Resource-Competitive Analysis:
A New Perspective on Attack-Resistant Distributed Computing

Seth Gilbert Valerie King Jared Saia Maxwell Young

Abstract

In the spirit of competitive analysis, approximation guarantees, and game-theoretic treatments, we introduce a fresh approach to evaluating the performance of attack-resistant algorithms in distributed systems. This new approach, which we call *resource-competitive analysis*, is concerned with the worst-case ratio of the cost incurred by an algorithm to the cost incurred by any adversarial strategy. Here, the notion of cost corresponds to any network resource such as bandwidth, computational power, or an onboard energy supply. An adversary who attacks the system is assumed to control and coordinate a large number of Byzantine users that can exhibit arbitrary deviation from any prescribed protocol; in other words, the adversary may select *any* strategy, whether it be rational or not.

In a homogeneous network where all devices, correct and Byzantine, are resource constrained, relative cost is an especially well-motivated metric. An adversary who successfully attacks the system will be penalized by incurring a significantly higher cost than that experienced by correct users. Consequently, the adversary is forced to either (i) cease her malicious behavior or (ii) rapidly deplete the resources of her Byzantine users in perpetrating her attack. For a multitude of well-known distributed denial-of-service (DDoS) scenarios, this type of guarantee can be extremely valuable. Indeed, the utility of this approach has already been demonstrated for settings involving wireless ad-hoc networks where devices are battery powered and, thus, energy-constrained. Ultimately, we believe that resource-competitive analysis constitutes a useful technique for mitigating Byzantine behavior and that this approach to designing attack-resistant algorithms will find application in many other areas of distributed computing.

He who wishes to fight must first count the cost.
— Ts'ao Kung, *The Art of War* by Sun Tzu [28]

1 Introduction

As the sun rose on Rome in mid-March 537 AD, the defenders perched upon the city fortifications were confronted with the sight of four colossal siege towers. Laboriously constructed by Ostrogothic engineers and mobilized by teams of oxen, these massive war machines slowly advanced on the Salarian Gate. Fewer than 5,000 Roman soldiers remained to repel the attack; a pitiful number in contrast to the roughly 45,000 Ostrogoths arrayed on the battlefield below eagerly anticipating a breaching of the city walls. Despite this dire situation, the famed Byzantine general charged with the defense of Rome, Flavius Belisarius, could not contain his laughter. As if this odd behavior was not enough to set them on edge, the soldiers were further demoralized as their general commanded that the enemy be permitted to advance without hindrance. However, when the towers had reached the city moat, Belisarius directed his archers to target not the Ostrogoth belligerents, but the oxen below, all of which were quickly felled. Left standing with no method to propel them the remaining distance, these elaborate siege towers were reduced to nothing more than useless ornaments on the battlefield [19].

While the defensive tactics of the Byzantine Empire may differ from our approaches to safeguarding modern-day distributed systems, this historical event illustrates an enduring truth regarding conflict: *Attacking a well-protected structure is more costly for the attackers than for the defenders*. When the structure in question is a city, Belisarius' actions provide an acute example of this principle — an aggressive siege attempt crippled in exchange for a few handfuls of arrows! When the structure is a collection of networked computers, we argue that the principle remains equally valid.

The ideas required to translate this key notion into the analytical domain are encompassed by a new approach to evaluating the effectiveness of attack-resistant distributed algorithms that we call *resource-competitive analysis*. At its heart, this method of analysis concerns itself with the worst-case ratio of the cost incurred by the algorithm to the cost incurred by an attacker who may adopt *any* strategy. In this position paper, we argue that this new metric can be used to inform algorithm design in distributed systems.

A Roadmap: The remainder of this position paper is comprised of four main sections. In Section 2, we motivate and introduce the notion of resource-competitive analysis. In Section 3, we provide several examples outside of distributed computing where adopting a relative-cost perspective has proven useful, often dramatically so. In Section 4, we apply this definition to problems from the area of wireless sensor networks and demonstrate that resource-competitive analysis yields promising solutions. Finally, we summarize our arguments in Section 5 and pose several open problems regarding both the full scope and limitations of this new approach.

