p = 0.75) = 0.8629$ which is a 63% increase from the performance of the nominal utility rule f^0 in this setting. However, as seen in Fig. 3b,



(a) Failure-prone (b) Not Failure-Prone

Fig. 3 **a** Price of anarchy in covering problems with failure prone agents under nominal utility rule f^0 and robust utility rule f^p. The robust utility rule offers significant performance improvements in the presence of failure prone agents. **b** Price of anarchy in covering problems under nominal utility rule f^0 and robust utility rule f^p designed for agents failing with probability p. The robust utility rule sacrifices notable performance in the original setting



(a) $p = 0.99$ (b) $p = 0.75$ (c) $p = 0.25$

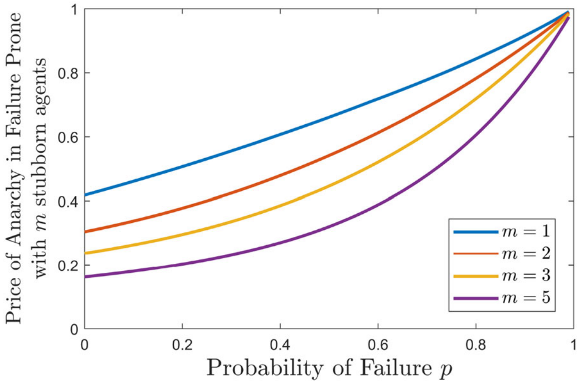
Fig. 4 Trade-off between nominal and robust performance guarantees for agents that fail with probability $p = 0.99$ (left), $p = 0.75$ (middle), and $p = 0.25$ (right). Each line represents an empirical Pareto-optimal frontier for the achievable joint price of anarchy bound in each setting. The left (red) endpoints represent the price of anarchy guarantees of the optimal, robust utility rule f^p, and the right (blue) endpoints represent the price of anarchy guarantees of the optimal, nominal utility rule f^0. By simulation, utility rules were generated that populated the grey region of each plot and up to the trade-off frontier (Color figure online)

if the system designer is incorrect and the agents are not failure-prone, the use of the robust utility rule causes a loss in performance; when $p = 0.75$ the price of anarchy of the robust utility rule without failure prone agents is $\text{PoA}(\mathcal{G}_{f^{p=0.75}}; p = 0) = 0.3129$, which is a 49% decrease from if the nominal had been used. This difference in price of anarchy guarantees again highlights a trade-off between the achievable nominal and robust performance. As the probability of failure increases, the optimal local utility rules in the robust setting struggle to offer good, nominal performance guarantees.

4.2.3 Performance Trade-off

The previous numerical result highlights a trade-off between optimizing nominal performance and robust performance in the setting of failure-prone. However, Fig. 3 only captures the fact that optimizing both objectives becomes more difficult as the probability of failure increases. Similar to Sect. 3.1, we would also like to understand the trade-off between the objectives

Fig. 5 Price of anarchy guarantee in the setting of failure-prone agents where m stubborn agents are present. The presence of stubborn agents unsurprisingly lowers the performance guarantees, but the amount by which this happens appears to be relatively independent of p, the probability of failure



when good (but not optimal) performance is desired in both settings. In Fig. 4, we offer a bound on this trade-off by means of Monte Carlo simulation. By randomly generating many possible local utility rules f, and using (9) to compute the performance guarantees in the nominal and robust setting, we are able to generate a lower bound on the Pareto-optimal frontier. Figure 4 demonstrates similar results to Fig. 1 in that the trade-off becomes more significant as the magnitude of the possible sub-system failures (here failure-prone agents) increases. Additionally, it is clear that reasonable concessions can be made to either the nominal or robust objective to guarantee better performance in both.

4.2.4 Failure-Prone and Stubborn Agents

Finally, we consider the case where agents are not only failure-prone (are ineffective but not stubborn), but that there also exists some defective, stubborn agents as introduced in Sect. 3.1. Here, we again utilize (9) with welfare function defined accordingly by Corollary 1 to represent the failure prone agent setting, but now with $m > 0$ stubborn agents. In Fig. 5, we see the price of anarchy guarantees unsurprisingly degrade as more stubborn agents are introduced into the system, however it is interesting to note that the relative amount of this degradation does not vary much with p. This essentially shows that the two forms of failures do not cause cascading issues but do degrade system performance.

5 Conclusion

This work studies the robustness of local utility rules to sub-system failures in the form of stubborn and failure-prone agents. We provide linear programs that compute and evaluate the optimal local utility rules in the face of these defective agents. Our results show that there is a trade-off in designing utility rules that are robust and that give good nominal performance, which is characterized for the setting of covering problems.

Data Availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Appendix A : Proof of Proposition 1

We note that this proof follows similarly to that of [5, 17] but now with the presence of stubborn agents. Here we go through the construction of the linear program and important steps of the proof, but direct the reader to [5, 17] for a more detailed explanation. We begin by identifying the problem of characterizing the price of anarchy in a class of games $\mathcal{G}^m_{\mathcal{W},\mathcal{F}}$ for a class of games with a single basis value function w (i.e., $\mathcal{W} = \{\alpha w \mid \alpha > 0\}$) while using a local utility rule $f = \mathcal{F}(w)$. Each resource welfare function can therefore be written as $w_r(x) = v_r w(x)$, where v_r is the 'value' of that specific resource. We will discuss at the end how the solution to a single basis function can extend to the original statement. In looking for price of anarchy bounds we note that a class of resource covering problems $\mathcal{G}$ with utility rule f has the same price of anarchy as the class of problems $\mathcal{G}^*$ where each agent has exactly two actions $\mathcal{A}_i = \{a_i^{\mathrm{Ne}}, a_i^{\mathrm{opt}}\}$,[2] thus we will search for price of anarchy bounds in these two-action games and note they hold more generally. The price of anarchy over $\mathcal{G}^m_{\mathcal{W},\mathcal{F}}$ while utilizing utility rule f, $\mathrm{PoA}(\mathcal{G}^m_{\mathcal{W},\mathcal{F}})$, can be written as

