Secure and efficient networks*

Oleh Stupak†

June 10, 2023

Abstract: This study aims to understand efficient network formation and optimal defensive resource distribution in the presence of an intelligent attacker. We present a two-player dynamic framework in which the Defender and the Attacker compete in a network formation and defence game with heterogeneous vertices' values. Such a model allows for studying the trade-off between network efficiency and security. Contrary to the literature, we find that a centrally protected star network does not yield the maximum payoff for the defending side in most circumstances, even being the most secure network formation. Additionally, it reveals a new type of network that often arises in an equilibrium of the games with limited defensive resources—a maxi-core network.

Keywords: *Strategic Network Formation, Conflict, Cybersecurity, Network Defence.*
JEL Classification: *D74, D85.*

1 Introduction

The scholarship of *economics of networks* generally focuses on a fundamental aspect: either network efficiency (Jackson, 2010) or network security (Dziubiński & Goyal, 2013; Goyal & Vigier, 2010). In most real-life scenarios, however, the issues of efficiency and security cannot be considered alone or at the expense of the other.

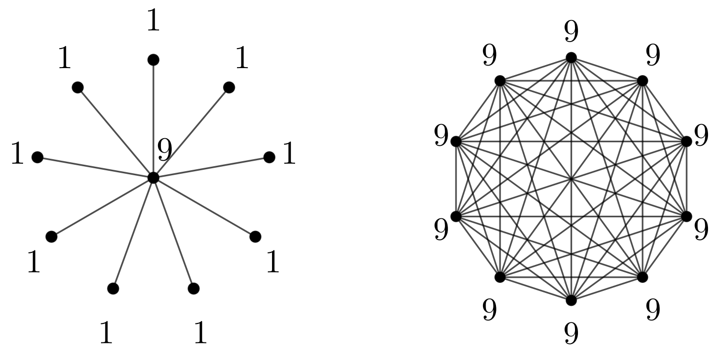
The fastest military supply chain is worthless during a war if it is not adequately protected with anti-aircraft defences, and can be easily disturbed by cheap unmanned

*We are grateful to Greg Taylor, Alexei Parakhonyak, Andrew Simpson, Daniel Arce, and Rainer Bohme for their invaluable contributions and suggestions. Thanks also to participants at various seminars and conferences for valuable comments and discussions.

†Corpus Christi College, University of Cambridge, os446@cam.ac.uk

combat aerial vehicles (UCAVs). Harsh social distancing measures can mitigate the spread of a virus but also reduce the density of social connections and severely damage business processes. Creating overconnected information communication technology (ICT) networks can significantly increase the data flow but also facilitate the hacker's navigation inside the enterprise's file system.

To illustrate the concepts of efficiency and security in a networked environment, we refer the reader to Figure 1. A star network (Figure 1a) is generally considered the most secure network: a significant share of its value is concentrated in the central node, making it easy to protect. However, any two peripheral nodes must overcome two links in order to communicate with each other, which makes it inefficient in some scenarios.



(a) Star network

secure

(b) Complete network

efficient

Figure 1: Examples of secure and efficient networks on nine nodes. Each node is valued according to the number of connections it has.

On the contrary, the complete network in Figure 1b is highly efficient. Each pair of nodes in this network are immediate neighbours, which allows for direct communication. However, complete networks are also the most insecure as it is hard to protect all high-value nodes when the defensive resources are scarce. The trade-off between security and efficiency is the central topic of this research.

More specifically, this study aims to understand how to construct an efficient network in the presence of an intelligent attacker. To tackle this problem, we develop a game in which two asymmetric players, the Defender and the Attacker, compete in network design and defence over two stages.

In the initial *wiring stage*, the Defender is endowed with some fixed number of nodes and is free to arrange them in any connected network structure by creating costly links. For instance, the Defender can represent an IT department that chooses an optimal information technology structure and hierarchical protocols, optimizes data and information exchange efficiency against the expected rate of security incidents.

At the beginning of the following *encounter stage*, the Attacker observes the network structure. For example, he can perform port scanning to retrieve the organisation of Internet Protocol (IP) address or use a specialised search engine to discover the underlying server structure of the organisation (e.g. Shodan Search Engine[1]). The Attacker and the Defender then engage in a simultaneous move game in which the Attacker attempts to compromise one of the vertices while the Defender tries to protect the network by distributing some scarce defensive resources among the vertices. If the Defender does not protect the vertex that the opponent attacked, the Attacker performs a successful attack and compromises the vertex, receiving its value, while the Defender receives the residual value of the network. On the contrary, the attack fails if the Defender protects a correct node, in which case the Defender keeps the full value of the network only to herself. For instance, if a hacker is trying to compromise the enterprise's network using a particular exploit, the cybersecurity specialist's success depends on whether she/he correctly predicted the attack vector and patched the vulnerable machines. The framework is stated formally and in greater detail at the beginning of Section 2.

The game utilises a Subgame Perfect Nash Equilibrium as a solution concept and is solved via the backward induction beginning at the encounter stage. The main technical challenge in the solution occurs in the wiring stage of the game. The stage can be stated as the Defender's maximisation problem in which she chooses a network which yields her the maximum expected payoff accounting for the overall network value and expected loss from an attack in the encounter stage. Formally it can be written as a constrained integer maximisation problem defined over a non-convex set. Problems of this type, in general, do not allow for direct analytical treatment. Thus, to characterise an equilibrium solution, we rely on the convex relaxation technique, which implicates the relaxation of integrality constraints and linearisation of non-convex constraints. This approach yields a simpler maximisation problem defined on a convex hull of the original set while maintaining the original form of the objective function. We then analytically identify the equilibrium

[1]Shodan is a search engine that allows users to search various ICT devices connected to the Internet.

solution to the relaxed maximisation problem and verify numerically whether this solution is indeed the equilibrium solution to the original game. The solution to the relaxed maximisation problem and numerical verification can be found in Section 3. Convex relaxation procedure is described in great detail in Appendix C.

The model produces three main results. First, contrary to the literature, it demonstrates that the centrally protected star does not lead to the highest payoff for the defending side in most circumstances. Instead, it is an extreme formation, which yields the highest possible protection and the lowest possible efficiency among all connected graphs. Second, it reveals the new type of network that is often optimal in games with scarce defensive resources—a maxi-core network with a completely connected core and sparse periphery. A *maxi-core* network is an intermediate option that optimally balances between a star network's security and the efficiency of a complete network. This network allows the Defender to enjoy the efficiency of a completely connected core while extracting the maximum possible value from a periphery without an increase in expected loss from an attack. Third, it shows that the density of the optimal network decreases in the cost of a single link. Additionally, we have developed a weighted model modification, which allows us to account for the possibility that the damage received by the Defender extends beyond the value of the compromised vertex (e.g. reputational losses) or constitute only a share of the value of a compromised vertex (e.g. partial or temporary loss of a node).

The paper is built as follows. This section provides an introduction and reviews related literature. Section 2 formalises the model and offers the analysis of the final encounter stage of the game. Section 3 analyses the overall equilibrium of the game. In Section 4 we present the weighted versions of the model. The discussion of limitations and future research is presented in Section 5. For those interested in computing the equilibria, related proofs, and supplementary materials, see Appendices A-E.

1.1 Related literature

This study contributes to the literature on network design and defence, which emerged on two pillars of theoretical economics: contest theory and network theory.

Contest theory is concerned with strategic resource allocation in conflict situations; see Konrad (2009) for a comprehensive survey. A considerable share of literature addresses optimal resource allocation over multiple battlefields. Roberson (2006), Hart (2008), Washburn (2013), and Kovenock and Roberson (2008, 2012, 2021) researched the family of Colonel Blotto and General Lotto games, in which players simultaneously distribute limited resources over several battlefields. Each player then obtains the payoff equal to the sum of the valuations of the battlefields she won. Battlefields' valuations in Colonel Blotto games are assumed to be exogenous. On the contrary, our model allows a defender to choose network structure and, consequently, manipulate the values of battlefields (named *vertices* in our setting). An additional strategic element allows us to study the optimal interconnected structures of the battlefields and how the choice of a network structure interacts with contest incentives.

Network theory focuses on strategic network formation, optimal structures, and performance of social and economic networks in diverse circumstances; Jackson (2010), and Goyal (2007) provided perhaps the most comprehensive surveys on the matter. While the network theory takes its origins in the early 17 century, studies on network design and defence in the presence of an intelligent adversary appeared only recently (Naumowicz, 2014).

Arguably, the most famous series of papers on the topic was started by Goyal and Vigier (2010, 2014). In their original framework, a designer and an adversary compete over several stages in a network formation and defence game. In the first stage, the designer chooses a network formation and distributes defensive resources. The adversary then observes the network and the distribution of defensive resources, allocates contagious attack resources on nodes, and chooses how successful resources should navigate the network. The network in their framework represented the interconnected system of computers, while the attack was modelled after a virus, which navigated through it. Similarly to our research, the authors then studied the trade-off between the connectivity and the increased vulnerability that the connectivity implies. Still, our study has two crucial differences.

Firstly, in the original papers, the authors assumed that all nodes were homogeneous and used a simple cardinality[2] function to evaluate the network. While useful for studying optimal defensive networks, the approach neglected the influence of network formation on overall graph efficiency, suggesting that a centrally-protected star is an optimal structure in most scenarios. On the contrary, the vertices in our framework are heterogeneous

[2]Cardinality refers to the number of vertices in the network.

and valued according to their degree centrality—the number of immediate neighbours. The alternative setting confirms that a centrally-protected star is indeed the most secure structure but also demonstrates that it is the least efficient connected network. It can only be optimal for a defending side whenever the cost of a single connection is sufficiently high. Secondly, the original model assumed perfect information, meaning that the attacker knew both the network structure and the defensive resource allocation. On the contrary, we employ a simultaneous game approach in the encounter stage that represents the players' uncertainty about the actions of their corresponding opponents, which is more in line with real-world cybersecurity incidents (Anderson, 2001).

Cerdeiro et al. (2015), Goyal et al. (2016), Dziubiński and Goyal (2017) utilised a very similar setting to the original Goyal and Vigier papers. The authors explored the defence of the edges and decentralised approaches to network protection, employing the same cardinality-based value function as described above. Cerdeiro et al. (2015) and Goyal et al. (2016) assumed every node to be a player, allowing them to unilaterally create connections and choose their own "immunisation" plan. The strategic capabilities of the adversary in the paper by Goyal et al. (2016), however, were limited, enabling him only to choose the type of nodes over which he mixes uniformly at random. Dziubiński and Goyal (2017) studied conflict intensity, denoted as the minimum sum of costs spent by the defender and the attacker. Even though the network was treated as given, the author discovered a new efficient star-like formation—the windmill graph. The decentralised defence is out of the scope of this study, but we expand on the ideas of those papers by introducing more sophisticated network value functions and uncertainty about the opponents' actions.

Gueye et al. (2011) proposed an alternative approach to the problem. In their simultaneous network topology game, a defender chooses a spanning tree of some connected undirected graph while an attacker chooses an edge to attack to disrupt the communication between vertices. The authors assumed that any spanning tree has the same cost (normalised at one) and that the attacker must guess which edges are present in the chosen tree. If the attacker does not guess the correct edge, the game stops, and the network wins. The authors concluded that the attacker mixes uniformly at random over the set of potential critical edges (defined as edges that yield the largest payoffs for the attacker). Similarly to Goyal and Vigier (2014), networks in this setting are differentiated only by defensive capabilities, neglecting the efficiency that the defending side might extract from the structure itself. Moreover, any connected network structure with the same cardinality has the same price for the network designer. Our framework utilises a more sophisticated network value function and assumes that connections are costly. This enables us to study the influence of an edge cost on the optimal choice of the network.

Acemoglu et al. (2016) studied the model of security investments in a network of interconnected agents. The network structure was treated as given, and the main focus was shifted towards the possibility of cascading failures following the exogenous or endogenous attacks, thus not covering our main findings.

The sequential game presented by Hoyer and Jaegher (2016) studied the optimal network structure against two modes of external attack: a link-deletion attack and a vertex-deletion attack. Contrary to the general literature, they found that star formation is the worst choice against a vertex-deletion attack. Their result is caused by the network designer's inability to protect (or immunise) vertices. The authors, however, discovered that the cost of a link is one of the leading influencers on optimal network formation, which echoes our results.

Studies of optimal network design and defence are not limited to economics literature. For instance, recent papers by Makridis (2021) and Wang et al. (2021) utilised a reinforcement learning approach to solve a very similar design and defence problem. However, due to the limitations of the methodology, the action sets of both the defender and the attacker were limited in both studies. In the former paper, a defender had an initial network and was only allowed to add links, while the attacker could attack only the nodes of the largest value. In the latter paper, the authors used a game-theoretic specification to set up a reward function and recognised the existence of mixed-strategy equilibrium, but did not provide any insights about the optimal network formation.

To conclude, our study contributes to the literature along three dimensions: (1) we introduce a more sophisticated network value function, which recognises the influence of connectivity on overall graph value and allows us to study the tension between security and efficiency; (2) we study the simultaneous version of the game that models the uncertainty of both defending and attacking sides, which is more in line with real-world cybersecurity events; (3) the costly links assumption allows us to study the optimal network formation under different cost regimes.

2 The model of network design and defence

2.1 Model setup

Two players, the Defender (She) and the Attacker (He), compete in the network defence game over two stages: the initial **wiring stage** and the following **encounter stage**.

In the **wiring stage** the Defender is endowed with a fixed quantity $n > 1$ of vertices (or nodes), which she arranges in some connected graph[3] G by creating costly edges (or links) between them. For instance, the vertices can represent the employees of some organisation, while edges are organisational links between them. Similarly, we could think of vertices as computers belonging to the organisational network and edges as literal connections between ICT devices. In our baseline model, we assume that every vertex in the newly arranged graph is valued according to its degree of centrality.

Definition 2.1 (Degree Centrality)**.** *Degree centrality is defined as the number of links incident upon a vertex (i.e. the number of ties that a vertex has).*

The usage of degree centrality represents the idea that more connected vertices are more valuable for organisational integrity and allows us to study the trade-off between the security and efficiency extracted from a network structure. The partially ordered set of vertices' values is represented by vector $\vec{v} = (v_1, \ldots, v_n)$. The network's overall value is assumed to be the sum of the value of all vertices minus the cost of all edges. As each additional edge increases the sum of values of all vertices by two, each network must have exactly $\frac{1}{2}\sum_{i=1}^{n} v_i$ edges. Multiplying the number of edges by the cost of a single link, c, yields the overall cost of the edges.

Assumption 2.1. *The overall value of the networks G is the difference between the sum of values of all vertices and the cost of all edges:*

$$V(\vec{v}) = \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i, \tag{2.1}$$

where c is the cost of a single edge.

At the beginning of the **encounter stage**, the Attacker observes the network structure G created by the Defender in the wiring stage. The Attacker can obtain information on

[3] A connected graph is a graph in which there is a path from any vertex to any other vertex.

the enterprise's organisational structure via social media (e.g. LinkedIn) or perform port scanning to determine the organisation of IP addresses, hosts, ports and their networked structure. The Attacker and the Defender then engage in a simultaneous zero-sum game: the Attacker attempts to compromise one of the vertices of his choosing, while the Defender attempts to prevent that by protecting $\delta \geq 1$ vertices. For example, the Attacker chooses an entry point to exploit in the network, and the Defender chooses an appropriate immunisation plan (e.g. device patching policies). Alternatively, the Attacker might choose an individual employee for a spear-phishing attack, while the Defender decides which employees to prioritise for cybersecurity awareness training.

The attack is successful if the Attacker targets an unprotected vertex, receiving a payoff equal to this vertex's value, while the Defender obtains the overall value of the network minus the value of the compromised node[4]. If the Attacker targets a protected node, the attack fails, and the game finishes. In this case, the Attacker obtains nothing, and the Defender loses nothing and obtains the value of the network in full. The payoff of the Attacker is derived in the following subsection, while the Defender's expected payoff at the beginning of the game is formally stated at the beginning of Section 3.

The game utilises Subgame Perfect Nash Equilibrium (SPNE) as a solution concept and assumes that each player rationalises against its opponent's optimal choices. The game is solved via backward induction, beginning at the encounter stage.

2.2 Analysis: encounter stage

In the encounter stage **the Defender** already possesses a set of nodes arranged in a connected network G and $\delta \geq 1$ defensive resources, which she uses to protect δ nodes. If some node i is defended, then any attack on this node is unsuccessful. Vertices' values are written as a finite set $\vec{v} = (v_1, \ldots, v_n)$ such that $v_1 \geq v_2 \geq \cdots \geq v_n$ without loss of generality.

The Attacker insidiously attempts to compromise one of the vertices from the network G. The strategic form of the zero-sum game when $\delta = 1$ is illustrated by Table 1.

Observe from Table 1 that the subgame played in the encounter stage does not have

[4]In Section 4 we consider the case where the damage is allowed to be larger or smaller than the value of the compromised node.

		Defender				
		1	2	3	$\cdots$	n
Attacker	1	0	v_1	v_1	$\cdots$	v_1
	2	v_2	0	v_2	$\cdots$	v_2
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	n	v_n	v_n	v_n	$\cdots$	0

Table 1: Strategic form game on the network with n vertices and $\delta = 1$, where the number in each cell gives the Attacker's payoff (or the loss of the Defender).

an equilibrium with pure strategies. For instance, suppose vertex i is attacked with probability 1. The best response of the Defender is to defend i with probability 1, leaving the Attacker with a payoff of zero. The Attacker then has a profitable deviation to attack a different node. A similar situation occurs when the Defender has multiple defensive resources, $\delta > 1$. In this case, if the Attacker attacks vertex i with probability 1, the Defender's best response is to allocate one of the defensive resources to node i with probability 1. Again, this leaves the Attacker with zero payoff and a profitable deviation to attack a different unprotected node. It follows that the equilibrium strategies of the Defender and the Attacker must be mixed.

Claim 2.1. *The encounter stage subgame does not have an equilibrium with pure strategies.*

Additionally, we assume the scarcity of defensive resources. Observe that if $\delta \geq n$, the Defender's best response is to protect every vertex, even if it is not attacked with a positive probability. In this case, the Attacker is indifferent to any strategy, as he knows that the attack will fail regardless of his actions. From now on, we focus on the more interesting case with $\delta < n$.

Assumption 2.2. *The Defender has $\delta < n$ defensive resources.*

It is now essential to establish the set of vertices attacked with positive probability, referred to later in the article as *the Attacker's support.*

Definition 2.2 (Attacker's Support)**.** *The Attacker's support ($supp_A$) is the set of nodes attacked with positive probability.*

Similarly, we refer to the set of nodes defended with positive probability as the Defender's support, supp_D.

First, observe that in the mixed equilibrium $|\,\text{supp}_A\,| = k > \delta$. If it is not the case (and $k \leq \delta$), the Defender can protect all k vertices included in the Attacker's support leaving the latter with zero payoff.

Claim 2.2. *The Attacker always attacks $k > \delta$ vertices with positive probability:*

$$|\text{supp}_A| > \delta.$$

Second, observe that if some vertex is not attacked with positive probability, the Defender never protects it. There is no incentive to protect the vertex, which will certainly not be attacked. Thus, the Attacker's support is always larger or equal to the Defender's.

Claim 2.3. *The Defender protects a vertex with positive probability only if it is attacked with positive probability:*

$$\text{supp}_D \subseteq \text{supp}_A.$$

Now we demonstrate that in any equilibrium, the Attacker mixes over $k > 1$ highest value vertices of the network G. If some vertex with index $m > 1$ in $\vec{v}$ is attacked with positive probability, then every vertex of a higher value is attacked with positive probability as well. The finding is summarised below.