2 Walking a Mile in the Adversary's Shoes

Over the past decade, Byzantine fault tolerance in distributed systems has consistently been a topic of great interest to the research community (for examples, see [1, 9, 20, 24] and surveys [29, 31]). In systems that lack reliable admission control or any central authority, a number of fundamental distributed tasks become extremely challenging when network devices or users — referred to hereafter as *nodes* — cannot be trusted to obey protocol. Typically, these Byzantine nodes are assumed to constitute a large portion of the network while colluding to disrupt the availability of some critical system functionality. To this end, a single adversary is often assumed to be coordinating the malicious actions of all Byzantine nodes.

In order to overcome such attacks, much of the work produced by the distributed computing community has focused on the number of faulty nodes that can be tolerated while minimizing the required time or communication overhead since these latter metrics translate into network costs. For example, a typical constraint is that the amount of resources required by correct nodes to execute an algorithm should be small relative to the network size n; typically, logarithmic or polylogarithmic in n. This approach makes sense for many settings where the correct nodes correspond to a homogeneous population of machines with relatively limited computational power or bandwidth. For example, in a peer-to-peer (P2P) network, we wish to ensure small local routing tables and low communication overhead for storage and retrieval operations; these properties should not be sacrificed in the face of Byzantine faults.

When the adversary is unconcerned with expending its own resources — perhaps because these resources are held in abundance or can be easily replenished — an attack-resistant algorithm that minimizes the burden on correct nodes is the best solution for which one can hope. However, what if the adversary suffers the same dearth of a common and critical resource?

In such a situation, this traditional approach to attack resistance now presents a one-sided picture. Crucially, it no longer makes sense to consider an adversary who may launch attacks at no cost to herself. Instead, in addition to emphasizing correctness and scalability, a more complete perspective comes from treating this scarce resource as a form of common currency in the system and gauging the performance of an algorithm by the relative costs inflicted upon both the correct and Byzantine nodes. If the costs to the latter are disproportionately high, then the algorithm has obvious value. Specifically, sustained attacks aimed at disrupting the availability of a critical functionality are no longer feasible since an adversary rapidly depletes her resources in engaging in such denial-of-service behaviour.

Defining Resource Competitiveness

In this position paper we provide a new notion of "competitiveness" that we call *resource competitiveness*. Each of n players p_i is classified as either *correct*, if its actions are prescribed by an algorithm $\mathcal{A}$, or *faulty* otherwise; define P as the set of correct players and F as the set of faulty players. The faulty players are assumed to collude and coordinate their attacks; therefore, we view them as being controlled by a single adversary. The resource budgets of the Byzantine players is assumed to be finite; however, the exact budgetary constraints may not be known.

Let $\mathcal{C}(\alpha, i)$ denote the resource expenditure (or cost) to a player p_i over an execution α. If p_i is correct, then $\mathcal{C}(\alpha, i)$ is the cost incurred by p_i for executing the actions prescribed by $\mathcal{A}$ in an execution α. Otherwise, p_i is faulty and $\mathcal{C}(\alpha, i)$ is the cost incurred by p_i for pursuing any arbitrary strategy in an execution α. The membership of P and F are not necessarily known to $\mathcal{A}$ *a priori* and may be, in part, selected adversarially.

We can now provide a concrete example of what it means to be resource competitive. Let $T = (\frac{1}{\rho} \cdot \sum_{p_j \,\in\, F} \mathcal{C}(\alpha, j))$ for a parameter $\rho > 1$. Then we can consider $\mathcal{A}$ to be resource competitive if $\max_{p_i \,\in\, P}\{\mathcal{C}(\alpha, i)\} \leq T + \tau$ for any execution α and some parameter $\tau > 0$. Ignoring τ for the moment, this means that $\mathcal{A}$ is resource competitive if the *maximum* cost incurred by any correct player is a ρ-factor less than the *sum* of the costs incurred by the faulty nodes; clearly, a large ρ is desirable. In terms of applicability, consider a network where maximizing the lifetime of all correct nodes is critical; this comparison between the maximum and aggregate costs motivates algorithms that minimize the worst-case relative cost to any single correct player.

Why do we require τ? Consider an execution where the faulty players incur zero cost. Clearly, in this case, no algorithm will allow the correct players to incur less cost than the adversary. This is to be expected since, even in the absence of malicious interference, the correct nodes must incur some unavoidable cost to perform an execution. The same reasoning applies when the faulty players incur small, but non-zero, cost in attacking the network. There must exist some τ where, for $T \leq \tau$ (ie. small attacks), each correct player will successfully terminate with a small *absolute* cost, but we may not spend less than the T spent by adversary. However, for $T \geq \rho\tau$ (ie. larger attacks), the term T dominates the right side of the resource-competitive equation and each correct player experiences less cost than the adversary. Clearly, a small τ is desirable.