$$\begin{aligned} &\min_{G\in\mathcal{G}^m_{\mathcal{W},\mathcal{F}}} \frac{W(a^{\mathrm{Ne}})}{W(a^{\mathrm{opt}})} \\ &\text{s.t. } U_i(a^{\mathrm{Ne}}; d) \geq U_i(a_i^{\mathrm{opt}}, a_{-i}^{\mathrm{Ne}}; d) \quad \forall i \in N. \end{aligned} \tag{A1}$$

where $G = (N, \mathcal{A}, \{U_i\}_{i\in N}, W)$ encodes all of the information about a problem instance. This program is not efficient to solve in general, however, we will make use of a parameterization that will greatly ease the computation of the price of anarchy. First, we will modify (A1) by normalizing $W(a^{\mathrm{Ne}}) = 1$, which can be done by homogeneously scaling each resource value and will not alter the problems price of anarchy. Next, we relax the equilibrium constraint from holding for every agent $i \in N$ to only hold as a summation over all agents, i.e., $\sum_{i\in N} U_i(a^{\mathrm{Ne}}; d) - U_i(a_i^{\mathrm{opt}}, a_{-i}^{\mathrm{Ne}}; d) \geq 0$. Note, that this relaxation will cause the new program to provide a lower bound for the original, however we will show that this bound is tight. Finally, we take the reciprocal of the objective and turn the minimization problem into a maximization problem. The new program, which the solution of will be a lower bound for the original, can be written

$$\begin{aligned} &\max_{G\in\mathcal{G}^m_{\mathcal{W},\mathcal{F}}} \quad W(a^{\mathrm{opt}}) \\ &\text{s.t.} \sum_{i\in N} U_i(a^{\mathrm{Ne}}; d) - U_i(a_i^{\mathrm{opt}}, a_{-i}^{\mathrm{Ne}}; d) \geq 0, \\ &\qquad W(a^{\mathrm{Ne}}) = 1. \end{aligned} \tag{A2}$$

Now, we make use of a parameterization that was also described in the proof of Theorem 2. Each resource is given a label (x_r, y_r, z_r, d_r) defined by $x_r = |a^{\mathrm{Ne}} \backslash a^{\mathrm{opt}}|_r$, $z_r = |a^{\mathrm{opt}} \backslash a^{\mathrm{Ne}}|_r$, $y_r = |a^{\mathrm{opt}} \cap a^{\mathrm{Ne}}|_r$, and d_r is the number of stubborn agents. The label denotes the number of agents that utilize a resource in only their Nash action x_r, only their optimal action z_r, or both y_r. The set of all such labels is $\mathcal{I}_n = \{(x, y, z) \in \mathbb{N}^3_{\geq 0} \mid 1 \leq x + y + z \leq n\}$. For each label we define a parameter $\theta(x, y, z, d) = \sum_{r\in\mathcal{R}(x,y,z,d)} v_r$, where $\mathcal{R}(x, y, z, d)$ is the set of resources with label (x, y, z, d). We can express several quantities using this parameterization as follows:

[2] This can be seen by transforming each game $G \in \mathcal{G}$ into one with two actions by removing all actions but the worst equilibrium a^{Ne} and the optimal allocation a^{opt}. Because a^{Ne} remains a Nash equilibrium, the price of anarchy is unchanged.

$$\begin{aligned}
\sum_{i\in N} U_i(a^{\mathrm{Ne}}; d) &= \sum_{r\in\mathcal{R}} v_r[(x_r+y_r)f(x_r+y_r+d_r)] \\
&= \sum_{x,y,z,d} (x+y)f(x+y+d)\theta(x,y,z,d), \\
\sum_{i\in N} U_i(a_i^{\mathrm{opt}}, a_{-i}^{\mathrm{Ne}}; d) &= \sum_{x,y,z,d} [yf(x+y+d)+zf(x+y+d+1)]\theta(x,y,z,d), \\
W(a^{\mathrm{Ne}}) &= \sum_{x,y,z,d} w(x+y)\theta(x,y,z,d), \\
W(a^{\mathrm{opt}}) &= \sum_{x,y,z,d} w(z+y)\theta(x,y,z,d).
\end{aligned}$$

Note that we write the sum over all labels in $\mathcal{I}_n$ as $\sum_{x,y,z,d}$ for brevity. Rewriting (A2) using this parameterization gives

$$\begin{aligned}
p^\star = \max_{\theta\in\mathbb{R}^{|\mathcal{I}_n|}} &\sum_{x,y,z,d} w(z+y)\theta(x,y,z,d) \\
\text{s.t.} &\sum_{x,y,z,d} [xf(x+y+d)-zf(x+y+d+1)]\theta(x,y,z,d) \geq 0, \\
&\sum_{x,y,z,d} w(x+y)\theta(x,y,z,d) = 1, \\
&\theta \geq 0.
\end{aligned} \tag{A3}$$

As discussed when introducing (A2), $p^\star$ offers a lower bound on the price of anarchy. We further show that using the solution to (A3), $\theta^\star$, one can construct a game whose price of anarchy is a tight upper bound.