Claim 2.4. *In any equilibrium if node m is attacked with positive probability, so is any node l such that $v_l > v_m$.*

Proof of Claim 2.4. See Appendix A. ■

An immediate corollary of Claim 2.4 is that in any organisational structure, the most valuable vertices should always be defended with positive probability in equilibrium. Ensuring the protection of the most valuable employees (e.g. board members, research personnel) and the most connected ICT devices (e.g. servers and routers) can minimise the expected loss from an attack.

Consider now some equilibrium candidate with the top k highest value vertices attacked with positive probability. Suppose that node j is defended with probability q_j and not defended with probability $x_j = 1 - q_j$. Then, the Attacker is indifferent between attacking vertex i and vertex j if:

$$v_i x_i = v_j x_j.$$

Observe that $\sum_{j=1}^{k} x_j = \sum_{j=1}^{k} \left(1 - q_j\right)$ and $\sum_{j=1}^{k} q_j = \delta$. It follows that $\sum_{j=1}^{k} x_j = k - \delta$. Using $x_i = \frac{v_k}{v_i} x_k$ we get:

$$\left(\frac{v_k}{v_1} + \frac{v_k}{v_2} + \cdots + \frac{v_k}{v_k}\right) x_k = k - \delta. \tag{2.2}$$

Rearranging equation (2.2) yields the probability of node k to remain undefended:

$$x_k = \frac{(k - \delta) \prod_{j=1}^{k} v_j}{v_k \sum_{j=1}^{k} \prod_{i \neq j}^{k} v_i}. \tag{2.3}$$

Since x_k is a probability, we must have $x_k \leq 1$, which yields the following necessary condition for k to be in the equilibrium Attacker's support:

$$\frac{(k - \delta) \prod_{j=1}^{k} v_j}{v_k \sum_{j=1}^{k} \prod_{i \neq j}^{k} v_i} \leq 1. \tag{2.4}$$

It is now necessary to check for the monotonicity in k of condition (2.4), such that the condition for k vertices implies the condition for top $k-1$ vertices. This would guarantee that any pure strategy in the Attacker's support yields him an equal expected payoff and, consequently, that there are no nodes $z < k$ excluded from the Attacker's support and that there is no profitable deviation to smaller support for the Attacker.

Claim 2.5. *The condition* (2.4) *for the top k vertices implies the condition for the top $k - 1$ vertices.*

Proof of Claim 2.5. See Appendix A. ∎

Moreover, condition (2.4) is also sufficient for there to exist an equilibrium with k nodes included in the support, as it also guarantees that any pure strategy in the Attacker's support yields him an equal or higher payoff than the value of any node excluded from the Attacker's support.

The encounter stage of the game follows the "no soft-spots" principle for Colonel Blotto games with multiple targets described by Dresher (1961). The Defender mixes over high-value nodes such that each becomes equally attractive (in expectation) to the Attacker, implying that there are no "soft-spots" among all the protected nodes. Furthermore, the nodes that are never protected with positive probability are of lesser value to the Attacker than the expected gain from an attack on any of the protected nodes. Thus, the Attacker mixes only over nodes protected with positive probability or nodes with a value equal to the expected gain from the attack on any of the protected nodes.

Now we denote the ordered set of values of nodes that are included in the Attacker's support as $\vec{v_k}$. Finally, we can write down the expected loss function for the Defender $D(\vec{v_k})$, or the expected gain function for the Attacker $A(\vec{v_k})$ on a given graph G knowing the Attacker mixes over top k highest value nodes:

$$D(\vec{v_k}) = -A(\vec{v_k}) = -\frac{(k-\delta)\prod_{j=1}^{k} v_j}{\sum_{j=1}^{k}\prod_{i\neq j} v_i} = -x_k v_k. \tag{2.5}$$

Observe that if condition (2.4) holds strictly for some node k and does not hold for node $k+1$, the game has a unique equilibrium in which the Attacker and the Defender have identical support $\text{supp}_D = \text{supp}_A$. The equilibrium is unique, as if the Attacker chooses smaller support mixing over $k-z$ nodes, where $z \in \mathbb{Z}$ and $z \in [1, k-\delta)$, then by Lemma 2.2, the Defender's optimal strategy is to protect only $k-z$ nodes with positive probability. In this case, if both players mix over the $k-z$ nodes such that the indifference principle is satisfied for both of them, the Attacker receives a strictly lower expected payoff than from mixing over the top k nodes.

Claim 2.6. *If condition* (2.4) *holds strictly for some node k, while it does not hold for node $k+1$, then the encounter stage has a unique equilibrium in which the Attacker and the Defender mix over the top k nodes* $|\,\text{supp}_D\,| = |\,\text{supp}_A\,| = k$ *and* $\text{supp}_D = \text{supp}_A$.

Proof of Claim 2.6. See Appendix A. ■

However, the equilibrium support might not be unique. If condition (2.4) holds strictly for some node y and with equality for nodes indexed $\nu \in (y, y+z]$, where $z \in \mathbb{Z}$ and $z \in [1, n-y]$, then the Attacker has multiple equilibrium supports, all of which are payoff-equivalent. This echoes a known result about the payoff-equivalence in zero-sum games by von Neumann (1928). We summarise this observation in Claim 2.7.

Claim 2.7. *If condition* (2.4) *holds strictly for some node y and with equality for nodes indexed $\nu \in (y, y+z]$ then the Attacker has multiple equilibrium supports $|\,\text{supp}_A\,| \in [y, y+z]$, all of which are payoff-equivalent.*

Proof of Claim 2.7. See Appendix A. ■

Therefore, the size of the Attacker's equilibrium support can be written down as follows:

$$|supp_A| \in [k, g],$$

where

$$k = \max\left\{ z : \frac{(z-\delta)\prod_{j=1}^{z} v_j}{v_z \sum_{j=1}^{z} \prod_{i\neq j} v_i} < 1 \right\},$$

and

$$g = \max\left\{ z : \frac{(z-\delta)\prod_{j=1}^{z} v_j}{v_z \sum_{j=1}^{z} \prod_{i\neq j} v_i} \leq 1 \right\}.$$

Now observe that the expected gain from an attack on top k nodes (2.5) is strictly decreasing in δ. It immediately follows that under Assumption 2.2, the Defender utilises all the available defensive resources in any equilibrium.

Claim 2.8. *The Defender utilises all the available defensive resources in any equilibrium.*

Observe that the Defender cannot influence k in the encounter stage. However, as the size of the Attacker's support is a function of vertices' values, it follows that the Defender can manipulate it in the wiring stage when choosing the network formation.

We now show that the Attacker's expected gain in the encounter stage is minimised when the game is played on a star network[5]. At the same time, it attains its maximum when the game is played on a complete network[6].

Observe first that the Defender's expected payoff is convex over $\vec{v_k}$, or, alternatively, that the Attacker's expected payoff is concave over $\vec{v_k}$, where index k corresponds to the number of nodes included in the Attacker's support.

Lemma 2.1. *For any strategy of the Attacker satisfying Claims 2.5–2.7 and condition* (2.4)*, the expected loss of the Defender/payoff of the Attacker in the encounter stage is convex/concave over* $\vec{v}_k = \{(v_1, \ldots, v_k) \in \mathbb{R}_+^k \mid 1 \leq v_i \leq n-1\}$.

Proof of Lemma 2.1. See Appendix A. ■

We now establish the results about which networks yield the highest and the lowest expected gain for the Attacker. We demonstrate that the Attacker obtains the highest expected gain if the game is played on a complete network, while he obtains the lowest possible gain if the game is played on a star network.

[5] A star network has a degree distribution in which one node is completely connected, and the rest of the nodes have one connection each, $G_* = (n-1, 1\cdots, 1)$.

[6] A complete network has a degree distribution in which each node is completely connected, $G_C = (n-1, \cdots, n-1)$.

Proposition 2.1. *The Attacker's expected gain attains its minimum when the game is played on a star network, while it attains its maximum when the game is played on a complete network.*

Proof of Proposition 2.1. See Appendix A. ■

The results of Proposition 2.1 should be intuitive. For a fixed number of nodes n, star and path[7] networks have the smallest possible number of links, limiting the expected gain of the Attacker. However, unlike in a path network, half of the value of a star is concentrated in a single core node. Thus, the Defender is always better off choosing a star network than a path network since protecting the core node with high probability limits her expected loss while keeping the same overall network value[8]. On the contrary, a complete network has the largest possible quantity of links, meaning that every single node attains the highest possible value. Given that defensive resources are scarce, it provides the Attacker with an opportunity to compromise a completely connected node with high probability, yielding the maximum possible gain. Since the encounter stage is a zero-sum game, it also implies that Defender's expected loss is maximised when the encounter stage is played on a complete network and minimised when it is played on a star network.

These findings echo the results obtained by Goyal and Vigier (2014), who also recognised that a centrally protected star leads to the highest possible payoff for the defending side in a strategic contest versus the intelligent Attacker. The star network is, however, not necessarily an optimal choice for the Defender if we also account for the overall network value—the research proceeds with the analysis of the initial **wiring stage** of the game.

3 Equilibrium

In the **wiring stage** the Defender chooses a network formation. She optimises the graph structure by considering the graph's overall value and the equilibrium expected payoff in the encounter stage.

[7]The path graph (or network) has two nodes of degree 1, and all other nodes of degree 2.

[8]In fact, the expected loss of the Defender in the encounter stage played on a star network never exceeds 1 in the equilibrium.

Consider first the expected value of the Defender at the beginning of the game if she chooses some network G.

$$U(\vec{v}) = V(\vec{v}) + D(\vec{v_k}), \tag{3.1}$$

where:

$V(\vec{v})$ is the overall value of the network, $V(\vec{v}) = \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i$ by Assumption 2.1;

$D(\vec{v_k})$ is the expected loss of the Defender from an attack on k highest value nodes in the encounter stage.

We restrict the cost of an edge to be $c \leq 2$. If $c > 2$, the Defender's payoff is always negative, and she always chooses the network with the least possible number of edges and best defensive capabilities—a star network. The following assumption rules out this possibility.

Assumption 3.1. *The cost of a single link is restricted to $c \leq 2$.*

The Defender's goal in the wiring stage is to maximise the expected value of the game (3.1). In order to do so, she must choose some degree sequence[9] which maximises her payoff. We call this sequence *a maximising degree sequence.* Note that two networks with identical degree sequences yield the same payoff for the Defender.

Definition 3.1 (Maximising Degree Sequence)**.** *A maximising degree sequence is a degree sequence which maximises the Defender's expected value function.*

It immediately follows from Lemma 2.1 that the Defender's expected utility at the beginning of the game (3.1) is convex over the set of node values' $\vec{v}$ for a given size of the Attacker's support k as the sum of a linear and convex functions.

Claim 3.1. *The Defender's expected value function at the beginning of the game, $U(\vec{v})$, is convex over $\vec{v} = \{(v_1, v_2, ...v_n) \in \mathbb{R}_+^n \mid 1 \leq v_i \leq n-1\}$ for a given size of the Attacker's support k.*

Observe that the objective function of the Defender (3.1) is discontinuous as the expected loss from an attack has a different form depending on how many nodes are

[9] A degree sequence of a graph is some monotonic nonincreasing sequence of positive integer numbers, which represents all of its vertex degrees.

included in the Attacker's support. To proceed, we exploit the fact that any network with a given degree distribution can be attributed to one or several *classes* with all networks in a given class sharing the same size of the Attacker's support, $|\operatorname{supp}_A|$. Given that the minimum Attacker's support must be $k > \delta$, there exist $n-\delta$ classes of networks, each of which has the same form of the expected loss from an attack. If the encounter stage on some network has multiple equilibrium supports, then this network belongs to several classes with respect to the Attacker's support. However, by Claim 2.7, regardless of which equilibrium support the Attacker chooses (and to which class the corresponding network is attributed), both players always obtain equivalent payoffs.

Thus, we divide the Defender's problem in the wiring stage into two sub-problems:

1. The full support case ($k = n$) in which all nodes pass the Attacker's support condition (2.4), meaning that all nodes will be attacked with positive probability in equilibrium.

2. The partial support case ($k \leq n$) in which k nodes pass the Attacker's support condition (2.4) and the other $n - k$ nodes do not. In this case, top k nodes will be attacked with positive probability in equilibrium, and other $n - k$ nodes will not. The partial support case yields $n - \delta - 1$ similar maximisation problems, which we consider simultaneously.

We can now state the Defender's maximisation problem in a standard form for a given class of networks k. Finding the solutions to each of $n - \delta$ maximisation problems and then finding among them the one that yields the maximum expected payoff for the Defender yields the equilibrium solution for the game.

$$\max_{v_i} \quad \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i - \frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}} \tag{3.2}$$

$$\text{s.t.} \quad v_i \in \mathbb{Z} \qquad \forall i \in [1, n], \tag{3.3}$$

$$1 \leq v_i \leq n - 1 \qquad \forall i \in [1, n], \tag{3.4}$$

$$\sum_{i=1}^{y} v_i \leq y(y-1) + \sum_{i=y+1}^{n} \min(v_i, y) \qquad \forall y \in [1, n], \tag{3.5}$$

$$\frac{k - \delta - 1}{\sum_{j \in K \setminus \{i\}} \frac{1}{v_j}} - v_i \leq 0 \qquad \forall i \in [1, k], \tag{3.6}$$

$$-\frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}} + v_j \leq 0 \qquad \forall j \in (k, n], \tag{3.7}$$

where:

K is the set of all vertices in the Attacker's support endogenously defined in the encounter stage;

(3.2) is the objective function of the Defender;

(3.3) is the integer constraint on the values of the nodes;

(3.4) is the set of degree constraints[10];

(3.5) is the set of Erdos-Gallai's sufficient and necessary conditions for the graphicality of a sequence (Tripathi & Vijay, 2003);

(3.6) is the lower boundary for the degrees of vertices inside the Attacker's support derived from condition (2.4);

(3.7) is the upper boundary for the degrees of vertices outside the Attacker's support derived from condition (2.4).

We denote the set of all degree sequences that satisfy constraints (3.3)–(3.7) as Γ.

Erdos-Gallai conditions (3.5) ensure that the degree of some vertex i in degree sequence G does not exceed the maximum possible value given the degrees of all other vertices. If the conditions are satisfied and the sum of all degrees in a sequence is even, the sequence is guaranteed to be graphic. A sequence is said to be graphic if it is possible

[10] Note that any vertex in a connected network must have at least one edge and at most $n - 1$ edges.

to construct a graph having the sequence as its degree sequence. For more information about Erdos-Gallai conditions, refer to Appendix B.

Constraints (3.6) and (3.7) are imposed by the Attacker's support condition (2.4) for a given class of networks with k top nodes attacked with positive probability. From (2.4) we have that the vertices included in the Attacker's support must be:

$$\frac{(k-\delta)}{\sum_{j=1}^{k}\frac{1}{v_i}} \leq v_i,$$

or rearranged and simplified:

$$\frac{k-\delta-1}{\sum_{j\in K\setminus\{i\}}\frac{1}{v_j}} \leq v_i. \tag{3.8}$$

Then the vertices outside the Attacker's support must be:

$$v_j \leq \frac{(k-\delta)}{\sum_{i=1}^{k}\frac{1}{v_i}}. \tag{3.9}$$

It follows from Lemma 2.1 that conditions (3.7) are convex in standard form and, therefore, must span a convex set. However, by the same lemma, conditions (3.6) are defined by a sum of the concave and linear functions, which might result in non-convexity of Γ even assuming the continuity of nodes' values. Therefore, the maximisation problem (3.2) is an integer maximisation problem defined over a non-convex set. Problems of this kind can be attributed to the family of Mixed-Integer Non-linear Problems (MINLP), which, in general, are known to be non-deterministic polynomial-time (NP) hard problems (Sahinidis, 2019). Therefore, to approach the maximisation problem (3.2), integrability and non-convex constraints must be relaxed.

Roadmap for the analysis We perform the equilibrium analysis in three steps. First, in Subsection 3.1, we establish general analytical results about the equilibrium of the game. Second, in Subsection 3.2, we analyse the base case of the game with $\delta = 1$ utilising the convex relaxation technique to obtain candidate solutions for the original maximisation problem (3.2). Third, in Subsection 3.3, we check the feasibility of relaxed problem solutions for the original maximisation problem. Finally, in Subsection 3.4, we perform a numerical analysis of the Defender's problem to verify that the solutions to the relaxed problem are indeed the solutions to the original problem.

3.1 Limiting properties of equilibrium

First, observe that the condition for some network G to be optimal is a function of the cost of a single link c. This is because the Defender's expected utility function at the beginning of the game (3.1) is linear in c. Assuming that we can write down the set of all possible connected networks on n nodes, we then can compare them for a given value c and choose the particular network which yields the maximum utility for the Defender for this c. Performing this exercise for some arbitrary large network is not tractable due to the sheer quantity of possible connected networks. However, we demonstrate that two network structures always occur in equilibrium for extreme values of c regardless of the number of vertices and defensive resources in the Defender's possession: complete and star networks. A complete network emerges as a maximising degree sequence if the cost of a single edge is sufficiently small, $c < c_l$, while a star is an optimal choice if the cost of a single edge is sufficiently high, $c > c_u$. We denote c_u and c_l as higher and lower cost boundaries. This result is summarised in Proposition 3.1.

Definition 3.2. *A cost boundary is a threshold which demarcates two different maximising degree sequences.*

Proposition 3.1. *A complete network is optimal for the Defender if the cost of a single edge is sufficiently small, $c < c_l$. A star network is optimal whenever the cost of an edge is sufficiently high, $c > c_u$.*

Proof of Proposition 3.1. See Appendix A. ■

Figure 2 illustrates the principles behind Proposition 3.1. We take $n = 5$ and $\delta = 2$ and draw 21 lines that correspond to the Defender's utility functions (w.r.t. c) calculated for all of the possible connected networks. For the relatively small quantity of vertices and defensive resources, it is fairly easy to determine all of the potential maximising degree sequences and then verify which ones can serve as maximisers. Observe that in the illustrated case, there are four maximising degree sequences and three corresponding cost boundaries. The quantity of the intermediate boundaries might be significantly increased with an increase in n and δ, but the lower-cost boundary (c_l) and the upper boundary (c_u) will always exist.

In Figure 3 we demonstrate the graphs which correspond to maximising degree sequences for $n = 5$ and $\delta = 2$.

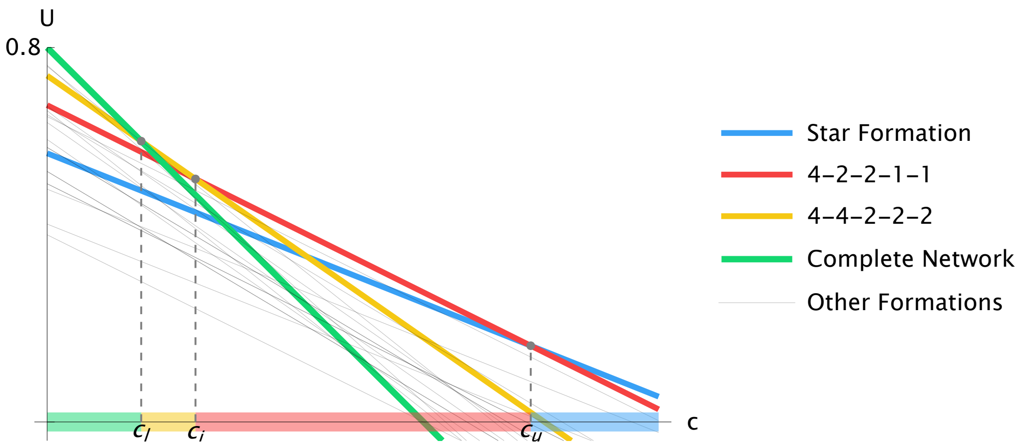
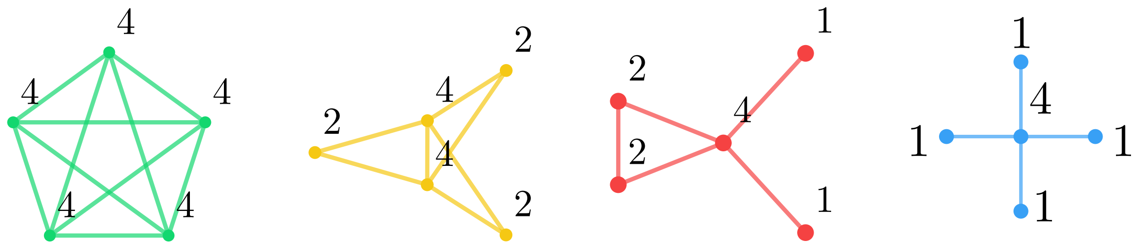


Figure 2: The Defender's expected value from a game on five vertices ($n = 5$) with two defensive resources ($\delta = 2$) as a function of a single link cost, c, built for every feasible connected network structure. Highlighted lines are spawned by the networks, which appear as equilibrium solution for some given c. Boundaries on the figure are: $c_l = 1.7$ is the lower cost boundary, $c_u = 1.78$ is the upper boundary, and $c_i = 1.71$ is an intermediate boundary.