In Section 4, we cite resource-competitive algorithms which require that each correct player initially incurs a small cost. If the faulty players are inactive or interfere very little, then each correct player will succeed. Otherwise, the correct players amplify their efforts throughout the execution, thus forcing the adversary to incur an increasingly larger cost in order to prevent their successful termination. This can be viewed as adaptive behaviour; the costs to the correct players increases as the adversary's costs increase and, ultimately, the former is smaller relative to the latter.

Another natural comparison is $\sum_{p_i \,\in\, P}\{\mathcal{C}(\alpha, i)\} \leq \frac{1}{\rho}(\sum_{p_j \,\in\, F} \mathcal{C}(\alpha, j)) + \tau$. For example, in a network where the primary concern is that at least one correct player survive the attack, perhaps to sound an alarm, we may be concerned that the aggregate cost over all correct players be less than the aggregate cost incurred by the adversary. Alternatively, a more stringent comparison of $\max_{p_i \,\in\, P}\{\mathcal{C}(\alpha, i)\} \leq \frac{1}{\rho}(\min_{p_j \,\in\, F}\{\mathcal{C}(\alpha, j)\}) + \tau$. For example, given that players have homogeneous finite resource budgets, such an algorithm implies that roughly ρ faulty players will exhaust their respective budgets for every one of the correct nodes that exhausts its budget.

We may also consider statistics such as the average or median cost. Furthermore, if a resource-competitive algorithm is randomized, then we can speak of cost in terms of expectation or with a high probability guarantee. Given this variety, we provide the following formal definition of resource competitiveness:

Definition 1. *Let $g(\cdot)$ and $a(\cdot)$ be functions that take as input the set of costs for the correct and faulty players, respectively. Let $T = \frac{1}{\rho} \cdot a(\{\mathcal{C}(\alpha, j)\}_{p_j \,\in\, F})$. Then, an algorithm $\mathcal{A}$ is (ρ, g, a, τ)-resource-competitive if $g(\{\mathcal{C}(\alpha, i)\}_{p_i \,\in\, P}) \leq T + \tau$ for any execution α and some $\tau > 0$. The parameter ρ is called the "resource-competitive ratio".*

When $g(\cdot)$, $a(\cdot)$, and τ are clear from the context, we simply state that $\mathcal{A}$ is *ρ-resource-competitive*. It is important to emphasize that ρ need not be a constant; in fact, a natural approach is to have ρ be a function of T as we will see in Section 4.

Finally, we note that Byzantine behaviour need not necessarily arise from intentionally malicious actions. For example, arbitrary faults may occur due to malfunctioning nodes. Treating such occurrences as having been engineered by an adversary simply allows us to prove guarantees under worst-case scenarios regardless of whether an adversary actually exists. In the case of malfunctioning nodes, a resource-competitive result provides guarantees on the level and duration of interference required to prevent successful network operations. Eventually, the malfunctioning nodes will expire and the correct nodes will succeed given their resource-competitive advantage.

3 Related Ideas — Towards Resource-Constrained Adversaries

The notion of inflicting higher relative costs on an opponent, with the ultimate goal of financial ruin, arguably has its analytical roots in economics. When asked whether he would buy a one-hundred dollar bullet-proof vest to protect himself from a five-cent bullet, Nobel Prize-winning economist Thomas Schelling says of the would be shooter that:

"He has wasted his money if the vest is cheap, made a splendid investment if my vest is expensive, and if asked what he accomplished by buying his bullet should have the good sense to say that he imposed a cost on me, not that he hoped to kill me and was frustrated." [25]

The concept of competitiveness has appeared in several areas of computer science. Indeed, in the analysis of algorithms, we often consider the "relative performance" of an algorithm — this notion underlies competitive analysis and approximation algorithms. One commonly analyzes the worst-case performance of an online algorithm $\mathcal{A}$ relative to an optimal algorithm $\mathcal{OPT}$ that has full information regarding the input [26]. A closely related technique for algorithm analysis is that of establishing approximation guarantees. In this case, a similar definition applies with the major difference being that $\mathcal{A}$ has full knowledge of the input *a priori*. Both notions allow us to measure performance relative to an optimal algorithm under worst-case inputs which are often thought of as having been selected by an adversary. In this context, resource-competitive analysis offers a new measure of relative performance or competitiveness where we may now be competing directly with truly malicious (or adversarially malfunctioning) entities.