For each label (x, y, z, d) such that $\theta^\star(x, y, z, d) > 0$, introduce n resources each with value $\theta^\star(x, y, z, d)/n$. As in Fig. 6, define each players action set to cover $x + y$ of these resources in their equilibrium action a_i^{Ne} and $z + y$ of these resources in their optimal action a_i^{opt} where y of the resources are in both actions. By considering the n resources in a ring, and offsetting each agents action sets by one resource, each agent can experience this set of resources symmetrically. Finally, let d stubborn agents be placed on each of these resources. If this is repeated for each label, then one can observe that player i will have utility

$$\begin{aligned}
U_i(a^{\mathrm{Ne}}; d) &= \sum_{x,y,z,d} (x+y)f(x+y+d)\theta(x,y,z,d) \\
&\geq \sum_{x,y,z,d} [yf(x+y+d)+zf(x+y+d+1)]\theta(x,y,z,d) \\
&= U_i(a_i^{\mathrm{opt}}, a_{-i}^{\mathrm{Ne}}; d),
\end{aligned}$$

where the inequality holds from the constraint in (A3) and $\theta^\star$ being a feasible solution; thus a^{Ne} is a Nash equilibria, and the price of anarchy of this game is at most $\frac{W(a^{\mathrm{Ne}})}{W(a^{\mathrm{opt}})} = \frac{\sum_{x,y,z,d} w(x+y)\theta^\star(x,y,z,d)}{\sum_{x,y,z,d} w(z+y)\theta^\star(x,y,z,d)} = \frac{1}{\sum_{x,y,z,d} w(z+y)\theta^\star(x,y,z,d)} = \frac{1}{p^\star}$, where the second equality holds from the constraint in (A3). The constructed game therefore offers an upper bound on the price of anarchy of $1/p^\star$, the solution to (A3), offers a matching lower bound, proving the bound is tight.

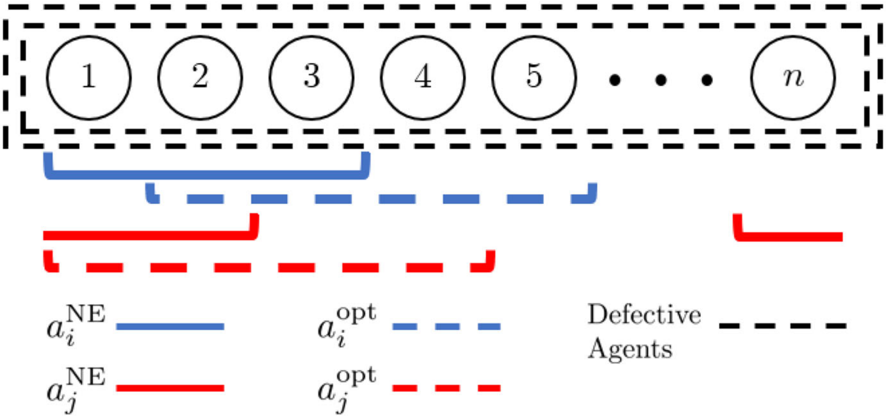


Fig. 6 Game construction for resource allocation problems utilizing the solution to (A1). For each tuple (x, y, z, d), n resources are created with value $\theta^\star(x, y, z, d)/n$. For a resource with label (x, y, z, d), design the action set of agent i to utilize the first $x + y$ of these resources in their first action, a_i^{Ne}, and the $x + 1$ to $x + y + z$ resources in their other action a_i^{opt}. For the proceeding agent, follow the same process but increasing the index of the starting resource by 1. If the agent were to use the non-existent $n + 1$ or greater resource, start the assignment from the beginning, essentially forming a ring. Once each action set is assigned for all n agents, each resource will be used by $x + y$ agents in the action a^{Ne} and $y + z$ agents in a^{opt}, matching the label it was assigned. We can observe that a^{Ne} is a Nash equilibrium from the constraints in (A1). For a similar, but more detailed explanation, see [17]

Notice that (A3) is a linear program with decision variable θ. Next we find the dual of (A3) to be

$$\begin{aligned} d^\star = &\min_{\lambda \geq 0, \mu \in \mathbb{R}_{\geq 0}} \mu \\ &\text{s.t. } w(z + y) - \mu w(x + y) \\ &\quad + \lambda[x f(x + y + d) - z f(x + y + d + 1)] \leq 0 \\ &\qquad \forall (x, y, z) \in \mathcal{I}_n,\ d \in \{1, \ldots, m\}. \end{aligned} \tag{A4}$$

Because (A3) is a linear program, and thus convex, by the principle of strong duality, $d^\star = p^\star = \mathrm{PoA}(\mathcal{G}^m_{\mathcal{W},\mathcal{F}})^{-1}$. Finally, to optimize the price of anarchy over local utility rules, we need only minimize (A4) over $f : [n + m] \to \mathbb{R}_{\geq 0}$, which can be treated as a vector in $\mathbb{R}^{n+m}$. Allowing f to be a decision variable in (A4) would cause each constraint to be bilinear in $f(i)$ and λ; however, every occurrence of λ is multiplied by an $f(i)$ for some i and vice versa, and therefore, the two decision variables can be combined into one giving a program of the form (9).

Finally, we note that (i) an optimal utility rule can be composed as the optimal utility rule for each basis function, i.e., for a resource with value $w_r = \sum_{b=1}^B \alpha_b w_b$ for some $\{\alpha_b\}_{b=1}^B$, then $f_r^{\mathrm{opt}} = \sum_{b=1}^B \alpha_b f_b^{\mathrm{opt}}$ where f_b^{opt} is the optimal utility rule for the basis function w_b described prior, and (ii) the worst case price of anarchy over the set of games with resource value functions in $\mathcal{W} = \{\sum_{b=1}^B \alpha_b w_b | \alpha_b \geq 0\ \forall b \in [B]\}$, is equal to the maximum of the sets of games with just one of these basis functions. These two observations have been shown in [5, 18] and follow identically here. This gives the final form of the optimal local utility design and the associated performance guarantee. □