Thus, Proposition 3.1 implies that whenever the cost of connection is sufficiently low, the Defender prioritises network connectivity over the potential security concerns—choosing a complete network. This is because low connection costs allow the Defender to extract large benefits from every additional link, which vastly outweigh potential losses from an attack. On the contrary, if the connection cost is sufficiently high, the Defender's benefits from additional links are limited and can be easily washed out by an external attack. In this case, the Defender prioritises network security over its connectivity. By choosing a star network and protecting a central node with high probability, the Defender can significantly reduce the potential losses from an attack while still extracting some value from limited connectivity.

Additionally, observe from Figures 2 and 3 that the higher cost of a single edge is associated with the optimal network structures of a lower density[11], while the lower cost of an edge is associated with higher density structures. In Proposition 3.2, we demonstrate that this observation is true in the general case.

[11]Network density is the portion of the *potential edges* in a graph that are actual edges. *Potential edge* is the edge that might exist between two nodes—regardless of whether it actually exists or not. Any network on n vertices has $\frac{n(n-1)}{2}$ potential edges. A network with n vertices and e edges has a density of $\frac{2e}{n(n-1)}$.



(a) $c < c_l$ (b) $c_l < c < c_i$ (c) $c_i < c < c_u$ (d) $c_u < c$

Figure 3: All optimal networks for a game with five vertices and two defensive resources. The colors of networks correspond to the colors of lines demonstrated in Figure 2.

Proposition 3.2. *The density of equilibrium degree sequence is a monotonically decreasing function of the cost of a single edge, c.*

Proof of Proposition 3.2. See Appendix A. ■

Propositions 3.1 and 3.2 suggest that the Defender evaluates the added benefits of connectivity against an increase in potential damage from an attack that the connectivity brings. Moreover, the density of an optimal network is a decreasing function of the cost of a connection, with the star and complete networks being extremes of this function. Thus, an organisation can optimise its network structure by balancing additional connectivity benefits against security concerns. Those results echo the findings of Slikker and van den Nouweland (2000), who have demonstrated that, in the general case, the higher cost of a connection in communication networks is associated with structures of lower density.

We now demonstrate that when the Defender possesses some arbitrary finite number of defensive resources, $\delta \geq 1$ and a large number of vertices $n \to \infty$, the Attacker can only compromise some infinitesimal share of the network. Therefore, if the Defender possesses a large number of nodes, $n \to \infty$, network effects outweigh the potential loss from an expected attack, and it is always optimal for her to choose a complete network. We summarise this claim below.

Proposition 3.3. *If the Defender possesses $n \to \infty$ vertices and some finite number of defensive resources $\delta \geq 1$, the game has a unique equilibrium solution in which the Defender chooses a complete network.*

Proof of Proposition 3.3. See Appendix A. ■

Thus, if some organisation has a very large number of nodes (e.g. a large international corporation) and has limited defensive resources, an attack is successful with a probability which tends to 1. However, the benefits extracted from the connectivity of a large network are disproportionally larger than any potential damage the organisation might encounter. In this case, a connected network yields the largest possible payoff for the organisation. However, note that this result might not hold if the upper boundary of a loss from an attack exceeds the value of the largest node. In Section 4, we offer a modified version of the model, which can account for this possibility.

To conclude, we obtained three general properties of the equilibrium: (1) if the Defender has some finite number of nodes, a star network is always optimal for a sufficiently high cost of a single edge, while a complete network is always optimal for a sufficiently small cost of a single edge; (2) the density of maximising degree sequence monotonically decreases with respect to a cost of a single edge and (3) a complete network always yields the maximum payoff for the Defender if $n \to \infty$.

To analyse the equilibrium further we now consider the base case of the game in which $\delta = 1$.

3.2 Relaxed problem solution, $\delta = 1$

Since the original maximisation problem of the Defender in the wiring stage (3.2) is defined over a non-convex set, it cannot be solved utilising standard convex optimisation methods. Thus, to proceed, we utilise a convex relaxation method widely used to find approximate solutions to MINLP problems (Pulleyblank, 1989; Tuy & Van Thuong, 1988).

Convex relaxation is a modelling technique in which some of the constraints of the original problem are relaxed, extending the objective function to the larger convex space. We perform convex relaxation in two steps: (1) we lift the integrality constraints (3.3) and Erdos-Gallai constraints (3.5) which are inherently integer, and (2) we linearise non-convex constraints (3.6).

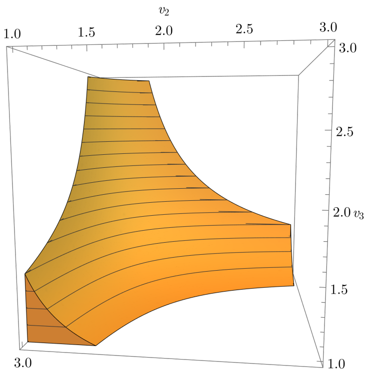
1. Integer relaxation Lifting integrality constraints (3.3) and Erdos-Gallai graphicality constraints results in the following relaxed maximisation problem:

$$
\begin{aligned}
\max_{v_i} \quad & \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i - \frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}} && && (3.10) \\
\text{s.t.} \quad & 1 \leq v_i \leq n-1 && \forall i \in [1,n], && (3.11) \\
& \frac{k-2}{\sum_{j \in K \setminus \{i\}} \frac{1}{v_j}} - v_i \leq 0 && \forall i \in [1,k], && (3.12) \\
& -\frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}} + v_j \leq 0 && \forall j \in (k,n]. && (3.13)
\end{aligned}
$$

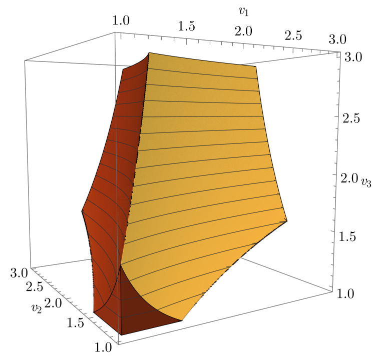
We denote the set over which the maximisation problem (3.10) is defined as Υ, $\Gamma \subseteq \Upsilon$.

2. Convex hull relaxation Observe that set Υ might still be non-convex due to the presence of non-convex Attacker's support conditions (3.12). In order to "convexify" the problem and preserve maximum information about the original set, we construct a convex hull of the set over which optimisation problem (3.10) is defined. Convex hull of Υ is defined as the smallest convex compact set (Υ) such that $\Upsilon \subseteq (\Upsilon)$. We then maximise the objective function (3.10) over the convex hull of set Υ.

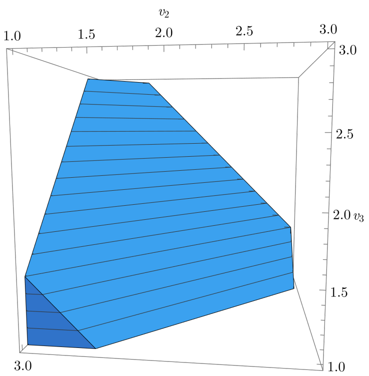
Since non-convex regions Υ are defined by smooth concave functions, the convex hull can be constructed by linearising constraints (3.12) (Boyd & Vandenberghe, 2004). Thus, in order to convexify Υ, we perform the following procedure: (1) we find all extreme points of non-convex regions of set Υ, (2) we find corresponding hyperplane equations spanned by the extreme points of non-convex regions of Υ, and (3) we replace non-convex constraints with linear constraints defined by hyperplane equations. The procedure is described in great detail in Appendix C. Figure 4 illustrates an example where $n = 4$ and the Attacker has full support.



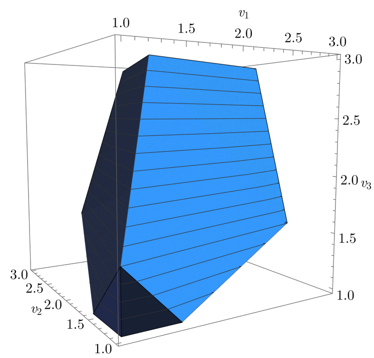
(a) Υ^4, $v_4 = 1$



(b) Υ^4, $v_4 = 3$



(c) (Υ^4), $v_4 = 1$



(d) (Υ^4), $v_4 = 3$

Figure 4: The set of feasible values, Υ^4, for nodes 1, 2, and 3 along the edge v_4, (a) and (b); and the convex hull of set of feasible values, (Υ^4), for nodes 1, 2, and 3 along the edge v_4, (c) and (d). More details about this example can be found in Appendix C.2.

The convexification of the set of feasible values yields an optimisation problem which requires a maximisation of a convex function over a convex compact set. Therefore, convex relaxation allows us to utilise the Bauer maximum principle (Bauer, 1958).

Theorem 3.1 (Bauer maximum principle)**.** *Any function that is convex and continuous, and defined on a compact convex set, attains its maximum at some extreme point of that set.*

Since the objective function of the original maximisation problem is convex by Claim 3.1 and continuous under integer relaxation for a given class of networks, and (Υ) is convex and compact by definition, it must then be maximised at some extreme point of a convex hull. If any extreme point of (Υ) is feasible for the original problem (3.2), then this point is a maximising degree sequence for the original problem (Geoffrion, 1971).

However, relaxation techniques might lead to sub-optimal maximising degree sequences, as by relaxing integer and non-convex constraints, we might lose some of the solutions that appear on non-convex regions of the original set. Thus, to back up our analysis, we numerically verify that the solution to the relaxed problem is indeed the optimal (equilibrium) solution for networks of up to 20 nodes in Subsection 3.4. The discussion on the limitations of relaxation techniques can be found in Subsection 5.

Also, observe that the original sets for full and partial support cases have different sets of constraints: in the partial support case, constraints (3.13) are active, while in the full support case, they are not. This implies that sets of extreme points for partial and full support cases might differ. Thus, we consider relaxed problems for both cases separately. To distinguish the cases, we denote the sets of feasible values for the full and partial support cases as Υ^C and Υ^P, respectively.

Characterising extreme points for the full support case and every partial support case and then finding the extreme points which yield the maximum payoff for the Defender for a given c yields a set of all maximisers for the relaxed maximisation problem.

Full support We state the relaxed maximisation problem for the full support case as:

$$\max_{v_i} \quad \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i - \frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}} \tag{3.14}$$

$$\text{s.t.} \quad v_i \in (\Upsilon^C). \tag{3.15}$$

The set of constraints which define (Υ^C) can be found in Appendix C.2.

In Appendix C.1 we show that the set (Υ^P) has extreme points that fall into one of four families:

1. Star network family:

$$\left(n-1, \underbrace{1, \cdots, 1}_{\times(n-1)}\right); \tag{3.16}$$

2. Complete network family:

$$\left(\underbrace{n-1, \cdots, n-1}_{\times n}\right); \tag{3.17}$$

3. Intermediate family:

$$\left(\underbrace{n-1,\cdots,n-1}_{\times q},\underbrace{\frac{(n-1)(q-1)}{q},\cdots,\frac{(n-1)(q-1)}{q}}_{\times(n-q)}\right), \tag{3.18}$$

where $q \in [2, n-1]$;

4. Lower family:

$$\left(n-1,\frac{n-1}{n-2},\underbrace{1,\cdots,1}_{\times(n-2)}\right). \tag{3.19}$$

Partial support We first demonstrate that any node excluded from the Attacker's support must attain the maximum possible value that satisfies constraints (3.13). To demonstrate that, we restate the objective function of the original maximisation problem to reflect that, in the partial support case, the network has two types of nodes: (i) nodes that are always attacked with a positive probability denoted v^k and (ii) nodes which are not attacked with positive probability v^s:

$$\max_{v_i^k, v_j^s} \quad U^P = \sum_{i=1}^{k} v_i^k + \sum_{j=k+1}^{n} v_j^s - \frac{c}{2}\left(\sum_{i=1}^{k} v_i^k + \sum_{j=k+1}^{n} v_j^s\right) - \frac{(k-1)}{\sum_{i=1}^{k}\frac{1}{v_i^k}}, \tag{3.20}$$

Since in the partial support case, nodes indexed $j > k$ are not included in the Attacker's support, the Defender is always better off choosing the maximum possible value for them. Since the upper boundary for nodes outside the Attacker's support is defined by convex constraint (3.13), it follows that it is always optimal for the Defender to choose the value for nodes indexed $j > k$ such that (3.13) binds. In this case, the Attacker is indifferent between adding nodes of value v^s to his support or not, as by Claim 2.7 in both cases, his expected payoffs are equivalent. Therefore, by choosing the maximum possible value for the nodes outside the Attacker's support, the Defender increases the overall value of the network without increasing her expected loss from an attack.

Lemma 3.1. *Consider the subset of networks with partial Attacker's support, $k < n$. In equilibrium, any node outside the Attacker's support must attain a value equal to the expected gain of the Attacker from an attack on top k nodes.*

Proof of Lemma 3.1. See Appendix A. ■

We now state the relaxed maximisation problem for the partial support case as:

$$\max_{v_i} \quad \left(1-\frac{c}{2}\right)\sum_{i=1}^{n} v_i - \frac{(k-\delta)}{\sum_{i=1}^{k}\frac{1}{v_i}} \tag{3.21}$$

$$\text{s.t.} \quad v_i \in (\Upsilon^P). \tag{3.22}$$

The set of constraints which define (Υ^P) can be found in Appendix C.4.

In Appendix C.3 we demonstrate that in addition to extreme point families (3.16)–(3.19), set (Υ^C) yields three more:

1. k-family:

$$\left(\underbrace{\frac{k}{k-1},\cdots,\frac{k}{k-1}}_{\times k},\underbrace{1,\cdots,1}_{\times(n-k)}\right); \tag{3.23}$$

   where $k \in [2, n-1]$.

2. Quasi-star family:

$$\left(n-1,\underbrace{\frac{k}{k-1},\cdots,\frac{k}{k-1}}_{\times(k-1)},\underbrace{\frac{k-1}{\frac{(k-1)^2}{k}+\frac{1}{n-1}},\cdots,\frac{k-1}{\frac{(k-1)^2}{k}+\frac{1}{n-1}}}_{\times(n-k)}\right); \tag{3.24}$$

   where $k \in [2, n-1]$.

3. Lower partial family:

$$\left(n-1,\frac{k(n-1)}{kn-2k-n+1},\underbrace{\frac{k}{k-1},\cdots,\frac{k}{k-1}}_{n-2}\right), \tag{3.25}$$

   where $k \in [3, n-1]$.

Global maximum We can now find all global maximisers for the relaxed maximisation problem. Via the Bauer maximum principle, the Defender must attain the maximum payoff by choosing one of the sequences that belong to families (3.16)–(3.19) or (3.23)–(3.25). We also know from Lemma 3.1 that the maximising degree sequence is conditional on the cost of a single edge, c. Thus, by comparing the Defender's expected payoffs from games on sequences defined by extreme points of sets Υ^C and Υ^P for a given c, we can obtain all of the maximisers for the relaxed maximisation problem.

In Lemma 3.2, we demonstrate that the relaxed maximisation problem for the base case with $\delta = 1$ has three global maximisers: (1) a complete network if the cost of an edge is sufficiently low, $c < c_l$, (2) a sequence of an intermediate family with $q = 2$ if the cost of an edge attains intermediate values, $c_l < c < c_u$, and (3) a star network if the cost of an edge is sufficiently high, $c_u < c < 2$.

Lemma 3.2. *The relaxed maximisation problem has the following solutions: the optimal network for the Defender is a complete network if $c < c_l$, a sequence of the intermediate family with $q = 2$ if $c_l < c < c_u$, and a star network if $c_u < c < 2$, where $c_l = 2 - \frac{2}{n}$ and*

$$c_u = 2 - \frac{2n - 4}{n^2 - 2n + 2}.$$

The Defender is indifferent between a complete network and a sequence of the intermediate family with $q = 2$ if $c = c_l$, and between a sequence of the intermediate family with $q = 2$ and a star network if $c = c_u$.

Proof of Lemma 3.2. See Appendix A ■

However, the solutions for the relaxed maximisation problem might not be feasible for the original problem since integrality (3.3) and Erdos-Gallai (3.5) constraints were relaxed. In the following subsection, we check whether the integrality and Erdos-Gallai constraints are satisfied by candidate solutions described in Lemma 3.2.

3.3 Feasibility of solutions

By Lemma 3.2 the relaxed maximisation problem yields three potential maximisers for the original problem: (1) a star network, (2) a complete network, or (3) an intermediate family sequence with two nodes of value $v^q = n - 1$. While a star network and a complete network are always graphic, it is not always the case for the sequence of the intermediate family (3.18) with $q = 2$. For instance, consider the sequence of the intermediate family (3.18) with $n = 7$ and $q = 2$. This sequence has two nodes of value 6 and five nodes of value 3. Since the overall sum of degrees in this sequence is odd ($2 \times 6 + 5 \times 3 = 27$), the sequence is not graphic.

In this subsection, we demonstrate that when the intermediate sequence with $q = 2$ is not feasible, the Defender then chooses between star, complete, or a sequence of the intermediate family with the minimum possible q such that this sequence is graphic. In Subsection 3.4 we numerically verify that this result holds for networks of up to 20 nodes.

First, we rule out the possibility that when the sequence of the intermediate extreme point family with $q = 2$ is not feasible, the Defender chooses a sequence that belongs to one of the families (3.19) or (3.23)–(3.25). This is because the majority of extreme points that belong to families (3.19) and (3.23)–(3.25) yield sequences that are not graphic (or even integer). The only exception is a degree sequence that belongs to k-family (3.23) with $n = 4$, but it is always dominated by a star network. Therefore, it cannot be an equilibrium solution to the original maximisation problem.

Claim 3.2. *Sequences that belong to a lower family* (3.19)*, quasi-star family* (3.24)*, and lower partial family* (3.25) *are never graphic; the k-family of extreme points* (3.23) *yields one sequence that is graphic when $n = 4$ and $k = 2$, but a star network always dominates this sequence.*

Proof of Claim 3.2. See Appendix A. ■

However, if a sequence of the intermediate family with $q = 2$ is not graphic, it might still be possible for the Defender to choose another sequence that belongs to the intermediate family with $q \geq 2$ and graphic. Moreover, any of those sequences dominate star and complete networks for intermediate values of c.