Not surprisingly, the idea of competition between correct and faulty participants arises more explicitly in the area of security. In their influential paper "New Directions in Cryptography" [11], Diffie and Hellman state that the goal in designing a cryptographic system *"is to make the enciphering and deciphering operations inexpensive, but to ensure that any successful cryptanalytic operation is too complex to be economical."* This idea that an attacker is burdened by a disproportionate cost in attempting to break a cryptosystem underlies all modern-day cryptosystems. From a complexity perspective, inverting certain functions is believed to require enormous computational resources. By tuning a security parameter this cost can be so exorbitantly high that compromising the cryptosystem is believed to be infeasible.

A crucial difference between these cryptographic approaches and resource competitiveness is that, in the latter, it is not necessarily possible to obtain a ρ that is so large that the adversary is prevented from successfully attacking. However, by launching a successful attack, the adversary will be required to incur a cost that is much larger than that incurred by the correct nodes; we provide concrete examples in Section 4. In this sense, resource competitiveness can be seen as providing a deterrent rather than a guarantee on the security of the system.

Another difference is that the length of a private key is decided prior to the encryption of data. This roughly determines how much the adversary must spend in order to compromise the cryptosystem. In contrast, as discussed in Section 2, resource competitive algorithms are, in some sense, adaptive. While this has advantages, it also implies that the algorithm may require nodes to expend some unknown amount of resources that is a function of what the adversary spends. For homogeneous, resource-constrained networks, this property seems appropriate.

Examples From Distributed Systems

Moving to the area of distributed computing, there are a number of places where related ideas make an appearance. A primary example arises in settings where identities are cheap. Consequently, once a node is revealed to be misbehaving, it may leave and rejoin the system without penalty. Therefore, it is sensible to issue a cost for joining; for example, the use of a Turing test or a CAPTCHA [6, 17, 32], computational puzzles [15, 27], or even monetary penalties [8] have been proposed. These approaches impose a resource penalty in order to obtain a rate-limiting effect on the number of adversarial identities that can be placed in the system. Similarly, the social cost of establishing links between two nodes in a social network graph has been exploited to prove topological properties that mitigate Sybil attacks [33].

A related idea involves examining the efficiency of an attack. In the wireless domain, the *jamming gain* of a protocol roughly provides a measure of how long the adversary can delay the successful termination of the correct nodes by spending one unit of energy to jam the communication channel (see [7, 12]). Note that this definition focuses on the time delay experienced by nodes executing the algorithm; this does not necessarily translate into a cost since a node may not perform any expensive operations over this duration (for example, it may be idle). Conversely, knowledge of the costs to the adversary and correct players over an execution may not imply anything about the jamming gain of a protocol; such information depends on the details of the protocol (see Section 4). Therefore, it seems these two metrics are distinct.

As a final illustration, we note that real-world adversaries are already employing similar tactics in the form of distributed denial-of-service (DDoS) attacks. Here, an attacker typically makes use of a botnet to pummel a server with a large volume of requests. At the transport layer, such attacks are aimed at consuming the maximum number of available connections to the server. At the application layer, the requests themselves are expensive to service and the server is overwhelmed. These DDoS attacks illustrate the effectiveness of inflicting a disproportionate cost, in terms of bandwidth or computational power, on an opponent. On the other hand, a number of results aim at mitigate DDoS attacks by forcing a client to make a "down payment" in terms of bandwidth or computational power prior to receiving service (see [30] and references therein).

Overall, these examples illustrate an underlying movement towards quantifying a cost that an adversary must pay. In this sense, resource competitiveness quantifies an idea that has already implicitly arisen in isolated instances. Our goal is to make this idea explicit and argue that translating this idea — an idea that has been successfully applied in other areas of computer science — to the realm of distributed computing provides us a fresh and valuable approach to adversarial fault tolerance.

Are We Playing a Game?

Given the economic flavour of resource-competitive analysis, we contrast this approach with game theory [18, 23]. Here, nodes (or players) are assumed to be autonomous and governed by self-interested behaviour. Game theory provides us with yet another measure of competitiveness known as the "price of anarchy" which is the ratio of the worst-case Nash equilibrium to the global social optimum.