Appendix B: Proof of Theorem 2

In this appendix, we give the full proof of Theorem 2 as well as several supporting lemmas. As in the proof of Proposition 1, we restrict our search to games where each player has two actions $\mathcal{A} = \{a_i^{\text{Ne}}, a_i^{\text{opt}}\}$ and note that the price of anarchy over this class is the same as the original with larger agent action sets. The price of anarchy bounds in (13) and (14) is tight along the Pareto-optimal frontier. To prove that each is an upper bound, we will make use of several examples; three structures of parameterized problem instances are shown in Fig. 7a–c. To show that these are lower bounds, we will make use of smoothness inequalities introduced in [19]. If, given a utility rule f, each Nash equilibria $a^{\text{Ne}} \in \text{NE}(G_f)$ satisfies

$$W(a^{\text{Ne}}) \geq \lambda \cdot W(a^{\text{opt}}) + \mu \cdot W(a^{\text{Ne}}), \tag{B5}$$

for some $\lambda, \mu \in \mathbb{R}$, then the price of anarchy will satisfy $\text{PoA}(\mathcal{G}_f) \geq \frac{\lambda}{1-\mu}$. We will provide lower bounds by finding values of λ and μ for different settings (e.g., with and without stubborn agents); often, to do so, we will utilize the fact that the welfare of a Nash equilibria can be lower bounded by

$$W(a^{\text{Ne}}) \geq \sum_{i \in N} u_i(a_i^{\text{opt}}, a_{-i}^{\text{Ne}}) - \sum_{i \in N} u_i(a^{\text{Ne}}) + W(a^{\text{Ne}}), \tag{B6}$$

which holds from the definition of a Nash equilibrium (1) where $u_i(a_i^{\text{opt}}, a_{-i}^{\text{Ne}}) \leq u_i(a^{\text{Ne}})$ for all $i \in N$, implying $\sum_{i \in N} u_i(a_i^{\text{opt}}, a_{-i}^{\text{Ne}}) - \sum_{i \in N} u_i(a^{\text{Ne}}) \leq 0$. Additionally, using the parameterization discussed in Section 3, where, in an allocation $(a, \overline{a})$, each resource $r \in \mathcal{R}$ is given a label (x_r, y_r, z_r, d_r) defined by $x_r = |a^{\text{Ne}} \backslash a^{\text{opt}}|_r$, $z_r = |a^{\text{opt}} \backslash a^{\text{Ne}}|_r$, $y_r = |a^{\text{opt}} \cap a^{\text{Ne}}|_r$, and d_r is the number of stubborn agents, where for two joint actions $a, a' \in \mathcal{A}$, $|a \backslash a'|_r$ is the number of agents that utilize resource r in action a but not a' and $|a \cap a'|_r$ is the number of agents that utilize resource r in both a and a'. This parameterization allows us to write $W(a^{\text{Ne}}) = \sum_{r \in \mathcal{R}} v_r \mathbb{1}_{[x_r+y_r]}$ and $W(a^{\text{opt}}) = \sum_{r \in \mathcal{R}} v_r \mathbb{1}_{[y_r+z_r]}$; additionally, (B6) can be rewritten as

$$\begin{aligned} W(a^{\text{Ne}}) \geq \sum_{r \in \mathcal{R}} v_r [&z_r f(x_r + y_r + d_r + 1) \\ &- x_r f(x_r + y_r + d_r) + \mathbb{1}_{[x_r+y_r]}], \end{aligned} \tag{B7}$$

where the welfare function $w(x) = \mathbb{1}_{[x]}$ is the indicator function that the argument is greater than zero in covering games. Manipulating the right hand side of (B7) into the form of (B5) will be the primary method of lower bounding the price of anarchy of a utility rule f in a class of games.

From [6], the optimal utility rule in covering games with no stubborn agents and arbitrarily many regular agents $\mathcal{G}^0$ is

$$f^0(j) := (j-1)! \frac{e - \sum_{i=0}^{j-1} \frac{1}{i!}}{e-1} \quad \forall j \geq 1, \tag{B8}$$

and $f^0(0) = 0$. This can also be seen by taking n to infinity in (12). The performance guarantee of f^0 is $\text{PoA}(\mathcal{G}_{f^0}^0) = 1 - \frac{1}{e}$, which can be seen from the following lemma.

Lemma 1 (Gairing [6]) *In the class of problems $\mathcal{G}^0$, with utility rule*

$$f^0(j) = (j-1)! \frac{e - \sum_{i=0}^{j-1} \frac{1}{i!}}{e-1} \quad \forall j \geq 1, \tag{B9}$$

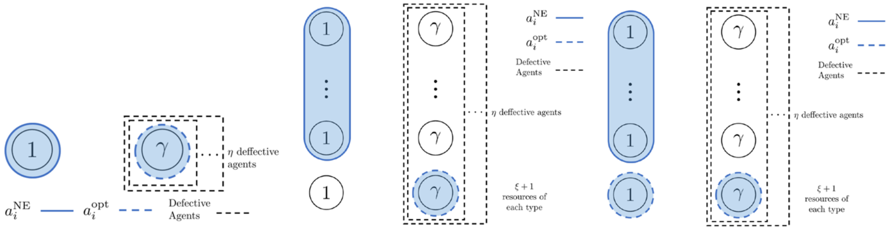


Fig. 7 **a** Example A: G^A A problem instance with one agent having two choices: a resource with value one and a resource with value γ covered by η defective agents. When $\gamma \le 1/f(\eta+1)$, the agent may pick the resource of value one in equilibrium leading to $\mathrm{PoA}(G_f^A) = \frac{1}{\gamma} \ge f(\eta+1)$. **b** Example B: G^B A problem with $\xi+1$ agents each with two choices: selecting ξ resources of value 1, a_i^{Ne}, or one resource of value γ with η defective agents, a_i^{opt}. The agents' equilibrium and optimal actions are distinct from one another, implying in the allocation a^{Ne}, xi agents cover each resource of value 1, and in a^{opt}, each resource of value γ is covered by one agent. When $\gamma \le \frac{\xi f(\xi)}{f(\eta+1)}$, a^{Ne} is an equilibrium allocation with $\mathrm{PoA}(G_f^B) = \frac{1}{\gamma}$. **c** Example C: G^C A problem with $\xi+1$ agents each with two choices: selecting ξ resources of value 1, a_i^{Ne}, or the remaining resource of value 1 and one resource of value γ with η defective agents, a_i^{opt}. The agents' equilibrium and optimal actions are distinct from one another, implying in the allocation a^{Ne}, ξ agents cover each resource of value 1, and in a^{opt}, every resource is covered by one agent. When $\gamma \le \frac{\xi f(\xi) - f(\xi+1)}{f(\eta+1)}$, a^{Ne} is an equilibrium allocation with $\mathrm{PoA}(G_f^C) = \frac{1}{1+\gamma}$

Equation (B5) *is satisfied with* $\lambda = 1$ *and* $\mu = -1/(e-1)$.