Lemma 3.3. *The Defender prefers a feasible intermediate sequence to star and complete networks if $c_l < c < c_u^q$, where $c_l = 2 - \frac{2}{n}$ and*

$$c_u^q = 2 - \frac{(n-1)(6q-4) - 2n^2(q-1)}{(n^2 - 2n + 2)\,(n(q-1) - q)}.$$

Proof of Lemma 3.3. See Appendix A. ■

Moreover, the Defender is always better off choosing an intermediate sequence with the lowest possible amount of nodes of value $v^q = n-1$. To see this, consider the following maximisation problem in which the Defender chooses the number of nodes of value v^q, optimising the expected value at the beginning of the game. Substituting $v^k = n - 1$ and $v^s = \frac{(q-1)(n-1)}{q}$ into the objective function (3.20) yields the following univariate maximisation problem:

$$
\begin{aligned}
\max_{q} \quad & U_I = -\frac{(n-1)(n(c-2)(q-1)+cq-2)}{2q} \\
\text{s.t.} \quad & 2 \leq q \leq n.
\end{aligned}
\tag{3.26}
$$

Choosing the minimum possible quantity of fully connected nodes when the game is played on a intermediate family sequence always yields the highest payoff for the Defender.

Lemma 3.4. *The Defender's expected payoff from a game on a intermediate family sequence* (3.18) *increases in quantity of completely connected nodes q if $c < 2 - \frac{2}{n}$, and decreases in q if $c > 2 - \frac{2}{n}$. The Defender is indifferent if $c = 2 - \frac{2}{c}$.*

Proof of Lemma 3.4. See Appendix A. ■

It follows from Lemmas 3.3 and 3.4 that whenever a sequence of the intermediate family is optimal for the Defender, she must choose the minimum possible number of fully connected nodes q.

We now analyse the conditions under which intermediate family sequences are graphic and feasible. We say that if some intermediate sequence is graphic, it yields a *maxi-core network.*

Definition 3.3 (Maxi-core Networks)**.** *A q-maxi-core network is a network which has q vertices of value $v^q = n-1$, which constitute the core and $n-q$ peripheral vertices of value $v^b = \frac{(n-1)(q-1)}{q} \in \mathbb{Z}$. We refer to a q-maxi-core network with minimum possible number of completely connected vertices $m = \min\left\{q : \frac{(n-1)(q-1)}{q} \in \mathbb{Z}\right\}$ as an m-maxi-core network.*

The example of an m-maxi-core network on nine nodes is demonstrated in Figure 5. The illustrated network has two core nodes, $q = 2$, and seven peripheral nodes.

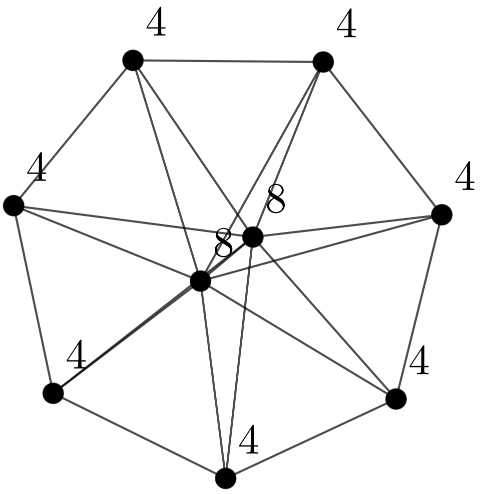


Figure 5: M-maxi-core network on nine vertices with two vertices in the core.

In Appendix D we demonstrate that any sequence of the intermediate extreme point family always satisfies Erdos-Gallai constraints (3.5) if $q \in [2, n-2]$. However, q-maxi-core networks are feasible if and only if the Defender has an odd number of nodes[12]. If the Defender has an even number of nodes, q-maxi-core networks are never feasible since for any $q \in [2, n-2]$, the value of a peripheral node is either not an integer, $v^b = \frac{(n-1)(q-1)}{q} \notin \mathbb{Z}$, or if it is an integer then the sum of all degrees is odd, which results in a non-graphic sequence.

It follows that if the Defender has an even number of nodes, the only two options that she has are a star or a complete network.

Proposition 3.4. *If the Defender has an even number of nodes and chooses among all feasible sequences which appear as extreme points of* (Υ^C) *and* (Υ^P)*, a star network yields the largest payoff if* $c > c_i$*, and a complete network if* $c < c_i$*, where*

$$c_i = 2 - \frac{2\left(n^2 - 2 + 1\right)}{n\left(n^2 - 2n + 2\right)}$$

The Defender is indifferent between a star and complete network if $c = c_i$*.*

Proof of Proposition 3.4. See Appendix A. ■

If the Defender has an odd number of nodes, she also has an intermediate option—an *m-maxi-core network.* Utilising the results of Lemma 3.4 and Proposition 3.2, we can

[12]For instance, it is always possible to create a network with $q = \frac{n-1}{2}$. In this case, any peripheral node has value $v^b = n - 3$. Such a sequence is always is always graphic as it satisfies Erdos-Gallai constraints (3.5) and the sum of the degrees is even.

formalise the optimality conditions for all feasible networks of the Defender in case she has an odd number of nodes.

Proposition 3.5. *If the Defender has an odd number of nodes and chooses among all feasible sequences which appear as extreme points of* (Υ^C) *and* (Υ^P)*, a star network yields the largest payoff if* $c > c_u^m$*, an m-maxi-core network if* $c_u^m > c > c_l$*, and a complete network if* $c < c_l$*, where* $c_l = 2 - \frac{2}{n}$ *and*

$$c_u^m = 2 - \frac{(6m-4)(n-1) - 2(m-1)n^2}{(n^2 - 2n + 2)\left((m-1)n - m\right)},$$

where $m = \min\left\{q : \frac{(n-1)(q-1)}{q} \in \mathbb{Z}\right\}$.

The Defender is indifferent between a star network and an m-maxi-core network if $c = c_u^m$ *and between an m-maxi-core network and a complete network if* $c = c_l$.

Hence, if an edge is sufficiently cheap, it is optimal for the Defender to have a complete network, despite the potential high loss from an attack, as the loss can be balanced out by value induced by increased connectivity. On the contrary, when the cost of a link is too expensive, it is efficient for the Defender to choose the least connected network with the highest defence capabilities—a star network. If the cost of a connection is moderate, the complete network is still too expensive, but the Defender can increase the efficiency by choosing an m-maxi-core network.

Propositions 3.4 and 3.5 also demonstrate that contrary to the literature (e.g. Goyal & Vigier, 2010, 2014), a star network does not yield the highest payoff for the Defender in most scenarios and appears as the equilibrium only when the cost of connection is sufficiently high.

In the following subsection, we numerically verify that the optimal solution described in Propositions 3.4 and 3.5 are indeed the solutions to the original game.

3.4 Numerical verification

Equilibrium analysis in the previous subsection relies on the results obtained with the help of the convex relaxation method. To back up our findings, we offer a numerical verification of our results in this subsection. We perform an iterative maximisation procedure to find degree sequences that maximise the Defender's expected payoff at the beginning of the game for a given number of nodes, n, and a given cost of an edge, c. We describe the numerical optimisation procedure for a given number of nodes below.

1. We compose the set of degree sequences, D_G, of all known complete graphs based on the *Wolfram GraphData database* (Wolfram Research, 2007). Each entry in the data set is a degree sequence corresponding to some connected graph with n nodes. For $n \leq 10$ the database is exhaustive.

2. We assign each entry of the data set D_G to a certain class of networks based on the minimum size of the Attacker's support $|\operatorname{supp} A| = k$ utilising condition (2.4).

3. We then create a set of corresponding expected utility functions, D_U. Each entry of D_U is the Defender's expected utility function of the form:

$$U(c) = \left(1 - \frac{c}{2}\right) \sum_{i=1}^{n} v_i - \frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}}, \quad i \in [1, n].$$

   where the size of the Attacker's support k corresponds to the class of the networks the entry belongs to.

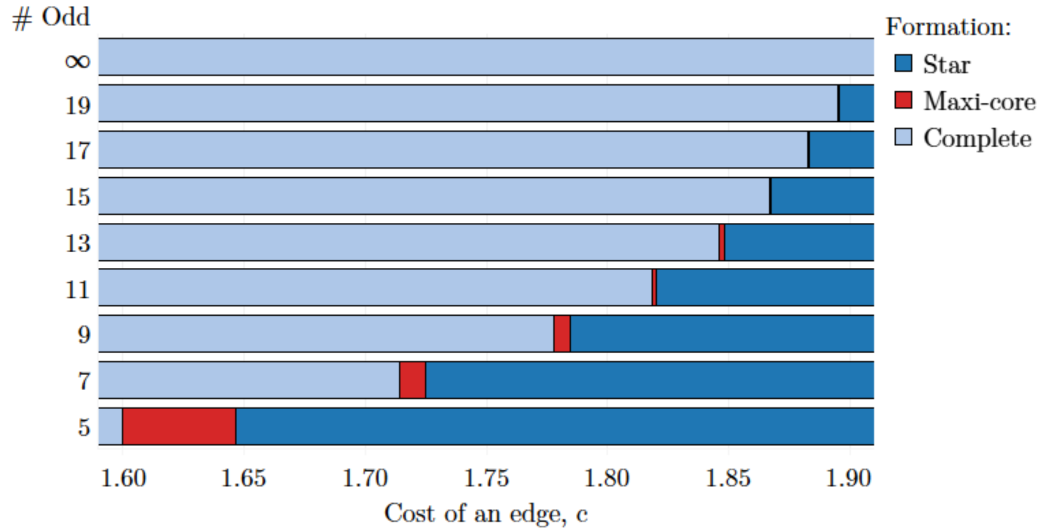
4. We then iterate the cost of an edge, c, with a stepsize 10^{-5} and find the entry in a set D_U which attains the maximum value for a given c, $\max\{D_U \ : \ c\}$.

5. The entry with a maximum value for a given c is then traced back to the original degree sequence in D_G.

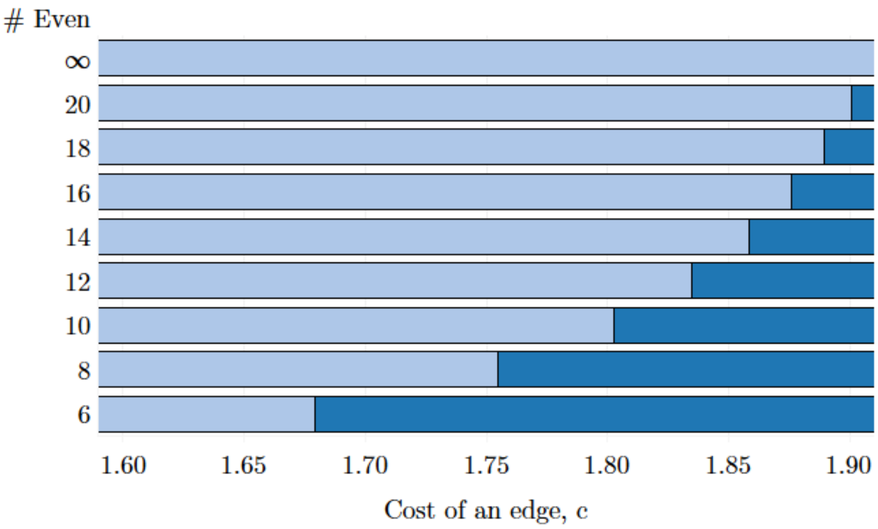
6. Finally, we create a data set of all global maximisers for the corresponding ranges of cost value c.

The iterative numerical procedure allows us to efficiently identify global maximisers for the original integer non-linear maximisation problem (3.2) for networks with $n \leq 20$ nodes. The results of the numerical optimisation are illustrated in Figure 6. Figure 6a demonstrates that when the Defender has an odd number of nodes $n \leq 19$ and a single defensive resource ($\delta = 1$), three networks arise in the equilibrium: a complete network if the cost of an edge is sufficiently low, an m-maxi-core network if the cost of an edge is moderate, and a star network if the cost of an edge is sufficiently high. At the same time, if the Defender has an even number of nodes, her choice is limited to either a complete network, if the cost of an edge is sufficiently small, or a star network otherwise. Thus, the numerical analysis confirms that the optima identified in Propositions 3.4 and 3.5 are indeed the solutions to the game if $n \leq 20$.

Observe also from Figure 6a that the interval of m-maxi-core network optimality diminishes in the number of nodes. For instance, whenever $n = 19$, maxi-core network is an optimal choice for the Defender only if $1.89474 < c < 1.89521$. Thus, while this network might appear in equilibrium when the number of nodes is small, it might not be an optimal choice for the Defender if she possesses a large number of nodes.



(a) Maximising degree sequences with an odd number of vertices



(b) Maximising degree sequences with an even number of vertices

Figure 6: Results of numerical optimisation. See Appendix E for precise values of cost boundaries and more details on optimal maxi-core networks.

It is also evident that the cost range for complete network optimality increases in the number of nodes for networks with both even and odd quantities of vertices, which confirms the result of Proposition 3.3. Thus, with an increasing number of nodes, eventually, a complete network becomes the only optimal choice for the Defender.

To conclude, for the base case with a single defensive resource $\delta = 1$, the solutions to the original problem could be found utilising a convex relaxation technique and convex hull analysis. We observe that if some extreme point of the convex hull of the set of the original problem yields an integer and graphic sequence, this sequence is highly likely to be the solution for the original problem. This echoes the results of Geoffrion (1971), who stated that if a relaxation of some integer maximisation problem yields a solution that is feasible for the original problem and the objective function in both original and relaxed problems are the same, then this feasible solution to the relaxed problem is highly likely to be the solution to the original MINLP problem.

However, the numerical verification must be taken with caution. Firstly, our database is complete only for networks with $n \leq 10$, while if $n \in [11, 20]$, it includes only classes of graphs discovered and extensively described in graph theory literature. Still, for a large number of nodes, the optimality range for the complete network covers almost the entire feasible interval for the cost of a single connection, and, therefore, a complete network must be the only optimal choice. Secondly, this method might overlook some maximising degree sequence if $\delta \geq 2$; we discuss this limitation in Section 5.

4 Weighted version of the model

In our benchmark model, in case of a successful attack, the Defender loses the value of a compromised vertex. However, there are two more scenarios to consider: (1) when a successful attack results in a partial node loss rather than a complete loss, and (2) when the damage from a successful attack is much more considerable than the value of vertex alone. Both of those scenarios can be seen in military logistics planning[13], civil transportation planning[14], and cybersecurity.

[13]For instance, Rogers et al. (2018) utilised simulations and "what if?" approach to analyse efficient military logistics planning. The authors considered a situation in which some nodes and vertices of military logistics networks are either completely destroyed or disrupted by an enemy.

[14]Ye and Kim (2019) analysed the vulnerability of heavy rail system networks and considered the influence of complete and partial node failure on the efficiency of the railway networks.

In cybersecurity, scenarios in which some node is not lost entirely or is simply disrupted can be observed in cloud computing protection. For instance, a DDoS attack might disrupt a cloud server dealing significant damage, but the disruption is temporary, and the attack does not usually result in a complete loss of a cloud computing node (Deshmukh & Devadkar, 2015; Somani et al., 2016). However, cybersecurity incidents can often result in damage that extends way beyond the value of a damaged node. For instance, Spanos and Angelis (2016), following their systematic literature review on the impact of information breaches on the stock market, revealed that the majority of studies (25 studies out of 28) find a negative impact of data breaches on the victim's stock price. Moreover, companies may face significant financial losses due to the reputational damage caused by a data breach. Schwartz and Janger (2007) argue that the contemporary legal regime is focused on incentivising business entities to address cybersecurity risks by imposing significant reputational sanctions in case of a breach (e.g. mandatory public disclosure of a breach). Makridis (2021) demonstrates that a data breach could lead to an up to 9% decline in reputational intangible capital, which inevitably results in a significant loss of long-run profits.

We introduce the weighted version of the model to account for the possibility of augmentation or reduction of cyberattack damage. In this extension, we allow the expected value function of the Defender to be a weighted linear combination of the overall network value and the expected loss from an attack. By changing the linear coefficients, we can adjust the Defender's expected value function to situations in which an attack's damage extends beyond the value of a vertex or constitutes only a share of the vertex's value. The objective function of the modified Defender's problem then has the following form:

$$\max_{v_i} \quad \alpha \left(\sum_{i=1}^{n} v_i - \frac{c}{2} \sum_{i=1}^{n} v_i \right) - (1-\alpha) \frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}}, \tag{4.1}$$

where $\alpha \in [0, 1]$.

We now demonstrate that depending on the value of linear coefficient α, the cost boundaries between different networks can shift.

Lemma 4.1. *Any cost boundary, c_i, weakly increases in the weight of the overall network value, α.*

Proof of Lemma 4.1. See Appendix A ■

Given that introduction of the weights does not affect the set of feasible degree sequences Γ, it must be the case that the set of potential maximising degree sequences for the Defender remains the same as in the unweighted case. However, Lemma 4.1 suggests that depending on the weight α, cost boundaries that separate different maximising degree sequences can shift. In practice, if the Defender prioritises the overall network value over the potential losses from an attack ($\alpha > 0.5$), denser networks will appear as maximisers for the broader range of c. For instance, it might happen if the Defender is confident that an attack will not cause any reputation losses and the compromised vertex can be easily replaced (e.g. a cyberattack on some NGO, which does not possess any financial data or trade secrets). On the contrary, if an event of a cyberattack is capable of significant reputational damage or can even lead to the loss of a competitive edge, the Defender might put a higher weight on an expected loss. In this case, the cost range for which less dense networks are preferred would be extended (e.g. a cyberattack on a financial organisation, which heavily relies on clients' data).

Moreover, the cost range for some networks' optimality could be diminished or increased for the different values of α, as cost boundaries rates of change w.r.t. α might differ. For instance, in the base case with $\delta = 1$, if the Defender puts high weight on an expected loss from an attack ($\alpha < \frac{1}{2}$), the interval of m-maxi-core optimality might be significantly increased.

Claim 4.1. *If the Defender has one defensive resource $\delta = 1$, the optimality range for an m-maxi-core network, $\delta_{MC} = c_u^\alpha - c_l^\alpha$, strictly decreases in α, $\frac{\partial \delta_{mc}}{\partial \alpha} < 0$.*

Proof of Claim 4.1. See Appendix A. ■

Thus, it follows from Claim 4.1 that if $\alpha < \frac{1}{2}$ the optimality ranges for m-maxi-core networks can be larger than the ones demonstrated in Figure 6a.

Furthermore, observe that the boundaries can shift to the point where only a complete or star network can be optimal for the Defender. When $\alpha \to 1$, the benefits of building a well-connected network vastly outweigh the security risks, while when $\alpha \to 0$, the damage from an attack is so severe that the Defender's only concern is network security. This observation also implies that shifting the boundaries through changing the weight α can influence the size of the set of maximising degree sequences.

5 Limitations and future research

This section offers a concise discussion of limitations and future research.

5.1 Limitations

Section 3 of this paper relies on the convex relaxation technique and characterises the solution for the relaxed Defender's maximisation problem. This technique has two limitations:

Non-convex regions of Γ. By convexifying set Γ and extending the set of feasible values to the convex hull, we potentially might lose some global maximisers that appear on the non-convex regions of the original set. While we numerically verified that the global maximisers for the original problem indeed coincide with the analytical solution to the relaxed problem when $\delta = 1$ and $n < 20$, it might not be true for a more general case. Moreover, any potential error from a convex relaxation approach is likely to be small, because a complete network is optimal for a wide range of c when $n \leq 20$ (e.g. for any $c < 1.9$ when $n = 20$) and will be optimal for an even bigger range of c as n increases further.