There are several differences in the approaches taken by resource-competitive analysis and game theory. In the former, each node either obeys protocol or it does not; in the latter, we typically consider self-interested players endeavoring to maximize their respective utility functions. Furthermore, incorporating Byzantine behaviour into game theory has proved challenging; indeed, this is a topic of recent interest (see [2]). The model of BAR (Byzantine, Altruistic, Rational) games [10] addresses these issues; indeed, the model is general enough to subsume traditional Byzantine approaches — in this context, resource-competitive analysis is concerned only with altruistic and Byzantine players. However, even in this model, the cost to a Byzantine player is difficult to quantify since doing so requires making assumptions about its utility function. In contrast, resource-competitive analysis is focused on quantifying the cost of a successful attack by the adversary without the need to specify an unknown utility function. It is this ability to quantify the adversary's cost that allows for the idea of depleting an adversary's (or malicious player's) resources.

4 Existing Resource-Competitive Results for Wireless Communication

In wireless sensor networks (WSNs), nodes correspond to battery-powered devices and energy consumption is a critical concern since, once depleted, it may be infeasible to replace the power supply. Furthermore, if we consider an adversary that compromises existing nodes or deploys its own nodes in the network, we have an attack model where the common constrained resource for both correct and Byzantine nodes is energy.

Communication in WSNs is notoriously unreliable and can easily be disrupted by an attacker who deliberately interferes with the wireless medium. Such *jamming attacks* have received significant attention from the research community given the ease of perpetrating such attacks and their effectiveness (see [31] and references therein). Since both sending on and listening to the channel is expensive, attempting to naively outspend a jamming attacker can be a losing strategy.

A fundamental problem is single-hop communication between two nodes, say Alice and Bob, in a WSN. Here, Alice wishes to guarantee transmission of a message m directly to Bob over a single communication channel. However, a jamming adversary wishes to prevent prevent communication. The main result in [14] is a Las Vegas communication protocol where, if the adversary spends T slots interfering with communication, the total expected cost to Alice and Bob is $O(T^{\varphi-1}+1) = O(T^{0.62}+1)$ prior to successfully communicating where φ is the golden ratio. Using Definition 1 in Section 2, $g(\cdot) = O(T^{\varphi-1})$ is the aggregate cost of both Alice and Bob, $a(\cdot)$ is the sum total cost T incurred by the adversary, $\tau = O(1)$, and $\rho = \Omega(T^{2-\varphi})$; note that ρ is a function of T here, as discussed earlier in Section 2.

A more recent resource-competitive algorithm for WSNs addresses a different attack model [13]. Here, a Monte Carlo protocol is given that allows for communication of a message m from a single sending node to n receiving nodes. Specifically, with high probability, all but a $(1 - \epsilon)$-fraction of the receivers obtain m even in the presence of $\Theta(n)$ Byzantine nodes. For an adversary that incurs a cost of T for interfering with communication, the sender and each of the receivers experience a cost of $\tilde{O}(T^{1/k} + 1)$ for a constant k chosen *a priori*. Again, using Definition 1, $g(\cdot)$ is the maximum individual cost over the sender and all receivers, $a(\cdot)$ is the sum total cost incurred by the $\Theta(n)$ Byzantine nodes, $\tau = \tilde{O}(1)$, and $\rho = \Omega(T^{(k-1)/k})$.

Finally, both protocols admit an analysis where, in order to have a constant probability of preventing successful termination within S slots, the adversary must jam $(1 - \delta)S$ slots for a small constant $\delta > 0$. Loosely interpreted: if the adversary spends one unit of energy, then she can expect to delay successful termination by roughly $1/(1 - \delta)$ slots. The correspondence is not exact since these protocols are randomized, but it does provide insight into how resource competitiveness can imply results related to jamming gain.

5 Conclusion

Regarding the fate of Rome, the efforts of Belisarius were ultimately successful in fending off the Ostrogothic siege and, in later years, greatly expanding Byzantium. Of course, in true Byzantine fashion, fearing Belisarius' growing popularity, the Emperor Justinian was later involved in having his star general arrested on false corruption charges [16]. However, politics aside, the defensive tactics of Belisarius illustrate a useful concept that has been employed, with less bloodshed, in the field of computer science. Indeed, the aim of this position paper has been to make explicit an idea whose effectiveness has already been demonstrated in isolated cases within the context of distributed computing. We feel that resource competitiveness is relevant to attack resistance and we believe it will be useful to the research community.