This utility rule is useful in constructing the optimal utility rules in the setting with stubborn agents. Additionally, the following claim is useful in proving several lower-bounds.

Lemma 2 *The local utility rule* f^0 *defined in* (B8) *satisfies*

$$jf^0(j) - f^0(j+1) = \frac{1}{e-1} \quad \forall j \in \mathbb{N}. \tag{B10}$$

Proof The claim can be proven directly by substitution:

$$\begin{aligned} jf^0(j) - f^0(j+1) &= j\left((j-1)!\frac{e - \sum_{i=0}^{j-1}\frac{1}{i!}}{e-1}\right) - (j)!\frac{e - \sum_{i=0}^{j}\frac{1}{i!}}{e-1} \\ &= \frac{j!}{e-1}\left(\sum_{i=0}^{j}\frac{1}{i!} - \sum_{i=0}^{j-1}\frac{1}{i!}\right) = \frac{1}{e-1}. \end{aligned}$$

□

The following several lemmas will define and quantify the smoothness coefficients of some useful local utility rules.

Lemma 3 *In the class of problems* $\mathcal{G}^m$, *with utility rule*

$$\overline{f}^m(j) = \begin{cases} \frac{m}{j} f^0(m), & \text{if } j \in \{1, \dots, m\} \\ f^0(j), & \text{otherwise,} \end{cases} \tag{B11}$$

Equation (B5) *is satisfied with* $\lambda = f^0(m+1)$ *and* $\mu = 1 - mf^0(m)$.

Proof Let $\mathcal{R}_a \subset \mathcal{R}$ be the set of all resources where $x_r + y_r + d_r \geq m + 1$ and let $\mathcal{R}_b \subset \mathcal{R}$ be the set of all resources where $x_r + y_r + d_r \leq m$, forming a partition of $\mathcal{R}$.

For the resources in $\mathcal{R}_a$,

$$\sum_{r\in\mathcal{R}_a} v_r[z_r f^0(x_r + y_r + d_r + 1) - x_r f^0(x_r + y_r + d_r) + 1]$$

$$\geq \sum_{r\in\mathcal{R}_a} v_r[(z_r + y_r) f^0(x_r + y_r + d_r + 1) - (x_r + y_r + d_r) f^0(x_r + y_r + d_r) + 1] \tag{B12a}$$

$$\geq \sum_{r\in\mathcal{R}_a} v_r[f^0(x_r + y_r + d_r + 1)\mathbb{1}_{[z_r+y_r]} - (x_r + y_r + d_r) f^0(x_r + y_r + d_r) + 1] \tag{B12b}$$

$$\geq \sum_{r\in\mathcal{R}_a} v_r\left[f^0(x_r + y_r + d_r + 1)\mathbb{1}_{[z_r+y_r]} - f^0(x_r + y_r + d_r + 1) - \frac{1}{e-1} + 1\right] \tag{B12c}$$

$$\geq \sum_{r\in\mathcal{R}_a} v_r\left[f^0(m + 1)\mathbb{1}_{[z_r+y_r]} - f^0(m + 1) - \frac{1}{e-1} + 1\right] \tag{B12d}$$

$$= \sum_{r\in\mathcal{R}_a} v_r[f^0(m + 1)\mathbb{1}_{[z_r+y_r]} + (1 - mf^0(m))\mathbb{1}_{[x_r+y_r]}], \tag{B12e}$$

where (B12a) and (B12d) hold from f^0 decreasing, (B12b) holds from f^0 positive, and (B12c) and (B12e) hold from Lemma 2.

For the resources in $\mathcal{R}_b$,

$$\sum_{r\in\mathcal{R}_b} v_r[z_r \overline{f}^m(x_r + y_r + d_r + 1) - x_r \overline{f}^m(x_r + y_r + d_r) + \mathbb{1}_{[x_r+y_r]}]$$

$$= \sum_{r\in\mathcal{R}_b} v_r\left[\frac{z_r}{x_r + y_r + d_r + 1}(mf^0(m)) - \frac{x_r}{x_r + y_r + d_r} mf^0(m) + \mathbb{1}_{[x_r+y_r]}\right] \tag{B13a}$$

$$\geq \sum_{r\in\mathcal{R}_b} v_r\left[(z_r + y_r) f^0(m + 1) - \frac{x_r}{x_r + y_r + d_r}(f^0(m + 1) + \frac{1}{e-1}) + \mathbb{1}_{[x_r+y_r]}\right] \tag{B13b}$$

$$\geq \sum_{r\in\mathcal{R}_b} v_r\left[f^0(m + 1)\mathbb{1}_{[z_r+y_r]} + \left(1 - f^0(m + 1) - \frac{1}{e-1}\right)\mathbb{1}_{[x_r+y_r]}\right] \tag{B13c}$$

$$= \sum_{r\in\mathcal{R}_b} v_r[f^0(m + 1)\mathbb{1}_{[z_r+y_r]} + (1 - mf^0(m))\mathbb{1}_{[x_r+y_r]}]. \tag{B13d}$$

where (B13b) holds from Lemma 2 and (B13c) holds from $x_r/(x_r + y_r + d_r) \leq 1$, providing the same lower bound for the price of anarchy.