Relaxation of Erdos-Gallai graphicality conditions. During the integer relaxation performed as a part of a convex relaxation, we relaxed the integrality constraints (3.3) and Erdos-Gallai sufficient and necessary conditions for the sequence graphicality. While the former should not lead to a loss of solutions since the objective function (3.2) is convex, the latter might result in losing some of the extreme points defined by graphicality conditions. While we numerically verified for $\delta = 1$ and $n < 20$ that it is not the case, it might happen when the Defender has some ad-hoc number of defensive resources $\delta \geq 2$. This is because the expected gain from an attack strictly decreases in δ, implying that the Attacker's support conditions (3.6) are the tightest when $\delta = 1$. Thus, with additional defensive resources and loosening of the Attacker's support condition (3.6), the Erdos-Gallai conditions play a more prominent role. For instance, one can observe that the degree sequence $(4, 2, 2, 1, 1)$ in Figures 2 and 3 arises as a global maximiser for some intermediate cost of a single edge $c_i < c < c_u$. However, this sequence does not appear as an extreme point of the set defined by linear constraints (3.4) or Attacker's support constraints (3.6) and (3.7). Therefore, it must be the case that this degree sequence results from the intersection of Erdos-Gallai conditions with some other constraints of the set. It also implies that with additional defensive resources, the problem might have a larger set of maximising degree sequences.

To conclude, the presented approach might not be efficient in solving the Defender's optimisation when $\delta \geq 2$. We see two possibilities on how to approach a more general case:

1. **Numerical estimation**. The iterative numerical estimation demonstrated its efficiency for reasonably small networks. Once one possesses the complete data set of all available graphs on n nodes, there should be no computational limitations for obtaining optimal solutions for a given value of c. However, given the size of the data sets of all connected networks for a larger number of nodes, this method might not be efficient for larger n. In this case, the approximate solutions can be obtained utilising heuristic methods for non-convex and combinatorial optimisation e.g. Eichfelder et al., 2021; Xu et al., 2020. These methods are out of the scope of this paper.

2. **Scope reduction**. It might also be possible to approach a general case with $\delta \geq 2$ by narrowing down the set of possible networks. For instance, reducing the scope to bipartite graphs[15] or networks which only have two types of nodes will result in a much simpler model, which might have an analytical solution for the general case.

5.2 Future research

Large networks The presented model demonstrates that the only network which appears in the equilibrium of a game with a large number of nodes ($n \to \infty$) is a complete network. This is because a successful attack can compromise only an infinitesimal share of the network's overall value. We consider developing an alternative setting with a more sophisticated vertex value function (e.g. a network size inflated vertex value function $v_i = nd_i$, where d_i is the vertex's degree centrality and n is the number of vertices in the network), which might lead to new results about the formation of large networks.

[15] A bipartite graph is a graph whose nodes can be separated into two disjoint sets T_1 and T_2, i.e. is every edge connects a node in set T_1 to one in set T_2.

Multiple offensive resources The baseline model considers the case where the Attacker has a single offensive resource. The framework is helpful for analysing the enterprise's cybersecurity capabilities against targeted information leakage attacks, as it is usually sufficient to compromise a single entry point to deal severe damage to the enterprise. Nevertheless, it does not cover the situation in which the network faces multiple attacks simultaneously (e.g. airstrikes on the military supply chain in several directions). The presented model can be extended to account for this possibility, but finding the equilibrium might be intractable in the general case. This is because if the Attacker has multiple offensive resources, the Attacker's support condition will no longer be monotonic, which makes it challenging to derive the corresponding constraints for the optimisation problem. We consider the development of an alternative attack mechanism in which the Attacker can choose some set of nodes for an attack following a defined rule (e.g. he can choose to attack all nodes with a specific value or choose a share of nodes in the network which will be attacked at random). This extension can provide insights into the network defence against various attack regimes.

Probabilistic defence function The deterministic defence function (when the Defender always wins the contest on the same vertex as the Attacker) might not correspond well to some real-life scenarios (e.g. air traffic security). Our model might be extended by imposing some probabilistic contest success function in the fashion of a rent-seeking contest (Gradstein & Konrad, 1999). We consider this extension for future research.

Edge attacks The baseline model assumes that the Attacker can only compromise nodes. Nevertheless, it is still possible to encounter situations where the attack targets connections between the nodes rather than the nodes themselves. We consider building an extended model which allows the Attacker to attack both edges and nodes to account for this opportunity.

Artificial intelligence Our game-theoretical formulation of the network design and defence problem can be used to create more sophisticated reward functions and serve as a benchmark for comparing reinforcement learning algorithms. We consider this possibility for future research.

6 Conclusion

In this study, we characterise efficient network formation in the presence of an intelligent attacker. In particular, the study investigates the trade-off between efficiency and security when the values of vertices are determined endogenously and are allowed to be heterogeneous. The model produces three main results. Firstly, it shows that a star network, contrary to the literature (e.g. Goyal & Vigier, 2010), does not generally yield the highest payoff for the Defender. A star network yields the lowest expected loss from an attack. Still, it is the least efficient connected network structure, which appears in equilibrium only if the cost of a single connection is sufficiently high. Secondly, we demonstrate that the density of an optimal network decreases with an increase in the cost of an edge. Finally, the study reveals a novel type of network, which appears in the equilibrium of games with scarce defensive resources—a maxi-core network with a completely connected core and sparse periphery. Moreover, we have presented a weighted model modification, which allows us to account for potential reputational losses from a cyberattack and partial vertex loss.

References

Acemoglu, D., Malekian, A., & Ozdaglar, A. (2016). Network security and contagion. *Journal of Economic Theory*, *166*, 536–585.

Anderson, R. (2001). *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley.

Bauer, H. (1958). Minimalstellen von Funktionen und Extremalpunkte. *Archiv der Mathematik 1957 9:4*, *9*(4), 389–393.

Boyd, S. S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Cerdeiro, D., Dziubiiski, M., & Goyal, S. (2015). *Contagion Risk and Network Design (FEEM Working Paper No. 056.2015)*.

Deshmukh, R. V., & Devadkar, K. K. (2015). Understanding DDoS Attack & its Effect in Cloud Environment. *Procedia Computer Science*, *49*, 202–210.

Dresher, M. (1961). *Games of Strategy: Theory and Applications*. Prentice Hall.

Dziubiński, M., & Goyal, S. (2013). Network design and defence. *Games and Economic Behavior*, *79*(1), 30–43.

Dziubiński, M., & Goyal, S. (2017). How do you defend a network? *Theoretical Economics*, *12*(1), 331–376.

Eichfelder, G., Klamroth, K., & Niebling, J. (2021). Nonconvex constrained optimization by a filtering branch and bound. *Journal of Global Optimization*, *80*(1), 31–61.

Erdős, P., & Gallai, T. (1960). Gráfok előírt fokszámú pontokkal. *Matematikai Lapok*, *11*, 264–274.

Geoffrion, A. M. (1971). Duality in Nonlinear Programming: A Simplified Applications-Oriented Development. *SIAM Review*, *13*(1), 1–37.

Goyal, S. (2007). *Connections: An introduction to the economics of networks*. Princeton University Press.

Goyal, S., Jabbari, S., Kearns, M., Khanna, S., & Morgenstern, J. (2016). Strategic Network Formation with Attack and Immunization. In E. Bertino, W. Gao, B. Steffen, & M. Yung (Eds.), *Lecture notes in computer science* (pp. 429–443, Vol. 10123). Springer.

Goyal, S., & Vigier, A. (2010). *Robust Networks (STICERD Working Paper)*. https://sticerd.lse.ac.uk/seminarpapers/et11032010.pdf

Goyal, S., & Vigier, A. (2014). Attack, Defence, and Contagion in Networks. *The Review of Economic Studies*, *81*(4), 1518–1542.

Gradstein, M., & Konrad, K. A. (1999). Orchestrating Rent Seeking Contests. *The Economic Journal*, *109*(458), 536–545.

Gueye, A., Walrand, J. C., & Anantharam, V. (2011). How to Choose Communication Links in an Adversarial Environment? In R. Jain & R. Kannan (Eds.), *Game theory for networks* (pp. 233–248). Springer.

Hart, S. (2008). Discrete Colonel Blotto and General Lotto games. *International Journal of Game Theory*, *36*(3-4), 441–460.

Hoyer, B., & Jaegher, K. D. (2016). Strategic Network Disruption and Defense. *Journal of Public Economic Theory*, *18*(5).

Jackson, M. O. (2010). *Social and Economic Networks*. Princeton University Press.

Konrad, K. A. (2009). *Strategy and Dynamics in Contests*. Oxford University Press.

Kovenock, D., & Roberson, B. (2008). Electoral Poaching and Party Identification. *Journal of Theoretical Politics*, *20*(3), 275–302.

Kovenock, D., & Roberson, B. (2012). Conflicts with Multiple Battlefields. In M. R. Garfinkel & S. Skaperdas (Eds.), *The oxford handbook of the economics of peace and conflict* (pp. 503–531). Oxford University Press.

Kovenock, D., & Roberson, B. (2021). Generalizations of the General Lotto and Colonel Blotto games. *Economic Theory*, *71*(3), 997–1032.

Li, D., Sun, X. L., Biswal, M. P., & Gao, F. (2001). Convexification, Concavification and Monotonization in Global Optimization. *Annals of Operations Research*, *105*, 213–226.

Makridis, C. A. (2021). Do data breaches damage reputation? Evidence from 45 companies between 2002 and 2018. *Journal of Cybersecurity*, *7*(1).

Naumowicz, A. (2014). A Note on the Seven Bridges of Königsberg Problem. *Formalized Mathematics*, *22*(2), 177–178.

Pulleyblank, W. (1989). Chapter V Polyhedral combinatorics. In *Handbooks in operations research and management science* (pp. 371–446).

Roberson, B. (2006). The Colonel Blotto game. *Economic Theory*, *29*(1), 1–24.

Rogers, M. B., McConnell, B. M., Hodgson, T. J., Kay, M. G., King, R. E., Parlier, G., & Thoney-Barletta, K. (2018). A military logistics network planning system. *Military Operations Research*, *23*(4), 5–25.

Sahinidis, N. V. (2019). Mixed-integer nonlinear programming 2018. *Optimization and Engineering*, *20*(2), 301–306.

Schwartz, P. M., & Janger, E. J. (2007). Notification of data security breaches. *Michigan Law Review*, *105*(5).

Slikker, M., & van den Nouweland, A. (2000). Network formation models with costs for establishing links. *Review of Economic Design*, *5*(3), 333–362.

Somani, G., Gaur, M. S., Sanghi, D., & Conti, M. (2016). DDoS attacks in cloud computing: Collateral damage to non-targets. *Computer Networks*, *109*, 157–171.

Spanos, G., & Angelis, L. (2016). The impact of information security events to the stock market: A systematic literature review. *Computers & Security*, *58*, 216–229.

Sýkora, S. (2009). Mathematical means and averages: basic properties.

Tripathi, A., & Vijay, S. (2003). A note on a theorem of Erdős & Gallai. *Discrete Mathematics*, *265*(1-3), 417–420.

Tuy, H., & Van Thuong, N. (1988). On the global minimization of a convex function under general nonconvex constraints. *Applied Mathematics and Optimization*, *18*(1), 119–142.

von Neumann, J. (1928). Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, *100*(1), 295–320.

Wang, Y., Sinha, A., Ch-Wang, S., & Wellman, M. P. (2021). Building Action Sets in a Deep Reinforcement Learner. *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 484–489.

Washburn, A. (2013). OR Forum—Blotto Politics. *Operations Research*, *61*(3), 532–543.

Weisstein, E. W. (n.d.). Squeezing Theorem. Retrieved May 9, 2022, from https://mathworld.wolfram.com/SqueezingTheorem.html

Wolfram Research. (2007). GraphData [Data set]. https://reference.wolfram.com/language/ref/GraphData.html

Xu, P., Roosta, F., & Mahoney, M. W. (2020). Newton-type methods for non-convex optimization under inexact Hessian information. *Mathematical Programming*, *184*(1-2), 35–70.

Ye, Q., & Kim, H. (2019). Assessing network vulnerability of heavy rail systems with the impact of partial node failures. *Transportation*.

A Mathematical appendix

Proof of Claim 2.4. Suppose that vertex $m < l$ is not attacked in some equilibrium, in which l is attacked. Then m is not defended by Claim 2.3. The expected payoff from attacking m is v_m, which is greater than the upper bound of the expected payoff from attacking l, v_l. Therefore, there is a profitable deviation: attacking m with probability 1. If the inequality is weak, i.e. $v_l = v_m$, we can relabel the nodes and preserve the monotonicity of the payoffs in the assumption. ■

Proof of Claim 2.5. From (2.2) we have the following condition for the top k vertices to be attacked:

$$\frac{v_k}{v_1} + \frac{v_k}{v_2} + \cdots + \frac{v_k}{v_{k-1}} + \frac{v_k}{v_k} \geq k - \delta.$$

Rearranging and simplifying yields:

$$\frac{v_k}{v_1} + \frac{v_k}{v_2} + \cdots + \frac{v_k}{v_{k-1}} \geq k - 1 - \delta,$$

which, because $v_{k-1} \geq v_k$, implies condition for the top $k-1$ vertices. ■

Proof of Claim 2.6. Suppose the condition (2.4) holds strictly for some node k and does hold for node $k+1$.

The Attacker's gain from an attack on top k nodes is then:

$$A(\mathbf{v}_k) = \frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}}.$$

If the Attacker chooses to mix over $k-1$ nodes, then by Lemma 2.4, the Defender protects only $k-1$ nodes with positive probability. Then, the Attacker's gain is

$$A(\mathbf{v}_{k-1}) = \frac{(k-\delta-1)}{\sum_{i=1}^{k-1} \frac{1}{v_i}}.$$

We now demonstrate that:

$$A(\mathbf{v}_k) > A(\mathbf{v}_{k-1}). \tag{A.1}$$

Expanding (A.1) yields:

$$\frac{(k-\delta)}{\sum_{i=1}^{k} \frac{1}{v_i}} > \frac{(k-\delta-1)}{\sum_{i=1}^{k-1} \frac{1}{v_i}},$$

$$(k-\delta)\sum_{i=1}^{k-1} \frac{1}{v_i} > (k-\delta-1)\sum_{i=1}^{k} \frac{1}{v_i},$$

$$(k-\delta)\sum_{i=1}^{k-1} \frac{1}{v_i} - (k-\delta-1)\sum_{i=1}^{k} \frac{1}{v_i} > 0,$$

and finally:

$$v_k > \frac{(k-\delta-1)}{\sum_{i=1}^{k-1} \frac{1}{v_i}},$$

which is always satisfied by the assumption that that condition (2.4) holds strictly for node k.

Thus, if the Attacker deviates to smaller support, he receives a strictly smaller payoff.

■

Proof of Claim 2.7. Suppose that for some node y condition (2.4) holds strictly, while for nodes $\nu \in (y, y+z]$, where $z \in [1, n-y]$ and $z \in \mathbb{Z}$, it holds with equality.

The expected gain of the Attacker from an attack on top k nodes is:

$$A(\vec{v_y}) = \frac{(y-\delta)}{\sum_{i=1}^{y} \frac{1}{v_i}}.$$

Then it follows from condition (2.4) must be the case that each node indexed $\nu \in (y, y+z]$ must be $v_\nu = A(\vec{v_y}) = -D(\vec{v_y})$.

Then the expected payoff from an attack on top $y+z$ nodes is:

$$\begin{aligned} A(\vec{v}_{y+z}) &= \frac{(y+z-\delta)}{\sum_{i=1}^{y} \frac{1}{v_i} + \frac{z}{\frac{(y-\delta)}{\sum_{i=1}^{y} \frac{1}{v_i}}}} = \\ &= \frac{(y+z-\delta)}{\sum_{i=1}^{y} \frac{y+z-\delta}{(y-\delta)v_i}} = \frac{(y-\delta)}{\sum_{i=1}^{y} \frac{1}{v_i}} = A(\vec{v_y}). \end{aligned} \tag{A.2}$$

Since equality (A.2) is satisfied regardless of the quantity of nodes that pass condition (2.4), it must be the case that $A(\vec{v}_{y+z}) = -D(\vec{v}_{y+z})$ for any $|\,\mathrm{supp}_A\,| \in [y, y+z]$. ■

Proof of Lemma 2.1. Let $(y_1, \ldots, y_k) \in \mathbb{R}^k_+$ and $(z_1, \ldots, z_k) \in \mathbb{R}^k_+$.

Consider Defender's expected loss function (2.5) rewritten as:

$$A(y) = \frac{k - \delta}{g(y)},$$

where $g(y) = \frac{1}{y_1} + \ldots + \frac{1}{y_k}$.

Consider now function $g(y)$. Via the Cauchy-Schwarz inequality:

$$\begin{aligned} \frac{1}{y_1} + \ldots + \frac{1}{y_k} &= \left(\frac{\sqrt{z_1}}{y_1}, \ldots, \frac{\sqrt{z_k}}{y_k}\right) \cdot \left(\frac{1}{\sqrt{z_1}}, \ldots, \frac{1}{\sqrt{z_k}}\right) \\ &\leq \left(\frac{z_1}{y_1^2} + \ldots + \frac{z_k}{y_k^2}\right)^{\frac{1}{2}} \left(\frac{1}{z_1} + \ldots + \frac{1}{z_k}\right)^{\frac{1}{2}}. \end{aligned} \tag{A.3}$$

Now we rewrite the function as:

$$\left(\frac{1}{z_1} + \ldots + \frac{1}{z_k}\right)^{-1} \leq \left(\frac{z_1}{y_1^2} + \ldots + \frac{z_k}{y_k^2}\right) \left(\frac{1}{y_1} + \ldots + \frac{1}{y_k}\right)^{-2}. \tag{A.4}$$

Observe that the right-hand side of (A.4) is exactly the tangent plane of the left-hand side at some point $(y_1, \ldots, y_k)$. Since the inequality holds for *any* such point $(y_1, \ldots, y_n)$, the left-hand side is a concave function. It immediately follows that $A(\vec{v_k})$ is concave, while $D(\vec{v_k})$ is convex.

■

Proof of Proposition 2.1. Consider the expected attack on k nodes in general form:

$$A(\mathbf{v}_k) = \frac{k - \delta}{\sum_{i \in K} \frac{1}{v_i}},$$

where $K = \mathrm{supp}_A$.

Consider now the first derivative of $A(\mathbf{v}_k)$ w.r.t. the value of some node $j \in K$:

$$\frac{\partial A(\mathbf{v}_k)}{\partial v_j} = \frac{k - \delta}{\left(\sum_{i \in K} \frac{1}{v_i}\right)^2 v_j^2} > 0, \tag{A.5}$$

which is always positive.

It follows that the expected gain from an attack on k nodes attains its maximum value when all nodes attain value $v^q = n - 1$. Substituting $v_i = v^q = n - 1$, $\forall i \in K$ yields:

$$A^{max}(\mathbf{v}_k) = \frac{(k - \delta)(n - 1)}{k}. \tag{A.6}$$

Now consider the first derivative of $A^{max}(\mathbf{v}_k)$ w.r.t. k:

$$\frac{\partial A(\mathbf{v}_k)}{\partial k} = \frac{\delta(n-1)}{k^2} > 0.$$

It implies that the maximum gain from an attack is attained whenever $k = n$ and the game is played on a complete network. In this case, the Attacker's payoff is:

$$A^c = \frac{(n-1)(n-\delta)}{n}.$$

We demonstrate that the Attacker obtains the minimum payoff when the game is played on a star network. Consider two cases: (1) full support case $k = n$, and (2) partial support case with $k < n$.