We have seen how this approach can be applied to attack-resistant single-hop communication. Can other canonical distributed computing problems be explored in this context? For example, can we find resource-competitive consensus protocols? Byzantine agreement? If so, what limits exist on the advantage yielded to the correct nodes? Conversely, the question of lower bounds on ρ remains largely open.

In the WSN domain is the attack model where, for every sufficiently large window of time, an adversary may jam the channel over a $(1 - \epsilon)$-fraction of the window where $\epsilon > 0$ is an arbitrarily small constant [5, 21, 22]. In some sense, such an adversary has its resources renewed in every new window, but is constrained within a single window. Can we apply a resource-competitive approach to such adversaries?

Another open problem involves using a resource-competitive approach to implement shared data structures in a Byzantine environment. Methods for doing this make use of large quorums to "read" and "write" to the data structure, and are thus open to denial-of-service type attacks where an adversary forces many messages to be sent by initiating, at relatively low cost, reads and writes [4].

Finally, the notion of resource competitiveness may provide a useful formalism for interpreting other results on mitigating attacks. For example, independent of [14], Asharaf *et al.* [3] propose a number of clever heuristic approaches aimed at "bankrupting" a jamming adversary. Can these results be formally analyzed using the framework of resource competitiveness? Another example is DDoS attacks in the client-server scenario. As mentioned in Section 3, many proposals for defending against such attacks rely on forcing the clients to incur larger costs. Perhaps these results can be interpreted via Definition 1; indeed, some preliminary results addressing DDoS attacks in the client-server scenario are discussed in [14].

Acknowledgements: We thank Elad Schiller for some useful discussion regarding game theory.

References

[1] Michael Abd-El-Malek, Gregory R. Ganger, Garth R. Goodson, Michael K. Reiter, and Jay J. Wylie. Fault-Scalable Byzantine Fault-Tolerant Services. In 20th *ACM Symposium on Operating Systems Principles (SOSP)*, pages 59–74, 2005.

[2] Ittai Abraham, Lorenzo Alvisi, and Joseph Y. Halpern. Distributed Computing Meets Game Theory: Combining Insights from Two Fields. *SIGACT News*, 42(2):69–76, June 2011.

[3] Farhana Ashraf, Yih-Chun Hu, and Robin Kravets. Demo: Bankrupting the Jammer. In *Proceedings of the* 9th *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2011.

[4] Hagit Attiya. Robust Simulation of Shared Memory: 20 Years After. *EATCS Distributed Computing Column*, 100:99–113, February 2010.

[5] Baruch Awerbuch, Andrea Richa, and Christian Scheideler. A Jamming-Resistant MAC Protocol for Single-Hop Wireless Networks. In *Proceedings of the* 27th *ACM Symposium on Principles of Distributed Computing (PODC)*, pages 45–54, 2008.

[6] Baruch Awerbuch and Christian Scheideler. Group Spreading: A Protocol for Provably Secure Distributed Name Service. In 31st *International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 183–195, 2004.

[7] Timothy Brown, Jesse James, and Amita Sethi. Jamming and Sensing of Encrypted Wireless Ad Hoc Networks. In *Proceedings of the* 7th *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 120–130, 2006.

[8] Miguel Castro, Peter Druschel, Ayalvadi Ganesh, Antony Rowstron, and Dan S. Wallach. Secure Routing for Structured Peer-to-Peer Overlay Networks. In 5th *Usenix Symposium on Operating Systems Design and Implementation (OSDI)*, pages 299–314, 2002.

[9] Miguel Castro and Barbara Liskov. Practical Byzantine Fault Tolerance and Proactive Recovery. *ACM Transactions on Computer Systems*, 20(4):398–461, 2002.

[10] Allen Clement, Jeff Napper, Harry Li, Jean-Philipe Martin, Lorenzo Alvisi, and Michael Dahlin. Theory of BAR Games. In *Proceedings of the* 26th *Annual ACM Symposium on Principles of Distributed Computing*, pages 358–359, 2007.