It follows that $\lambda = f^0(m + 1)$ and $\mu = 1 - mf^0(m)$ satisfy (B5). □

Lemma 4 *In the class of problems $\mathcal{G}^0$, with utility rule*

$$\overline{f}^m(j) = \begin{cases} \frac{m}{j} f^0(m), & \text{if } j \in \{1, \ldots, m\} \\ f^0(j), & \text{otherwise,} \end{cases} \tag{B14}$$

Equation (B5) *is satisfied with $\lambda = mf^0(m)$ and $\mu = \frac{e-2}{e-1} - mf^0(m)$.*

Proof Let $\mathcal{R}_c \subset \mathcal{R}$ denote the set of resources where $x_r > 0$ or $y_r > 0$, and let $\mathcal{R}_d \subset \mathcal{R}$ be the set of resources where $x_r = y_r = 0$. First recall the bound from (B12e) and (B13d) that together give

$$\sum_{r \in \mathcal{R}} v_r [z_r \overline{f}^m(x_r + y_r + 1) - x_r \overline{f}^m(x_r + y_r) + \mathbb{1}_{[x_r + y_r]}] \tag{B15}$$

$$\geq \sum_{r \in \mathcal{R}} v_r [f^0(m+1) \mathbb{1}_{[z_r + y_r]} + (1 - mf^0(m)) \mathbb{1}_{[x_r + y_r]}], \tag{B16}$$

in the special case where $d_r = 0$, as is the case for games the class $\mathcal{G}^0$. For the set $\mathcal{R}_c$,

$$\begin{aligned} &\sum_{r \in \mathcal{R}_c} v_r [f^0(m+1) \mathbb{1}_{[z_r + y_r]} + (1 - mf^0(m)) \mathbb{1}_{[x_r + y_r]}] \\ &= \sum_{r \in \mathcal{R}_c} v_r \left[\left(mf^0(m) \mathbb{1}_{[z_r + y_r]} + \frac{1}{e-1} \right) + (1 - mf^0(m)) \mathbb{1}_{[x_r + y_r]} \right] && \text{(B17a)} \\ &\geq \sum_{r \in \mathcal{R}_c} v_r \left[(mf^0(m)) \mathbb{1}_{[z_r + y_r]} + \frac{1}{e-1} + (1 - mf^0(m)) \mathbb{1}_{[x_r + y_r]} \right] && \text{(B17b)} \\ &= \sum_{r \in \mathcal{R}_c} v_r \left[(mf^0(m)) \mathbb{1}_{[z_r + y_r]} + \left(\frac{e-2}{e-1} - mf^0(m) \right) \mathbb{1}_{[x_r + y_r]} \right], && \text{(B17c)} \end{aligned}$$

where (B17a) holds from Lemma 2, (B17b) holds from $\mathbb{1}_{[x]} \leq x$ for all non-negative integer x, and (B17c) holds from definition of $\mathcal{R}_c$ that $\mathbb{1}_{[x_r + y_r]} = 1$. For the remaining resources in $\mathcal{R}_d$,

$$\begin{aligned} &\sum_{r \in \mathcal{R}_d} v_r [z_r \overline{f}^m(x_r + y_r + 1) - x_r \overline{f}^m(x_r + y_r) + \mathbb{1}_{[x_r + y_r]}] \\ &= \sum_{r \in \mathcal{R}_d} v_r [z_r f^0(1)] \geq \sum_{r \in \mathcal{R}_d} v_r [z_r m f^0(m)] && \text{(B18a)} \\ &\geq \sum_{r \in \mathcal{R}_d} v_r \left[(mf^0(m)) \mathbb{1}_{[z_r + y_r]} + \left(\frac{e-2}{e-1} - mf^0(m) \right) \mathbb{1}_{[x_r + y_r]} \right], && \text{(B18b)} \end{aligned}$$

where (B18a) holds from the definition of $\overline{f}^m$ and $mf^0(m) < f^0(1) = 1$, and (B18b) holds from f^0 positive and $x_r = y_r = 0$. From (B17c) and (B18b), $\lambda = mf^0(m)$ and $\mu = \frac{e-2}{e-1} - mf^0(m)$ satisfy (B5). □

Lemma 5 *In the class of problems $\mathcal{G}^m$, with utility rule*

$$f^0(j) = (j-1)! \frac{e - \sum_{i=0}^{j-1} \frac{1}{i!}}{e-1} \quad \forall j \geq 1, \tag{B19}$$

Equation (B5) *is satisfied with $\lambda = f^0(m+1)$ and $\mu = 0$.*

Proof As in Lemma 3, let $\mathcal{R}_a \subset \mathcal{R}$ be the set of all resources where $x_r + y_r + d_r \geq m + 1$ and let $\mathcal{R}_b \subset \mathcal{R}$ be the set of all resources where $x_r + y_r + d_r \leq m$, forming a partition of $\mathcal{R}$. For the resources in the set $\mathcal{R}_a$, follow the steps of (B12a)–(B12e) and note that $jf^0(j) \leq 1$ for all j, therefore (B12e) is further lower-bounded by

$$\sum_{r \in \mathcal{R}_a} v_r [\mathbb{1}_{[z_r + y_r]} f^0(m+1)]. \tag{B20}$$

For the resources in $\mathcal{R}_b$,

$$\begin{aligned}
&\sum_{r \in \mathcal{R}_b} v_r [z_r f^0(x_r + y_r + d_r + 1) - x_r f^0(x_r + y_r + d_r) + \mathbb{1}_{[x_r + y_r]}] \\
&\quad = \sum_{r \in \mathcal{R}_b} v_r [(z_r + y_r) f^0(x_r + y_r + d_r + 1) && \text{(B21a)} \\
&\quad\quad - (x_r + y_r) f^0(x_r + y_r + d_r) + \mathbb{1}_{[x_r + y_r]}] \\
&\quad \geq \sum_{r \in \mathcal{R}_b} v_r [(z_r + y_r) f^0(x_r + y_r + d_r + 1)] && \text{(B21b)} \\
&\quad \geq \sum_{r \in \mathcal{R}_b} v_r [\mathbb{1}_{[z_r + y_r]} f^0(m+1)], && \text{(B21c)}
\end{aligned}$$

where (B21a) holds from f^0 decreasing, (B21b) holds from $jf^0(j) \leq 1$ for all $j \geq 0$, and (B21c) holds from $\mathbb{1}_{[x]} \leq x$ for all non-negative integer x. From (B20) and (B21c), $\lambda = f^0(m+1)$ and $\mu = 0$ satisfy (B5). □

Proof of Theorem 2 To prove that the curve defined by (13) and (14) represents a Pareto-optimal frontier of the multi-criterion problem of minimizing $\mathrm{PoA}(\mathcal{G}_f^m)$ and $\mathrm{PoA}(\mathcal{G}_f^0)$, we first give a parameterized utility rule that draws the curve then show a tight lower and upper bound on it's price of anarchy, and finally show this utility rule is indeed Pareto-optimal. Let $f^t(j) = t\overline{f}^m(j) + (1-t)f^0(j)$ for some $t \in [0, 1]$, be a local utility rule parameterized by $t \in [0, 1]$. Through some rearranging, this is equivalent to (15). We will show the price of anarchy guarantees of this utility rule draw the Pareto-optimal frontier.