Case 1. $k = n$**.** By Lemma 2.1, the expected gain from an attack on k nodes is concave. Thus, by the Bauer minimum principle, it must attain the minimum value at some extreme point of the set of feasible values. In the full support support case each $v_i \in \{1, n-1\}$. Therefore, each extreme point can be represented as the following ordered set:

$$\mathbf{v}_n^e = \left(\underbrace{n-1, \cdots, n-1}_{q}, \underbrace{1, \cdots, 1}_{n-q} \right). \tag{A.7}$$

The expected attack on the network with degree sequence (A.7) is[16]:

$$A^q(\mathbf{v}_n^e) = \frac{(n-\delta)}{\frac{q}{n-1} + n - q}.$$

Since we demonstrated that $\frac{\partial A^q(\mathbf{v}_n)}{\partial v_i} > 0$, $\forall i \in N$, it must be the case that $A^q(\mathbf{v}_n^e)$ is minimised when the Defender chooses the minimum possible number of completely connected nodes. As we assume that any network must be connected, $q^{min} = 1$[17]. Therefore, it implies that the expected gain from an attack attains its minimum when the game is played on a star network. It can be written down as follows:

$$A^s = \frac{(n-\delta)(n-1)}{(n-1)^2 + 1} < 1. \tag{A.8}$$

Observe that (A.8) is strictly smaller than 1.

[16] We ignore the fact that any node of value $v^u = 1$ would not be included in the Attacker's support for now

[17] Observe that choosing $q = 0$ yields a disconnected network with degree sequence $(1, \cdots, 1)$.

Case 2. $k < n$**.** Observe that if k nodes are included in the Attacker's support, it implies that the expected attack on the top k nodes is strictly larger than 1. If it is smaller than 1, then any node of value $v_i \geq 1$ must also be included in the Attacker's support, which yields a full support case and contradicts the partial support assumption.

Therefore, the Attacker's payoff is minimised when the game is played on a star network.

■

Proof of Proposition 3.1. Observe that expected value function of the Defender (3.1) is linear in c. Therefore, we can represent the utility function for any possible network formation as a straight down-sloping line defined over $c \in [0, 2]$. Consider two cases:
Case 1. $c = 2$**.** In this case, the Defender does not attain any value from network's connectivity and her payoff is equal to the expected loss in the encounter stage.

Now note that line equations associated with star and path networks have the lowest possible slope among all the networks, since both of those networks have the lowest possible overall nodes' value, which is equal to $S = -2(n-1)$. However, we know from Claim 2.1 that the Defender minimises her losses whenever she chooses a star, and, therefore, an equation associated with star has a larger intercept. An intercept of a line associated with a star network is $I = 2(n-1) - \frac{(n-1)(n-\delta)}{(n-1)^2+1}$. It implies that a star is certainly preferred to a path network around $c = 2$. Therefore, there must exist some point $c_u < 2$ in the neighbourhood of 2, where the line spawned by star formation is intersected by some line spawned by the network, which is optimal for the Defender if $c = c_u - \epsilon$ (where ϵ is some small positive real number). We call c_u an upper-cost boundary.
Case 2. $c = 0$. If the edges are free for the Defender, the expected value function becomes:

$$U_G(\vec{v}_n) = \sum_{i=1}^{n} v_i + D(\vec{v}_k), \tag{A.9}$$

which is also convex and, therefore, must attain its maximum at some extreme point of the set it is defined over.

Firstly, observe that (A.9) is maximised whenever the Defender chooses a complete network. Consider the first derivative of the function w.r.t. to the value of some vertex inside the Attacker's support:

$$\frac{\partial U_G(\vec{v}_n)}{\partial v_i} = 1 + \underbrace{\frac{(k-\delta)\prod_{j=1}^{k} v_j \sum_{j\in K\setminus\{i\}} \prod_{i\neq j} v_i}{\left(\sum_{j=1}^{k}\prod_{i\neq j} v_i\right)^2}}_{m_1} - \underbrace{\frac{(k-\delta)\prod_{j=1}^{k} v_j}{v_k \sum_{j=1}^{k}\prod_{i\neq j} v_i}}_{m_2} > 0, \qquad \text{(A.10)}$$

where K is the set of indices of all vertices in the Attacker's support.

Observe that term m_1 is positive and term $m_2 \leq 1$ as it is exactly the Attacker's support condition (2.4). It follows that the function (A.9) always increases in the value of vertex inside the Attacker's support. If some vertex j is outside the Attacker's support the first derivative of $U_G(\vec{v}_n)$ w.r.t. to v_j is constant and equal to one, $\frac{\partial U_G(\vec{v}_n)}{\partial v_i} = 1$.

The vertices outside the Attacker's support are also bounded from above by the Attacker's support condition. Given that the Defender chooses $v_i = v^q = n-1$, we can write down the upper boundary for the vertices outside the Defender's support as $v_j \leq \frac{(k-\delta)(n-1)}{k}$. It is now left to demonstrate that if $c = 0$, the Defender is strictly better off to choose a complete network than any other network which does not have full support:

$$n(n-1) - \frac{(n-\delta)(n-1)}{n} > k(n-1) + (n-k-1)\frac{(k-\delta)(n-1)}{k},$$

which is true for any $n > k > 2$.

We deduce that it must be the case that at extreme point $c = 0$, the Defender receives a maximum possible payoff whenever she chooses a complete network formation. It also follows that the line spawned by complete network formation must have the highest intercept point with the ordinate axis. Given that the line also must have the steepest slope, we conclude that there must exist some dot $c_l > 0$ in the neighbourhood of 0 where it is intersected by some other line spawned by the network formation, which is optimal if $c = c_l + \epsilon$. We call c_l a lower cost-boundary. ■

Proof of Proposition 3.2. Observe that the value of the network (2.1) strictly increases in v_i for any $c < 2$. We also know from Lemma 2.1 that the expected damage from an attack weakly increases in the value of a single vertex v_i.

Now observe the indifference condition for the Defender between some dense structure G_D and some sparse structure G_S:

$$V_D - \frac{c}{2}V_D + D_D = V_S - \frac{c}{2}V_S + D_S,$$

or:

$$c = 2 + \frac{D_D - D_S}{V_D - V_S} = c_i. \tag{A.11}$$

Now observe that the dense graph is preferred to the sparse graph if $c < c_i$, and vice versa if $c > c_i$. As this fact applies to any pair of network structures, it must be the case that less dense maximising degree sequences are associated with a higher cost of a single edge, while higher density maximising degree sequences are associated with a lower edge cost. ■

Proof of Proposition 3.3. If the Defender possesses $n \to \infty$ vertices and $\delta \geq 1$ defensive resources, there might be ∞ potential maximising degree sequences and ∞ cost boundaries as a result. The only fact we know about all other non-extreme potential maximising degree sequences is that they must be of lower density than the complete network. Consider some cost boundary c_i^f, which is derived by comparing the complete network and some other network with Attacker's support $k \in [\delta + 1, n]$:

$$\begin{aligned} c_i^f &= 2 - \frac{\frac{(n-\delta)(n-1)}{n} - \frac{n-\delta}{\sum_{i=1}^{k} \frac{1}{(n-1)e_i}}}{n(n-1) - \sum_{i=1}^{n}(n-1)e_i} = \\ &= 2 - \frac{(n-d)\left(\frac{1}{n} - \frac{1}{\sum_{i=1}^{k}\frac{1}{e_i}}\right)}{n - \sum_{i=1}^{n} e_i}, \end{aligned} \tag{A.12}$$

where $e_i = \frac{v_i}{n-1}$ and v_i is the value of some node i of some other network. Since $v_i \in [1, n-1]$, it follows that $e_i \in \left[\frac{1}{n-1}, 1\right]$.

Thus, e_i is a coefficient calculated as a portion of the number of connections incident upon a vertex and the maximum possible connections that the node might have $(n-1)$. Given that we assume that nodes' values of each network are weakly ordered it follows that coefficients must be order weakly ordered too: $e_1 \geq e_2 \geq \ldots \geq e_n$. Note that any incomplete network must have $1 \leq q < n$ completely connected vertices, for which $e_i^C = 1$, and $n - q$ vertices with lower degree centralities, for which $e_i^{NC} \in \left[\frac{1}{n-1}, 1\right)$.

Now, let

$$HM(e_1, \ldots, e_k) = \sum_{i=1}^{k} \frac{1}{e_i}$$

be the Harmonic Mean of first k coefficients and

$$AM(e_1, \dots, e_n) = \frac{1}{n}\sum_{i=1}^{n} e_i$$

be the Arithmetic Mean of all n coefficients.

Then expression (A.12) can be rewritten as:

$$c_i^f = 2 - \left(\frac{n-\delta}{n^2}\right)\underbrace{\left(\frac{1 - HM(e_1, \dots, e_k)}{1 - AM(e_1, \dots, e_n)}\right)}_{\tau}. \tag{A.13}$$

Now we consider the upper-bound for the numerator of τ and the lower bound of the denominator of τ.

The upper-bound of the numerator of τ can be expressed as (Sýkora, 2009):

$$1 - HM(e_1, \dots, e_k) \le 1 - \min(e_1, \dots, e_k) = 1 - e_k < 1.$$

Consider now the denominator of τ. Since e_i's are nonincreasing and $e_i < 1$ for all $i > q$, there must exist $\iota \in \mathbb{N}$ such that for all $n \ge \iota$:

$$AM(e_1, \dots, e_n) \le AM(e_1, \dots, e_\iota) < 1.$$

Then then lower bound of the denominator of τ can be expressed as:

$$1 - AM(e_1, \dots, e_n) \ge 1 - AM(e_1, \dots, e_\iota) = \tau_L > 0,$$

which is a positive constant for any fixed number $\iota \in \mathbb{N}$ and $\iota > q$.

Thus, we can now write the upper and lower bounds of expression (A.13):

$$2 - \left(\frac{n-\delta}{n^2}\right)\left(\frac{1}{\tau_L}\right) \le c_i^f \le 2,$$

where the upper boundary is achieved when the Attacker has only completely connected nodes in his support, $k \in [1, q]$.

Finally, consider now the limit of the lower boundary of c_i^f when $n \to \infty$:

$$\lim_{n\to\infty} 2 - \left(\frac{n-\delta}{n^2}\right)\left(\frac{1}{\tau_L}\right) = 2.$$

Then, by Squeeze Theorem, $c_i^f = 2$ as $n \to \infty$ (Weisstein, n.d.). ∎

Proof of Lemma 3.1. Consider the first derivative of expected utility function of the Defender (3.20) w.r.t. v_j^s:

$$\frac{\partial U^P}{\partial v_j^s} = 1 - \frac{c}{2} \geq 0.$$

Therefore, the Defender must always choose the maximum possible value for the nodes outside the Attacker's support. Observe that the upper limit of a value of a node outside that Attacker's support is given by constraint (3.13) and is equal to the expected gain from an Attack on top k nodes. ■

Proof of Lemma 3.2. From Proposition 3.1, we know that a complete and a star network must be maximisers at the extreme values of c. Therefore, we sequentially compare all the sequences (3.18)-(3.25) against a star network and a complete network and determine which sequences can act as maximisers for intermediate values of c.

First, observe the Defender's expected payoffs from a game on a star network:

$$U_* = 2\left(1 - \frac{c}{2}\right)(n-1) - \frac{(n-1)^2}{(n-1)^2+1},$$

and complete network:

$$U_C = n\left(1 - \frac{c}{2}\right)(n-1) - \frac{(n-1)^2}{n}.$$

Intermediate sequence An expected Defender's payoff from a game on an intermediate sequence (3.18) can be stated as follow:

$$U_I = \left(1 - \frac{c}{2}\right)\left(\frac{(n-1)(q-1)(n-q)}{q} + (n-1)q\right) - \frac{(n-1)(q-1)}{q},$$

where $q \in [2, n-1]$.

The Defender prefers an intermediate sequence to a star network when:

$$U_I > U_*,$$

which is satisfied IFF:

$$c < 2 - \frac{2n^2(q-1) - n(6q-4) + 6q - 4}{(n^2-2n+2)(n(q-1)-q)} = c_u. \tag{A.14}$$

Similarly, the Defender prefers an intermediate sequence to a complete network when:

$$U_I > U_c,$$

which is satisfied when:

$$c > 2 - \frac{2}{n} = c_l.$$

Now observe that $c_u > c_l$:

$$2 - \frac{2n^2(q-1) - n(6q-4) + 6q - 4}{(n^2 - 2n + 2)\,(n(q-1) - q)} > 2 - \frac{2}{n}. \tag{A.15}$$

Rearranging and simplifying (A.15) yields:

$$(n-2)n\left(n^2 - 2n + 2\right) q(n(q-1) - q) > 0,$$

which is always satisfied for any $n > q \geq 2$.

Therefore, if $c_l < c < c_u$, the optimal choice for the Defender is an intermediate sequence.

Now observe, that if $c_l < c < c_u$, the optimal intermediate sequence has $q = 2$. To see that observe the first derivative of U_I w.r.t. to q:

$$\frac{\partial U_I}{\partial q} = -\frac{(n-1)((c-2)n+2)}{2q^2},$$

which is always positive if $c > 2 - \frac{2}{n} = c_l$ and negative otherwise.

Since when $c > c_l$ it is always optimal for the Defender to choose a complete network, it must be the case that in the range of intermediate sequence optimality, the Defender must choose an intermediate sequence with $q = 2$. Then the condition (A.14) becomes:

$$2 - \frac{2n-4}{n^2 - 2n + 2}.$$

Lower family The Defender's expected payoff in a game in which she chooses a lower family sequence (3.19) is:

$$U_L = \left(1 - \frac{c}{2}\right)\left(\frac{n-1}{n-2} + 2n - 3\right) - 1.$$

The Defender prefers a lower family sequence to the star network when:

$$U_L > U_*,$$

which is satisfied when:

$$c < 2 - \frac{2n-3}{n^2 - 2n + 2} = c_L^1.$$

The Defender prefers a lower family sequence to an intermediate sequence with $q = 2$ when:

$$U_L > U_I,$$

which is satisfied when:

$$c > 2 - \frac{2n-3}{n^2-2n+2} = c_L^2.$$

Since $c_L^1 = c_L^2$, the Defender never prefers a lower family sequence to a star or an intermediate sequence. However, she is indifferent between star, lower family sequence, and intermediate sequence if $c = c_L^1 = c_L^2$.

***k*-family** An expected payoff in a game in which the Defender chooses a k-family sequence (3.23) is:

$$U_K = \left(1 - \frac{c}{2}\right)\left(\frac{k^2}{k-1} - k + n\right) - 1.$$

Observe that the Defender prefers a k-family sequence to a star sequence if:

$$U_K > U_*,$$

which satisfied IFF:

$$c > 2 + \frac{2(k-1)}{((n-2)n+2)(k(n-3)-n+2)} > 2,$$

which is always larger than 2 for any $n \geq 4$.

It implies that any sequence of the k-family cannot be a maximiser for the relaxed optimisation problem.

Quasi-star partial family The Defender's expected payoff from a game when she chooses a family (3.24) is:

$$U_{QS} = \left(1 - \frac{c}{2}\right)\left(\frac{(k-1)(n-k)}{\frac{(k-1)^2}{k} + \frac{1}{n-1}} + k + n - 1\right) - \frac{k-1}{\frac{(k-1)^2}{k} + \frac{1}{n-1}}.$$

First, observe that $U_{QS} > U_*$ IFF:

$$c < 2 - \frac{2(n-1)\left(k^2 + k(n-3)n + k - (n-1)^2\right)}{((n-2)n+2)\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)} = c_{qs}^1.$$

Now observe that $U_{QS} > U_C$ IFF:

$$c > 2 - \frac{2(n-1)\left((k-1)^2n^2 - 3(k-2)kn + (k-3)k - 2n + 1\right)}{n\left(k^2(n((n-4)n+5)-3) - 2k(n-2)(n-1)^2 + (n-1)^3\right)} = c_{qs}^2.$$

We now demonstrate that $c_{qs}^2 > c_{qs}^1$:

$$\begin{aligned} 2 - \frac{2(n-1)\left((k-1)^2n^2 - 3(k-2)kn + (k-3)k - 2n + 1\right)}{n\left(k^2(n((n-4)n+5)-3) - 2k(n-2)(n-1)^2 + (n-1)^3\right)} > \\ 2 - \frac{2(n-1)\left(k^2 + k(n-3)n + k - (n-1)^2\right)}{((n-2)n+2)\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)}, \end{aligned} \tag{A.16}$$

Rearranging and simplifying (A.16) yields:

$$\begin{aligned}&\underbrace{\left(k^2(n((n-4)n+5)-3)-2k(n-2)(n-1)^2+(n-1)^3\right)}_{\tau_1}\\&\underbrace{\left(k^2+k(n-2)(n-1)-(n-1)^2\right)}_{\tau_2}\underbrace{\left(k(k(n-1)-2n+3)+n-1\right)}_{\tau_3}\\&\underbrace{\left(k^2+k((n-4)n+2)-(n-1)^2\right)}_{\tau_4}>0\end{aligned} \tag{A.17}$$

We now consider first derivatives of τ_1, τ_2, τ_3, and τ_4 w.r.t. k.

τ_1:

$$\frac{\partial\tau_1}{\partial k}=2k(n((n-4)n+5)-3)-2(n-2)(n-1)^2. \tag{A.18}$$

To analyse the sign of (A.18) we now consider the second derivative of τ_1 w.r.t. k:

$$\frac{\partial^2\tau_1}{\partial k^2}=2(n((n-4)n+5)-3)>0, \tag{A.19}$$

which is always positive for any $n>3$. Now consider the sign of the first derivative (A.18) when $k=2$:

$$\frac{\partial\tau_1}{\partial k}=2n((n-4)n+5)-8>0, \tag{A.20}$$

which is always positive for any $n>3$.

τ_2 :

$$\frac{\partial\tau_2}{\partial k}=2k+(n-2)(n-1)>0,$$

which is always positive for any $n>k\geq 2$.

τ_3 :

$$\frac{\partial\tau_3}{\partial k}=2k(n-1)-2n+3>0.$$

which is always positive for any $n>k\geq 2$.

τ_4 :

$$\frac{\partial\tau_4}{\partial k}=2k+(n-4)n+2>0.$$

which is always positive for any $n>k\geq 2$.

Since $\frac{\partial\tau_j}{\partial k}>0$, $\forall j\in\{1,2,3,4\}$, it is sufficient to verify condition (A.17) when $k=2$. Substituting $k=2$ and simplifying (A.17) yields:

$$(n-2)(n-1)n(n+1)((n-6)n+7)((n-4)n+7)((n-2)n+2)(n((n-3)n+3)-5) > 0,$$

which is always positive for any $n \geq 5$.

It implies that sequence (3.24) cannot be a maximiser for the relaxed maximisation problem.

Lower partial family The Defender's expected payoff from a game when she chooses a sequence (3.25):

$$U_{LP} = \left(1 - \frac{c}{2}\right)\left(\frac{k(n-2)}{k-1} + \frac{k(n-1)}{k(n-2)-n+1} + n - 1\right) - \frac{k}{k-1}.$$

Observe that $U_{LP} > U_*$ IFF:

$$c < 2 - \frac{2\left(k + (n-1)^2\right)(k(n-2)-n+1)}{((n-2)n+2)\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)} = c^1_{LP}.$$

Similarly, $U_{LP} > U_I$ IFF:

$$c > 2 - \frac{2kn - 4k - 2n + 2}{k((n-2)n+2) - n^2 + n} = c^2_{LP}.$$

Now observe that $c^2_{LP} > c^1_{LP}$:

$$2 - \frac{2\left(k + (n-1)^2\right)(k(n-2)-n+1)}{((n-2)n+2)\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)} < 2 - \frac{2kn - 4k - 2n + 2}{k((n-2)n+2) - n^2 + n}.$$

Rearranging and simplifying yields:

$$\lambda\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)(k(n-2)-n+1)\left(k((n-2)n+2) - n^2 + n\right) > 0, \tag{A.21}$$

where $\lambda = (k-1)(n-2)(n-1)^2((n-2)n+2)(k(n-2)-n+1) > 0$.