[11] Whitfield Diffie and Martin E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[12] Seth Gilbert, Rachid Guerraoui, and Calvin C. Newport. Of Malicious Motes and Suspicious Sensors: On the Efficiency of Malicious Interference in Wireless Networks. In *International Conference On Principles Of Distributed Systems (OPODIS)*, pages 215–229, 2006.

[13] Seth Gilbert and Maxwell Young. Making Evildoers Pay: Resource-Competitive Broadcast in Sensor Networks. Accepted to the 31th *Symposium on Principles of Distributed Computing (PODC)*, 2012.

[14] Valerie King, Jared Saia, and Maxwell Young. Conflict on a Communication Channel. In *Proceedings of the* 30th *Symposium on Principles of Distributed Computing (PODC)*, pages 277–286, 2011.

[15] Frank Li, Prateek Mittal, Matthew Caesar, and Nikita Borisov. SybilControl: Practical Sybil Defense with Computational Puzzles. arXiv:1201.2657, 2012.

[16] Lord Mahon. *The Life of Belisarius*. Westholme Publishing, 2005.

[17] A. Nambiar and M. Wright. Salsa: A Structured Approach to Large-Scale Anonymity. In *Proceedings of the* 13th *ACM Conference on Computer and Communications Security*, pages 17–26, 2006.

[18] Noam Nisan, Time Roughgarden, Éva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[19] Procopius. History of the Wars, English translation by Henry B. Dewing. `https://www.gutenberg.org/files/20298/20298-h/20298-h.htm`, 2007.

[20] Michael K. Reiter. The Rampart Toolkit for Building High-Integrity Services. In *International Workshop on Theory and Practice in Distributed Systems*, pages 99–110, 1995.

[21] Andrea Richa, Christian Scheideler, Stefan Schmid, and Jin Zhang. A Jamming-Resistant MAC Protocol for Multi-Hop Wireless Networks. In *Proceedings of the International Symposium on Distributed Computing (DISC)*, pages 179–193, 2010.

[22] Andrea Richa, Christian Scheideler, Stefan Schmid, and Jin Zhang. Competitive and Fair Medium Access Despite Reactive Jamming. In *Proceedings of the* 31^{st} *International Conference on Distributed Computing Systems (ICDCS)*, pages 507–516, 2011.

[23] Sara Robinson. The Price of Anarchy. *SIAM News*, 37(5):1–4, 2004.

[24] Rodrigo Rodrigues and Barbara Liskov. Rosebud: A Scalable Byzantine-Fault-Tolerant Storage Architecture. Technical Report TR/932, MIT LCS, December 2003.

[25] Thomas C. Schelling. The Strategy of Inflicting Costs. In *Issues in Defense Economics*, NBER Chapters, pages 105–128. National Bureau of Economic Research, Inc, February 1967.

[26] D. Sleator and R. Tarjan. Amortized Efficiency of List Update and Paging Rules. *Communications of the ACM*, 28(2):202–208, 1985.

[27] Florian Tegeler and Xiaoming Fu. SybilConf: Computational Puzzles for Confining Sybil Attacks . In *INFOCOM IEEE Conference on Computer Communications Workshops*, pages 1–2, 2010.

[28] Sun Tzu. *The Art of War, Translation by Lionel Giles*. El Paso Norte Press, 2005.

[29] Guido Urdaneta, Guillaume Pierre, and Maarten van Steen. A Survey of DHT Security Techniques. *ACM Computing Surveys*, 43(2):1–53, 2011.

[30] Michael Walfish, Mythili Vutukuru, Hari Balakrishnan, David Karger, and Scott Shenker. DDoS Defense by Offense. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pages 303–314, 2006.

[31] Maxwell Young and Raouf Boutaba. Overcoming Adversaries in Sensor Networks: A Survey of Theoretical Models and Algorithmic Approaches for Tolerating Malicious Interference. *IEEE Communications Surveys & Tutorials*, 13(4):617–641, 2011.

[32] Maxwell Young, Aniket Kate, Ian Goldberg, and Martin Karsten. Practical Robust Communication in DHTs Tolerating a Byzantine Adversary. In *International Conference on Distributed Computing Systems (ICDCS)*, pages 263–272, 2010.

[33] Haifeng Yu, Michael Kaminsky, Phillip B. Gibbons, and Abraham Flaxman. SybilGuard: Defending Against Sybil Attacks via Social Networks. *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, 36:267–278, August 2006.