Part 1 Upper Bound We will give problem instances that upper bound the price of anarchy over the set $\mathcal{G}^m$ and $\mathcal{G}^0$ for the utility rule f^t. For the nominal price of anarchy, let $G^C \in \mathcal{G}^0$ be a covering game as described in Fig. 7c with $\eta = 0$, $\gamma = \frac{\xi f(\xi) - f(\xi+1)}{f(1)}$. By selecting $\xi \geq m + 1$ agents in the game (where m is the number of defective agents for which f^t is designed), from Lemma 2

$$\gamma = \frac{1}{(e-1)f^t(1)} = \frac{1}{(e-1)(tmf^0(m) + (1-t))}.$$

Defining $\Gamma_m = mf^0(m) - 1 = m!\frac{e - \sum_{i=0}^{m-1}\frac{1}{i!}}{e-1} - 1$, the price of anarchy of the described game is

$$\mathrm{PoA}(G^C_{f^t}) = \frac{1}{1+\gamma} = \frac{(e-1)(1+t\Gamma_m)}{1+(e-1)(1+t\Gamma_m)}.$$

Because $G^C \in \mathcal{G}^0$, $\mathrm{PoA}(\mathcal{G}^0_{f^t}) \leq \mathrm{PoA}(G^C_{f^t})$. For the price of anarchy in the perturbed agent setting, let $G^A \in \mathcal{G}^m$ be a covering game as described in Fig. 7a with $\eta = m$ and $\gamma = f^t(1)/f^t(m+1)$. From the definition of f^t and Lemma 2, the price of anarchy of this game, with utility rule f^t is

$$\mathrm{PoA}(G^A_{f^t}) = \frac{f^0(m+1)}{1+t(mf^0(m)-1)} = \frac{\Gamma_m + \frac{e}{e-1}}{1+t\Gamma_m}.$$

Because $G^A \in \mathcal{G}^m$, $\mathrm{PoA}(\mathcal{G}^m_{f^t}) \leq \mathrm{PoA}(G^A_{f^t})$. This provides our upper bounds for the price of anarchy over $\mathcal{G}^0$ and $\mathcal{G}^m$ while using the utility rule f^t.

Part 2 Lower Bound To lower bound the price of anarchy, we again look for coefficients λ, μ that satisfy (B5). From the definition of f^t, (B7) can be rewritten

$$\begin{aligned} W(a^{\mathrm{Ne}}) \geq \sum_{r\in\mathcal{R}} & t v_r[z_r \overline{f}^m(x_r+y_r+d_r+1) \\ & - x_r \overline{f}^m(x_r+y_r+d_r) + \mathbb{1}_{[x_r+y_r]}] \\ & + (1-t)v_r[z_r f^0(x_r+y_r+d_r+1) \\ & - x_r f^0(x_r+y_r+d_r) + \mathbb{1}_{[x_r+y_r]}], \end{aligned} \tag{B22}$$

where $\overline{f}^m$ is as defined in (B11). For any game in $\mathcal{G}^m$, from Lemmas 3 and 5, (B22) can be lower bounded by

$$f^0(m+1)\cdot W(a^{\mathrm{opt}}) + t(1-mf^0(m))\cdot W(a^{\mathrm{Ne}}),$$

producing for the lower bound on the price of anarchy of

$$\mathrm{PoA}(\mathcal{G}^m_{f^t}) \geq \frac{f^0(m+1)}{t(1-mf^0(m))} = \frac{\Gamma_m + \frac{e}{e-1}}{1+t\Gamma_m}. \tag{B23}$$

For the price of anarchy over the nominal setting $\mathcal{G}^0$ with utility law f^t, (14) needs to be lower bounded for the case where $d_r = 0$ for all $r \in \mathcal{R}$. From Lemmas 4 and 1, this lower bound is

$$\begin{aligned} &(tmf^0(m) + (1-t))W(a^{\mathrm{opt}}) \\ &+ \left(t\left(\frac{e-2}{e-1} - mf^0(m)\right) + (1-t)\frac{-1}{e-1}\right) W(a^{\mathrm{Ne}}). \end{aligned}$$

This gives a lower bound on the nominal price of anarchy while using f^t of

$$\begin{aligned} \mathrm{PoA}(\mathcal{G}^0_{f^t}) &\geq \frac{tmf^0(m)+(1-t)}{1-\left(t\left(\frac{e-2}{e-1}-mf^0(m)\right)+(1-t)\frac{-1}{e-1}\right)} \\ &= \frac{(e-1)(1+t\Gamma_m)}{1+(e-1)(1+t\Gamma_m)}. \end{aligned}$$