Dividing both sides of (A.21) by λ yields:

$$\underbrace{\left(k^2 + k(n-2)(n-1) - (n-1)^2\right)}_{\iota_1}\underbrace{\left(k((n-2)n+2) - n^2 + n\right)}_{\iota_2} > 0.$$

Now consider first derivatives of ι_1 and ι_2 w.r.t. k:

$$\frac{\partial \iota_1}{\partial k} = 2k + (n-2)(n-1) > 0, \qquad \frac{\partial \iota_2}{\partial k} = (n-2)n + 2 > 0,$$

which are always positive.

Thus, it is sufficient to verify that the condition is satisfied when $q = 2$:

$$\left(n^2 - 4n + 7\right)\left(n^2 - 3n + 4\right) > 0,$$

which is always positive.

It follows that sequence (3.25) cannot be a maximiser either.

■

Proof of Claim 3.2. First consider family (3.19). The second largest element of this family always has the following form:

$$v_2 = \frac{n-1}{n-2} = 1 + \frac{1}{n-1},$$

which is not integer.

Now consider family (3.23). The first k elements of the *k-family* have the following form:

$$\frac{k}{k-1} = 1 + \frac{1}{k-1},$$

which is integer only when $k = 2$.

Thus, the only sequence which belongs to *k-family* and graphic is a degree sequence with $n = 4$ and $k = 2$[18]. However, this sequence is always dominated by a star network. To see then consider the Defender's expected utility functions from games on a star network (U_*) and on a *k-family* network (U_K), when $n = 4$ and $q = 2$:

$$U_* = 2(1 - \frac{c}{2})(n-1) - \frac{(n-1)^2}{(n-1)^2 + 1} = 51/10 - 3c,$$

$$U_K = \left(1 - \frac{c}{2}\right)\left(\frac{2k}{k-1} - k + n\right) - 1 = 5 - 3c.$$

Thus, $U_* > U_K$.

Now consider a quasi-star family (3.24). As was stated above any element which has value $\frac{k}{k-1}$ is integer IFF $k = 2$. Then elements indexed $j \in (k, n]$ have value:

$$v_k = \frac{k-1}{\frac{(k-1)^2}{k} + \frac{1}{n-1}} = \frac{2(n-1)}{n+1} = 2 - \frac{4}{n+1},$$

which is integer IFF $n = 3$.

[18] Any sequence with $n > 4$ does not pass the Erdos-Gallai constraint (3.5) and, therefore, is not graphic.

However, eqrefquasi-star family sequence with $n = 3$ elements and $k = 2$ is not graphic $(2, 2, 1)$.

The last sequence to consider is *lower partial family* sequence. However, this family exists only when $k \geq 3$, and therefore any element of value $\frac{k}{k-1}$ is not integer.

■

Proof of Lemma 3.3. The Defender's expected payoff at the beginning of the game on a star network:

$$U_* = 2\left(1 - \frac{c}{2}\right)(n-1) - \frac{(n-1)^2}{(n-1)^2 + 1},$$

and on a complete network:

$$U_C = n\left(1 - \frac{c}{2}\right)(n-1) - \frac{(n-1)^2}{n}.$$

An expected Defender's payoff from a game on a sequence of the intermediate family (3.18) is:

$$U_I = \left(1 - \frac{c}{2}\right)\left(\frac{(n-1)(q-1)(n-q)}{q} + (n-1)q\right) - \frac{(n-1)(q-1)}{q},$$

where $q \in [2, n-1]$.

An intermediate sequence is preferred to a star network if:

$$U_I > U_*,$$

which is satisfied IFF:

$$c < 2 - \frac{2n^2(q-1) - n(6q-4) + 6q - 4}{(n^2 - 2n + 2)(n(q-1) - q)} = c_u^q. \tag{A.22}$$

Similarly, the Defender prefers an intermediate sequence to a complete network when:

$$U_I > U_C,$$

which is satisfied when:

$$c > 2 - \frac{2}{n} = c_l.$$

Now observe that $c_u^q > c_l$:

$$2 - \frac{2n^2(q-1) - n(6q-4) + 6q - 4}{(n^2 - 2n + 2)(n(q-1) - q)} > 2 - \frac{2}{n}. \tag{A.23}$$

Rearranging and simplifying (A.23) yields:

$$(n-2)n\left(n^2-2n+2\right)q(n(q-1)-q)>0,$$

which is always satisfied for any $n>q\geq 2$.

Therefore, if $c_l<c<c_u^q$, the optimal choice for the Defender is an intermediate family sequence. ■

Proof of Lemma 3.4. Consider the first derivative of the objective function of maximisation problem (3.4) w.r.t. q:

$$\frac{\partial U_I}{\partial q}=-\frac{(n-1)(n(c-2)+2)}{2q^2}. \tag{A.24}$$

Observe that the sign of (A.24) does not depend on q itself, but rather on the cost of connection, c, relative to the overall number of vertices. Partial derivative $\frac{\partial U_I}{\partial q}$ is positive whenever $c<2-\frac{2}{n}$ and negative whenever $c>2-\frac{2}{n}$. The Defender is indifferent between any quantity of completely connected nodes if $c=2-\frac{2}{n}$. ■

Proof of Proposition 3.4. Consider expected payoffs of the Defender at the beginning of the game on a star network:

$$U_*=2\left(1-\frac{c}{2}\right)(n-1)-\frac{(n-1)^2}{(n-1)^2+1},$$

and on a complete network:

$$U_C=n\left(1-\frac{c}{2}\right)(n-1)-\frac{(n-1)^2}{n}.$$

A star network is preferred to a complete network if $U_*>U_C$, which is true IFF:

$$c>2-\frac{2\left(n^2-2+1\right)}{n\left(n^2-2n+2\right)}=c_i$$

■

Proof of Lemma 4.1. Suppose the Defender must choose between two network structures G_1 and G_2. Assume that the overall value of degree sequence G_1 is larger than the overall value of G_2: $V_1>V_2$. Then it must be the case that the expected payoff of the Defender in the encounter stage from the game on G_1 is weakly smaller than an expected payoff on the game G_2: $D_1 \leq D_2$. Now observe that the Defender is indifferent between the two structures IFF:

$$c_{12} = 2 - \frac{(1-\alpha)(D_2 - D_1)}{\alpha(V_1 - V_2)}. \tag{A.25}$$

Now consider the first derivative of the boundary (A.25) with respect to α:

$$\frac{\partial c_{12}}{\partial \alpha} = \frac{D_2 - D_1}{\alpha^2 (V_1 - V_2)} \geq 0.$$

Therefore, any cost boundary weakly increases in α. ∎

Proof of Claim 4.1. Consider weighted expected utility functions of the Defender from games on a star, complete, and m-maxi-core networks:

$$U_*^\alpha = 2\alpha \left(1 - \frac{c}{2}\right)(n-1) - \frac{(1-\alpha)(n-1)^2}{(n-1)^2+1}, \tag{A.26}$$

$$U_C^\alpha = \alpha \left(1 - \frac{c}{2}\right)(n-1)n - \frac{(1-\alpha)(n-1)^2}{n}, \tag{A.27}$$

$$U_{MC}^\alpha = \alpha \left(1 - \frac{c}{2}\right)\left(\frac{(n-1)(q-1)(n-q)}{q} + (n-1)q\right) - \frac{(1-\alpha)(n-1)(q-1)}{q}. \tag{A.28}$$

An m-maxi-core network is preferred to a star and complete network IFF:

$$\underbrace{2 + \frac{2(\alpha-1)}{\alpha n}}_{=c_l^\alpha} < c < \underbrace{2 + \frac{2(\alpha-1)(n(n(q-1)-3q+2)+3q-2)}{\alpha((n-2)n+2)(n(q-1)-q)}}_{=c_u^\alpha}.$$

Now consider the size of optimality interval $\delta_{MC} = c_u^\alpha - c_l^\alpha$:

$$\delta_{MC} = \frac{2(\alpha-1)(n-2)q}{\alpha n((n-2)n+2)(n(-q)+n+q)}.$$

Now consider the first derivative of δ_{MC} w.r.t α:

$$\frac{\partial \delta_{MC}}{\partial \alpha} = \frac{2(n-2)q}{\alpha^2 n((n-2)n+2)(n+q-nq)} < 0.$$

Since $(n+q-nq) < 0$, it follows that $\frac{\partial \delta_{MC}}{\partial \alpha}$ is always negative.

∎

B Supplementary material

Erdos-Gallai theorem is a known results from graph theory proposed by Erdős and Gallai (1960). The theorem solves the graph realisation problem, i.e. provides a set of necessary and sufficient conditions for a finite sequence of positive integer numbers to be the degree sequence of a graph. The sequence that satisfies the conditions is said to be "graphic". The original theorem demonstrated that a sequence of positive integers $d_1 \geq \cdots \geq d_n$ is graphic IFF $\sum_{i=1}^{n} d_i$ is even and:

$$\sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min(d_i, k)$$

holds for $\forall k \in [1, n]$.

Therefore, Erdos-Gallai conditions ensure that any given node does not have more links than can be supported by the degrees of all other nodes in the network.

C Extreme points characteristics and the convex hull

The non-convexity of set Υ is caused by non-convex constraints (3.12). Given that non-convex constraints are defined by smooth concave functions (by Lemma 2.1), we can construct a convex hull of set Υ, (Υ) by replacing them with linear constraints defined by hyperplanes spanned by extreme intersection points of non-convex constraints (3.12) and the surface of a hypercube defined by linear constraints (C.1) (Boyd & Vandenberghe, 2004). We denote extreme points defined by intersections of non-convex constraints (3.12) and linear constraints (3.11) as *extreme points of non-convex regions.*

This process is often referred to as set "convexification" (Li et al., 2001). However, the convexification process for complete and partial support cases is different. While in the partial support case, there are $n - k$ convex constraints (3.13), there are no convex constraints in the full support case. Thus, the set of extreme points of non-convex regions can differ. We consider full and partial support cases separately.

C.1 Full support

Consider set Υ^C defined by the following inequality constraints:

$$1 \leq v_i \leq n-1 \qquad \forall i \in [1,n], \tag{C.1}$$

$$\frac{n-2}{\sum_{j\in N\setminus\{i\}} \frac{1}{v_j}} - v_i \leq 0 \qquad \forall i \in [1,n], \tag{C.2}$$

To construct (Υ^C), we first characterise all of the extreme points of non-convex regions of the original set Υ^C. We denote the set of extreme points of Υ^C as $E(\Upsilon^C)$. There might be two types of extreme points:

Type 1 extreme points defined by the intersection of a single non-convex constraint (C.2) and the surface of a hypercube (C.1);

Type 2 extreme points defined by the intersection of $t \geq 2$ non-convex constraints (C.2) and the surface of a hypercube (C.1).

Type 1 extreme points. Consider a non-convex constraint for the value of some node $x \in N$:

$$v_x \geq \frac{n-2}{\sum_{j\in N\setminus\{x\}} \frac{1}{v_j}} = A_x, \tag{C.3}$$

where A_x is an expected gain from an attack on nodes indexed $j \in N \setminus \{x\}$. Non-convex constraint intersects the surface of a hyperplane (C.1) in two cases:

Case 1a. Node x takes values $v_x = A_x \in \{1, n-1\}$;

Case 2a. Nodes indexed $j \in N \setminus \{x\}$ take extreme values $v_j \in \{1, n-1\}$ and $v_x = A_x$.

Also observe that expected gain from an attack on nodes indexed $j \in N \setminus \{x\}$ does not depend on the order of values in set $N \setminus \{x\}$. It follows that for a given set of values that satisfy (C.3) there are $(n-1)!$ ways to write coordinates of some point in which coordinate v_x has a fixed value, while all other coordinates permute. We denote the set of all $(v_{\sigma(1)}, \cdots, v_{\sigma(n)})$ as σ ranges over permutations of $\{1, \cdots, n\}$ that fix coordinate x, as a *family of points*. To simplify the notation, we represent each family of points with a partially ordered set in which first $n-1$ elements v_j indexed $j \in N \setminus \{x\}$ are ordered from the largest to the smallest and the last element is v_x—we denote this partially ordered set as the *composition of coordinates*.

Case 1a: $v_x = A_x \in \{1, n-1\}$. First observe that any point which has coordinate $v_x = n-1$ cannot be an extreme point of non-convex region. This is because the maximum possible expected value from an attack on $n-1$ nodes is $\max_{v_j} A = \frac{(n-1)(n-2)}{n-1} = n-2 < n-1$, where $i \in N \setminus \{x\}$.

Claim C.1. *Any point which has coordinate $v_x = A_x = n-1$ is not an extreme point of non-convex region of Υ^C.*

Now consider some point which has coordinate $v_x = 1$. First observe that when $v_x = 1$, the nodes indexed $j \in N \setminus \{x\}$ cannot all simultaneously attain extreme values $v_j \in \{1, n-1\}$, i.e. $A_x = 1$.

Claim C.2. *If $v_j \in \{1, n-1\}$, $\forall j \in N \setminus \{x\}$, then $A_x \neq 1$.*

Proof of Claim C.2. Consider some point in which q coordinates have value $v^q = n-1$, and $n-q-1$ coordinates have value $v^u = 1$ and $v_x = A_x = 1$. Substituting the sequence with this composition of coordinates in (C.3) yields:

$$A_x = 1 = \frac{n-2}{\frac{q}{n-1} + n - q - 1}. \tag{C.4}$$

Solving (C.4) for q yields:

$$q = \frac{n-1}{n-2},$$

which is not feasible since q is the number of coordinates that attain value $v^q = n-1$, $q \in \mathbb{Z}$. ■

However, we can still find a point which has composition of coordinates, i.e. $v_x = 1$, $v_j \in \{1, n-1\}$, $\forall j \in N \setminus \{x, y\}$, and some coordinate $v_y \in [1, n-1]$. The composition of coordinates of the family mentioned above can be written as follows:

$$\mathbf{v}_b^q = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, v_y, \underbrace{1, \cdots, 1}_{\times(n-q-2)}, \underbrace{1}_{=A_x} \right), \tag{C.5}$$

where q is a number of nodes which attain value $n-1$.

We now demonstrate that the only case in which $\mathbf{v}_b^q \in E(\Upsilon^C)$ is when $q = 1$.

Claim C.3. $\mathbf{v}_b^q \in E(\Upsilon^C)$ *IFF: $q = 1$. In this case, $v_y = \frac{n-1}{n-2}$.*

Proof of Claim C.3. First, consider points which have composition of coordinates $\mathbf{v}_b^0$ in which $q = 0$:

$$\mathbf{v}_b^0 = \left(v_y, \underbrace{1, \cdots, 1}_{\times(n-2)}, \underbrace{1}_{=A_x} \right). \tag{C.6}$$

Substituting (C.6) into (C.3) yields:

$$A_x^0 = 1 = \frac{n-2}{(n-2) + \frac{1}{v_y}},$$

which does not have a solution for $v_y \in \mathbb{R}_+$.

Now consider some points which have composition of coordinates $\mathbf{v}_b^{q \geq 1}$. Substituting (C.5) in (C.3) yields:

$$A_x^{q \geq 1} = 1 = \frac{n-2}{\frac{q}{n-1} + (n-2-q) + \frac{1}{v_y}}. \tag{C.7}$$

Solving equation (C.7) for v_y yields:

$$v_y = \frac{n-1}{(n-2)q}. \tag{C.8}$$

Observe that the RHS of equation (C.8) is smaller than 1 for any $q \geq 2$. Therefore, any point with composition of coordinates $v_b^{\geq 2}$ lies beyond a hypercube defined by linear constraints (C.1) and $\mathbf{v}_b^{q \geq 2} \notin E(\Upsilon^C)$.

Thus, the only possibility in which $v_b^{q \geq 1} \in E(\Upsilon^C)$ is when $q = 1$. In this case:

$$v_y = \frac{n-1}{n-2} > 1.$$

$\blacksquare$

It follows from Claims C.1 and C.3 that the only family of extreme points that *Case 1a* yields must have the following composition of coordinates:

$$\mathbf{v}_b = \left(n-1, \frac{n-2}{n-1}, \underbrace{1, \cdots 1}_{\times(n-3)}, \underbrace{1}_{=A_x} \right). \tag{C.9}$$

As in sequence (C.9), coordinate v_x attains the lowest possible value $v_x = 1$, we denote $\mathbf{v}_b$ as the *lower family of extreme points.*

Case 2a. $v_j \in \{1, n-1\}, \forall j \in N \setminus \{x\}$**.** In this case, any point has the following composition of the coordinates:

$$\mathbf{v}_t^q = \left(\underbrace{n-1, \cdots n-1}_{q}, \underbrace{1, \cdots 1}_{n-q-1}, A_x \right), \tag{C.10}$$

where q is the number of nodes that attain value $n-1$.

The value A_x given the composition (C.10) can be expressed as:

$$A_x^q = \frac{(n-2)(n-1)}{(n-1)(n-q-1)+q}. \tag{C.11}$$

First, we demonstrate that families of extreme points (C.10) with $q = \{0, 1\}$ are not extreme points of set Υ^C.

Claim C.4. $\mathbf{v}_t^q \notin E(\Upsilon^C)$ *if* $q \in \{0, 1\}$.

Proof of Claim C.4. Substituting composition (C.10) in which $q = 0$, $\mathbf{v}_t^0$ into (C.11) yields:

$$A_x^0 = \frac{(n-2)(n-1)}{(n-1)(n-1)} = \frac{n-2}{n-1} < 1.$$

Therefore, points of family $\mathbf{v}_t^0 \notin E(\Upsilon^C)$.

Substituting composition (C.10) in which $q = 1$, $\mathbf{v}_t^1$ into (C.11) yields:

$$A_x^1 = \frac{(n-2)(n-1)}{(n-1)(n-2)+1} < 1.$$

Thus, points of family $\mathbf{v}_t^1 \notin E(\Upsilon^C)$ as well. ■

We now demonstrate that the only family of extreme points that is included in $E(\Upsilon^C)$ must have the composition of coordinates (C.10) with $q = n-1$.

Claim C.5. $\mathbf{v}_t^q \in E(\Upsilon^C)$ *IFF:* $q = n-1$.

Proof of Claim C.5. Consider families of points $\mathbf{v}_t^q$ with $q \in [2, n-2]$. Any of those families has at least one node of value $v^u = 1$. Now consider non-convex constraint (C.2) written for some node $l \in N \setminus \{x\}$ that attains value v^u:

$$v_l \geq \underbrace{\frac{(n-2)^2}{n^2 - n(q+3) + 2q + 3}}_{\iota}. \tag{C.12}$$

Consider the first derivative of ι w.r.t. q:

$$\frac{\partial \iota}{\partial q} = \frac{(n-2)^3}{\left(n^2 - n(q+3) + 2q + 3\right)^2} > 0.$$

Therefore, it is sufficient to rule-out (C.12) when $q = 2$:

$$v_l \geq \frac{(n-2)^2}{(n-5)n+7}. \tag{C.13}$$

Substituting $v_l = 1$ in (C.13) and simplifying yields:

$$n \leq 3.$$

Therefore, constraint (C.12) is never satisfied for any $n > 3$. It follows that $\mathbf{v}_t^q \notin E(\Upsilon^C),\ \forall q \in [2, n-2]$.

The only case left to verify is when $q = n-1$. Substituting $\mathbf{v}_t^{n-1}$ into (C.11) yields:

$$A_x^{n-1} = \frac{(n-2)(n-1)}{(n-1)(n-n+1-1)+n-1} = n-2.$$

It follows that $\mathbf{v}_t^{n-1} \in E(\Upsilon^C)$ as all nodes always satisfy all constraints. ■

Therefore the only family of extreme points which *Case 2a* yields must have the following composition of coordinates:

$$\mathbf{v}_t = \left(\underbrace{n-1, \cdots n-1}_{\times(n-1)}, \underbrace{n-2}_{=v_x} \right). \tag{C.14}$$

As in composition (C.14), coordinate $v_x = n-2$, we denote $\mathbf{v}_t$ as the *upper family of extreme points.*

Type 2 extreme points. This type of extreme points results from intersection of a surface of a hypercube (C.1) with $t \in [2, n]$ non-convex constraints (C.2).