Part 3 Pareto-Optimality Consider a local utility rule f with nominal price of anarchy guarantee

$$\mathrm{PoA}(\mathcal{G}^0_f) > x \tag{B24}$$

for some $x \in [0, 1]$. Consider a game $G^C \in \mathcal{G}^0$ following Fig. 7c where $\eta = 0$ and $\xi = m+1$. If $\gamma = ((m+1)f(m) - f(m+2))/f(1)$, then

$$\text{PoA}(G_f^C) = \frac{1}{1 + \frac{1}{(e-1)f(1)}},$$

from the assumption that $f(j) = f^0(j)\ \forall j \geq m+1$ and Lemma 2. To satisfy the price of anarchy guarantee in (B24),

$$f(1) > \frac{x}{(e-1)(1-x)}. \tag{B25}$$

Now, consider the game $G^A \in \mathcal{G}^m$ described in Fig. 7a where $\eta = m$ and $\gamma = f(1)/f(m+1) = f(1)/f^0(m+1)$. The price of anarchy of this game is $\text{PoA}(G_f^A) = 1/\gamma$. From (B25),

$$\text{PoA}(G_f^A) < \frac{(e-1)f^0(m+1)(1-x)}{x}. \tag{B26}$$

In (B26), choose $x = \frac{(e-1)(1+t\Gamma_m)}{1+(e-1)(1+t\Gamma_m)}$ for some $t \in [0, 1]$ and

$$\text{PoA}(\mathcal{G}_f^m) \leq \text{PoA}(G_f^A) < \frac{\Gamma_m + \frac{e}{e-1}}{1 + t\Gamma_m} \tag{B27}$$

from the fact $\Gamma_m = f^0(m+1) + \frac{1}{e-1}$. The monotonicity of each price of anarchy expression shows the logic is reversible, matching the theorem. A similar argument could be followed for other values of the utility rule. $\square$

References

1. Bertizzolo L, D'Oro S, Ferranti L et al (2020) SwarmControl: an automated distributed control framework for self-optimizing drone networks. In: IEEE INFOCOM 2020—IEEE conference on computer communications, pp 1768–1777. https://doi.org/10.1109/INFOCOM41043.2020.9155231
2. Brown GW (1951) Iterative solution of games by fictitious play. In: Koopmans TC (ed) Activity analysis of production and allocation. Wiley, New York, pp 374–376
3. Brown R, Rossi F, Solovey K et al (2021) On local computation for network-structured convex optimization in multi-agent systems. IEEE Trans Control Netw Syst. https://doi.org/10.1109/TCNS.2021.3050129
4. Bullo F, Frazzoli E, Pavone M et al (2011) Dynamic vehicle routing for robotic systems. Proc IEEE 99(9):1482–1504. https://doi.org/10.1109/JPROC.2011.2158181
5. Chandan R, Paccagnan D, Marden JR (2019) When Smoothness is not enough: toward exact quantification and optimization of the price of anarchy. In: Proceedings of the IEEE conference on decision and control, pp 4041–4046. arXiv:1911.07823v2
6. Gairing M (2009) Covering games: approximation through non-cooperation. In: Internet and economics, pp 184–195. https://doi.org/10.1007/978-3-642-10841-9_18
7. Goemans M, Li E, Mirrokni V et al (2004) Market sharing games applied to content distribution in ad-hoc networks. pp 55–66. https://doi.org/10.1145/989459.989467
8. Grimsman D, Ali MS, Hespanha JP et al (2018) Impact of information in greedy submodular maximization. In: 2017 IEEE 56th Annual conference on decision and control, CDC 2017 2018-Janua(Cdc), pp 2900–2905. https://doi.org/10.1109/CDC.2017.8264080
9. Hannan J (1957) Approximation to Bayes risk in repeated plays. Contribution to the theory of games, 3rd edn. Princeton University Press, Princeton, pp 97–139
10. Jaleel H, Abbas W, Shamma JS (2019) Robustness of stochastic learning dynamics to player heterogeneity in games. In: 2019 IEEE 58th conference on decision and control (CDC), pp 5002–5007. https://doi.org/10.1109/CDC40024.2019.9029471

11. Kitano H, Tadokoro S, Noda I et al (1999) RoboCup Rescue: search and rescue in large-scale disasters as a domain for autonomous agents research. In: IEEE SMC'99 conference proceedings. 1999 IEEE international conference on systems, man, and cybernetics (Cat. No.99CH37028), vol 6, pp 739–743. https://doi.org/10.1109/ICSMC.1999.816643
12. Li N, Marden JR (2013) Designing games for distributed optimization. IEEE J Sel Top Sign Proces 7(2):230–242. https://doi.org/10.1109/JSTSP.2013.2246511
13. Marden JR (2017) The role of information in distributed resource allocation. IEEE Trans Control Netw Syst 4(3):654–664. https://doi.org/10.1109/TCNS.2016.2553363
14. Mei W, Friedkin NE, Lewis K et al (2016) Dynamic models of appraisal networks explaining collective learning. In: 2016 IEEE 55th conference on decision and control, CDC 2016, pp 3554–3559. https://doi.org/10.1109/CDC.2016.7798803, arXiv:1609.09546
15. Nugraheni CE, Abednego L (2016) Multi agent hyper-heuristics based framework for production scheduling problem. In: 2016 International conference on informatics and computing (ICIC), pp 309–313. https://doi.org/10.1109/IAC.2016.7905735
16. Paccagnan D, Gairing M (2021) In congestion games, taxes achieve optimal approximation. arXiv:2105.07480v1
17. Paccagnan D, Chandan R, Marden JR (2018) Distributed resource allocation through utility design—Part I: optimizing the performance certificates via the price of anarchy. CoRR. arXiv:1807.01333
18. Paccagnan D, Chandan R, Ferguson BL et al (2021) Optimal taxes in atomic congestion games. ACM Trans Econ Comput. https://doi.org/10.1145/3457168
19. Roughgarden T (2012) Intrinsic robustness of the price of anarchy. Commun ACM 55(7):116–123. https://doi.org/10.1145/2209249.2209274
20. Salazar M, Tsao M, Aguiar I et al (2019) A congestion-aware routing scheme for autonomous mobility-on-demand systems. In: European control conference
21. Shamma JS (2007) Cooperative control of distributed multi-agent systems. Wiley Online Library, New York
22. Vetta A (2002) Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions. In: The 43rd annual IEEE symposium on foundations of computer science, 2002 proceedings, pp 416–425. https://doi.org/10.1109/SFCS.2002.1181966
23. Yildiz E, Acemoglu D, Ozdaglar A et al (2011) Discrete Opinion dynamics with stubborn agents. Public choice: analysis of collective decision-making eJournal