We first characterise the function of intersection of $t \leq n$ non-convex constraints (C.2). We denote the set of nodes whose constraints intersect as $T \subseteq N$, $|T| = t$. Without loss of generality we relabel the nodes which belong to set T and write their values as a set $\mathbf{v}_z = (v_{z_1}, \cdots, v_{z_t})$. Set $\mathbf{v}_z$ is not necessarily ordered. The function of intersection must satisfy the following system of equations:

$$\begin{cases} v_{z_1} = \frac{n-2}{\sum_{j \in N \setminus \{z_1\}} \frac{1}{v_j}}, \\ \vdots \\ v_{z_t} = \frac{n-2}{\sum_{j \in N \setminus \{z_t\}} \frac{1}{v_j}}. \end{cases} \tag{C.15}$$

We now demonstrate that system (C.15) is satisfied IFF $v_{z_i} = \frac{n-1-t}{\sum_{j \in N \setminus T} \frac{1}{v_j}},\ \forall z_i \in T$.

Claim C.6. *System of equations* (C.15) *is satisfied IFF*

$$v_{z_i} = \frac{n-1-t}{\sum_{j \in N \setminus T} \frac{1}{v_j}},\ \forall z_i \in T. \tag{C.16}$$

Proof of Claim C.6. We prove by induction on the number of constraints that intersect, t.

Base case, $t = 2$. Consider the base case in which constraints for some nodes z_1 and z_2 intersect. The corresponding system of equations is then:

$$\mathrm{v}_{z_1} = \frac{n-2}{\frac{1}{z_2}+\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}},$$

$$v_{z_2} = \frac{n-2}{\frac{1}{z_1}+\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}.$$

Substituting (C.1) into (C.1) yields:

$$v_{z_1} = \frac{n-2}{\frac{\frac{1}{z_1}+\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}{n-2}+\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}.$$

Rearranging and simplifying yields:

$$v_{z_1} = \frac{n-3}{\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}. \tag{C.17}$$

Substituting (C.17) into (C.1), rearranging, and simplifying yields:

$$v_{z_2} = \frac{n-3}{\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}. \tag{C.18}$$

Thus, the system is satisfied whenever $v_{z_1} = v_{z_2} = \frac{n-3}{\sum_{j\in N\setminus\{z_1,z_2\}}\frac{1}{v_j}}$ and the claim holds.

Induction step. Assume that the statement holds for some $2 < w < n$. Now consider the case in which $w+1$ constraints intersect:

$$\begin{cases} v_{z_1} = \frac{n-2}{\sum_{j\in N\setminus\{z_1\}}\frac{1}{v_j}}, \\ \vdots \\ v_{z_{w+1}} = \frac{n-2}{\sum_{j\in N\setminus\{z_{w+1}\}}\frac{1}{v_j}}. \end{cases} \tag{C.19}$$

As we assumed that statement holds for w it must be the case that solving first w equations of the system (C.19) for $v_{z_i},\ i \in [1, w]$ yields:

$$v_{z_1} = v_{z_2} = \cdots = v_{z_w} = \frac{n-w-1}{\sum_{j\in N\setminus W}\frac{1}{v_j}}, \tag{C.20}$$

where $W \in \{z_1, \cdots, z_w\}$.

Substituting (C.20) into equation $w+1$ of system (C.19) yields:

$$v_{z_{w+1}} = \frac{n-2}{\frac{\frac{1}{v_{z_{w+1}}}+\sum_{j\in N\setminus W^1}\frac{1}{v_j}}{n-1-w}+\sum_{j\in N\setminus W^1}\frac{1}{v_j}}, \tag{C.21}$$

where $W^1 \in \{z_1, \cdots, z_{w+1}\}$. Solving (C.21) for $v_{z_{w+1}}$ yields:

$$v_{z_{w+1}} = \frac{n-w-2}{\sum_{j\in N\setminus W^1}\frac{1}{v_j}}. \tag{C.22}$$

We now substitute (C.22) into (C.20):

$$v_{z_i} = \frac{n-w-1}{\frac{\sum_{j\in N\setminus W^1}\frac{1}{v_j}}{n-w-2}+\sum_{j\in N\setminus W^1}\frac{1}{v_j}}. \tag{C.23}$$

Simplifying (C.24) yields:

$$v_{z_i} = \frac{n-w-2}{\sum_{j\in N\setminus W^1}\frac{1}{v_j}} = v_{z_{w+1}},\ \forall z_i \in W. \tag{C.24}$$

Thus, $v_{z_1} = \cdots = v_{z_{w+1}} = \frac{n-w-2}{\sum_{j\in N\setminus W^1}\frac{1}{v_j}}$. ■

It immediately follows from Claim C.6 that the maximum number of non-convex constraints that can intersect must be $t = n-2$, since otherwise system (C.15) does not have solutions in $\mathbb{R}^t_+$.

Claim C.7. *The maximum possible number of constraints that can intersect is $t = n-2$.*

Proof of Claim C.7. Observe that intersection function (C.16) is positive IFF:

$$n-1-t \geq 1,$$

which can be rearranged as:

$$t \leq n-2.$$

■

We can now characterise the extreme points that result from intersection of linear constraints (C.1) and t non-convex constraints (C.2). To simplify the notation we denote the set of nodes whose non-convex constraints are not intersected as $M = N\setminus T$. Without loss of generality we relabel nodes in set M and write the values of those nodes as a set $\mathbf{v}_x = (v_{x_1}, \cdots, v_{x_{n-t}})$. Set $\mathbf{v}_x$ is not necessarily ordered.

The composition of coordinates of each extreme point that results from intersection of t non-convex constraints and linear constraints then can be expressed as:

$$\mathbf{v}_i = \left(v_{x_1}, \cdots, v_{x_{n-t}}, v_{z_1}, \cdots, v_{z_t}\right), \tag{C.25}$$

where $x_i \in M$ and $z_i \in T$, and $v_{z_i} = \frac{n-1-t}{\sum_{j \in M} \frac{1}{v_j}}$

We now consider two cases:

Case 1b. Each node indexed z_i attains value $v_{z_i} \in \{n-1, 1\}$;

Case 2b. Each node indexed x_i attains value $v_{x_i} \in \{n-1, 1\}$.

Case 1b. $v_{z_i} \in \{1, n-1\}$. First observe that none of nodes indexed $z_i \in T$ can attain value $v_{z_i} = n-1$, as the maximum expected gain from at attack is equal to $A^c = \frac{(n-1)^2}{n} < n-1$.

Second, if nodes $z_i \in T$ each attain unit values $v_{z_i} = 1$, it yields the extreme point described by composition (C.9). Therefore, this case does not yield any new extreme points.

Case 2b. $v_{x_i} = \{1, n-1\}$. First, observe that nodes indexed $x_i \in M$ cannot all simultaneously attain unit values $v_{x_i} = v^u = 1$, as in this case $v_{z_i} < 1$, meaning that points like this lie beyond the hypercube (C.1).

Claim C.8. $\mathbf{v}_i \notin E(\Upsilon^C)$ *if* $v_{x_i} = 1$, $\forall x_i \in M$.

Proof of Claim C.8. Substituting $v_{x_i} = 1, \forall x_i \in M$ into (C.16) yields: $v_{z_i} = \frac{n-1-t}{n-t} < 1$ ■

Moreover, $\mathbf{v}_i \in E(\Upsilon^C)$ only when all nodes indexed $x_i \in M$ all attain maximum possible value $v_{x_i} = n-1$.

Claim C.9. $\mathbf{v}_i \in E(\Upsilon^C)$ *IFF* $v_{x_i} = n-1$, $\forall x_i \in M$.

Proof of Claim C.9. First, consider the case in which some node $x_b \in M$ attains value $v_b = n-1$, while the rest nodes x_j, $\forall j \in M \setminus b$ attain a unit value, $v_{x_j} = v^u = 1$. Then by Proposition 2.1, the expected attack on nodes $x_i \in M$ must be less than 1. Therefore, these points lie beyond the hypercube defined by (C.1).

Now consider the case in which $q \geq 2$ nodes indexed $x_i \in Q$ where $Q \subset M$ attain value $v_{x_i} = v^q = n-1$ while the rest $n-q-t$ nodes indexed $x_j \in M \setminus Q$ attain a unit value, $v_{x_j} = v^u = 1$. The corresponding composition of coordinates is then:

$$\mathbf{v}_i^{qu} = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{1, \cdots, 1}_{\times(n-q-t)}, v_{z_1}, \cdots, v_{z_t} \right),$$

where $z_i \in M$, and $v_{z_i} = \frac{n-1-t}{\sum_{j \in M} \frac{1}{v_j}}$.

However, composition $\mathbf{v}_i^{qu}$ violates the full support assumption as expected attack on $q \geq 2$ nodes of value $v^q = n-1$, $A^q = \frac{(n-1)(q-1)}{q} > 1$ for any $n > q \geq 2$.

Thus, $\mathbf{v}_i \in E(\Upsilon^C)$ IFF all nodes indexed $x_i \in M$ attain value $v_{x_i} = v^q = n-1$.

$\blacksquare$

Therefore, coordinates of each extreme point defined by the intersection of t non-convex constraints and a hypercube defined by linear constraints (C.1) can be written as follows:

$$\mathbf{v}_i^q = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{\frac{(n-1)(q-1)}{q} \cdots \frac{(n-1)(q-1)}{q}}_{\times(n-q)} \right). \tag{C.26}$$

By Claim C.7, any set Υ^C must have $n-2$ families of extreme points that have composition of coordinates (C.26). Each family differs in the number of coordinates $q \in [2, n-2]$. We denote the families of extreme points which have composition of coordinates (C.26) as *intermediate families of extreme points.*

Observe that intermediate families of extreme points (C.26) and the upper family of extreme points (C.14) can be described by a composition (C.26) allowing $q \in [2, n-1]$. Thus, we add the upper family of extreme points to the intermediate extreme points.

Therefore, set Υ^C has two classes of families of extreme points:

1. lower family of extreme points:

$$\mathbf{v_b} = \left(n-1, \frac{n-2}{n-1}, \underbrace{1, \cdots 1}_{\times(n-2)} \right). \tag{C.27}$$

2. intermediate families of extreme points:

$$\mathbf{v_i^q} = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{\frac{(n-1)(q-1)}{q} \cdots \frac{(n-1)(q-1)}{q}}_{\times(n-q)} \right), \tag{C.28}$$

where $q \in [2, n-1]$.

C.2 Convex hull construction for the full support case

In order to construct (Υ), it is sufficient to find a series of hyperplanes which span through all extreme points of non-convex regions and derive the corresponding constraints.

We first derive the form of the equation of a hyperplane that spans through all of extreme points of some family.

Claim C.10. *An equation of a hyperplane that spans through all of the extreme points of some family can be expressed as:*

$$\sum_{j \in N \setminus \{x\}} v_j + C v_x + D = 0,$$

where C and D are constants.

Proof of Claim C.10. Every extreme point from the same family has one fixed coordinate v_x, while all other coordinates $v_i \in N \setminus \{x\}$ permute. Let $P = N \setminus \{x\}$. Without loss of generality we relabel all coordinates $v_i \in P$, i.e. $i \in \{p_1, \cdots, p_{n-1}\}$. Then let S_P denote the set of all permutations of elements of P, i.e. each element $\sigma_j \in S_P$, $\sigma_j = \big(\sigma_j(v_1), \cdots, \sigma_j(v_{n-1})\big)$, where $j \in [1, (n-1)!]$.

If there exists a hyperplane which spans through all of the points of some family then the following system of equations must be satisfied for any $y \in [1, n-1]$ and $-y \in [1, n-1]$:

$\sum_{i=1}^{n-1} a_i \sigma_y(v_{p_i}) + a_x v_x + a_0 = 0,$

$\sum_{i=1}^{n-1} a_i \sigma_{-y}(v_{p_i}) + a_x v_x + a_0 = 0.$

Now assume that $a_1 = a_2 = \cdots = a_{n-1} = a$. Then, subtracting equation (C.2) from equation (C.2) and rearranging yields:

$$a \sum_{i=1}^{n-1} \sigma_y(v_{p_i}) + a_x v_x + a_0 = a \sum_{i=1}^{n-1} \sigma_{-y}(v_{p_i}) + a_x v_x + a_0. \tag{C.29}$$

By simplifying (C.29) we obtain:

$$\sum_{i=1}^{n-1} \sigma_y(v_i) = \sum_{i=1}^{n-1} \sigma_{-y}(v_i), \tag{C.30}$$

which always holds as the sum of elements of any permutation is always constant.

Therefore, the equation of a hyperplane that spans through points of one family can be expressed as follows:

$$a \sum_{j \in N \setminus \{x\}} v_j + a_x v_x + a_0 = 0. \tag{C.31}$$

Dividing both sides of (C.31) by a yields:

$$\sum_{j \in N \setminus \{x\}} v_j + \frac{a_x}{a} v_x + \frac{a_0}{a} = 0, \tag{C.32}$$

or

$$\sum_{j \in N \setminus \{x\}} v_j + C v_x + D = 0, \tag{C.33}$$

where $C = \frac{a_x}{a}$ and $D = \frac{a_0}{a}$. ■

We now demonstrate that extreme points of all intermediate families (C.28) lie on the same hyperplane.

Claim C.11. *There exists a hyperplane that spans through all extreme points of intermediate families. The corresponding hyperplane equation is:*

$$\sum_{j \in N \setminus \{x\}} v_j - (n-1) v_x - (n-1) = 0. \tag{C.34}$$

Proof of Claim C.11. Assume that there exists a hyperplane that spans through extreme points of two intermediate families with exactly q and $q+1$ coordinates attaining value $v^q = n-1$. Then the coefficients of corresponding hyperplane equation must satisfy the following system of equations:

$$\begin{cases} q(n-1) + (n-q-1)\frac{(n-1)(q-1)}{q} + \frac{(n-1)(q-1)}{q} C + D = 0, \\ (q+1)(n-1) + (n-q-2)\frac{(n-1)q}{q+1} + \frac{(n-1)q}{q+1} C + D = 0. \end{cases} \tag{C.35}$$

Solving system (C.35) for C and D yields:

$$C = -\frac{n^2 - 2n + 1}{n-1} = -(n-1), \qquad D = -(n-1).$$

Since neither C nor D depend on q, it is possible to build a hyperplane that spans through extreme points of all intermediate families simultaneously. The corresponding hyperplane equation is then:

$$\sum_{j \in N \setminus \{x\}} v_j - (n-1)v_x - (n-1) = 0.$$

■

However, extreme points of the lower family do not lie on the hyperplane (C.34). Thus, to cover all of the non-convex regions it is required to span one additional hyperplane. In the next claim we find a hyperplane which spans through lower family of nodes and the closest family of intermediate nodes[19].

Claim C.12. *The hyperplane which spans through extreme points of the lower family and the closest intermediate family of extreme points (with $q = 2$) can be described by the following hyperplane equation:*

$$\sum_{j \in N \setminus \{i\}} v_j - \frac{n^2 - 3n + 4}{n-2} v_i - \frac{(n-3)(n-1)}{n-2} = 0. \tag{C.36}$$

Proof of Claim C.12. To derive a corresponding hyperplane the following system of equations must be solved:

$$\begin{cases} 2(n-1) + (n-3)\frac{(n-1)}{2} + \frac{(n-1)}{2}C + D = 0, \\ (n-1) + \frac{n-1}{n-2} + (n-3) + C + D = 0. \end{cases} \tag{C.37}$$

Solving system (C.37) for C and D yields:

$$C = -\frac{n^2 - 3n + 4}{n-1} \qquad D = -\frac{(n-3)(n-1)}{n-2}.$$

The corresponding hyperplane equation is then:

$$\sum_{j \in N \setminus \{i\}} v_j - \frac{n^2 - 3n + 4}{n-2} v_i - \frac{(n-3)(n-1)}{n-2} = 0. \tag{C.38}$$

■

[19] The difference between coordinates of a 'lower' family extreme points and 'intermediate' extreme points is minimal when $q = 2$.

Therefore, in order to obtain (Υ^C), each non-convex constraint for some node $x \in [1, n]$ must be replaced with two linear constraints of the form:

$$\begin{cases} \sum_{j \in N \setminus x} v_j - (n-1)v_x - (n-1) \leq 0, \\ \sum_{j \in N \setminus x} v_j - \frac{n^2-3n+4}{n-1} v_x - \frac{(n-3)(n-1)}{n-2} \leq 0. \end{cases} \tag{C.39}$$

Thus, (Υ^C) is defined by the following set of constraints:

$$1 \leq v_i \leq n-1 \qquad \forall i \in [1, n], \tag{C.40}$$

$$\sum_{j \in N \setminus \{i\}} v_j - (n-1)v_i - (n-1) \leq 0 \qquad \forall i \in [1, n], \tag{C.41}$$

$$\sum_{j \in N \setminus \{i\}} v_j - \frac{n^2 - 3n + 4}{n-2} v_i - \frac{(n-3)(n-1)}{n-2} \leq 0 \qquad \forall i \in [1, n]. \tag{C.42}$$

We demonstrate the construction of a convex hull for the full support case with the example in which $n = 4$.

Example, $n = 4$. Consider the original set Υ^4 for the full support maximisation problem with $n = 4$:

$$\begin{aligned} &1 \leq v_i \leq 3 && \forall i \in [1, 4], \\ &\frac{2}{\sum_{j \in N \setminus \{i\}} \frac{1}{v_j}} - v_i \leq 0 && \forall i \in [1, 4]. \end{aligned}$$

Then, (Υ^4) is defined by the following constraints:

$$\begin{aligned} &1 \leq v_i \leq 3 && \forall i \in [1, 4], \\ &\sum_{j \in N \setminus \{i\}} v_j - 3v_i - 3 \leq 0 && \forall i \in [1, 4], \\ &\sum_{j \in N \setminus \{i\}} v_j - 4v_i - \frac{3}{2} \leq 0 && \forall i \in [1, 4]. \end{aligned}$$

Figure 4 in the main text illustrates the set of feasible values for v_i, where $i \in [1, 3]$ along the edge v_4.

C.3 Partial support

In this subsection we characterise the set of all extreme points of non-convex regions for the partial support case.

Let Υ^P be the set of feasible values for nodes $i \in N$ for the partial support case. Set Υ^P is defined by the following constraints:

$$1 \leq v_i \leq n-1 \qquad \forall i \in [1,k], \tag{C.43}$$

$$\frac{k-2}{\sum_{j \in K \setminus \{i\}} \frac{1}{v_j}} - v_i \leq 0 \qquad \forall i \in [1,k], \tag{C.44}$$

$$-\frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}} - v_j \leq 0 \qquad \forall j \in (k,n]. \tag{C.45}$$

This set of constraints can be significantly simplified. First, by Lemma 3.1 constraints (C.45) can be written with equality sign:

$$-\frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}} = v_j \quad \forall j \in (k,n].$$

Thus, every node outside set K attains a value that equals exactly the expected gain from an attack on nodes in set K.

We can now reduce the problem's dimensionality by ensuring that the gain from an attack on top k nodes is always larger than 1 (since any node outside the Attacker's support cannot attain a value smaller than 1). Moreover, nodes inside the Attacker's support cannot attain a value of 1 as well, as, in this case, all the nodes must then included in the Attacker's support violating the partial support assumption. To guarantee that all the nodes inside the Attacker's support are larger than 1 and all the nodes outside the Attacker's support are not smaller than 1, the lower boundary of constraint (C.43) must be increased. Since the expected attack on top k nodes attains the smallest value when all of the nodes attain the minimum possible value v_{min}, in order to find a new lower boundary for (C.43), it is sufficient to solve the following equation:

$$\frac{k-1}{\frac{k}{v_{min}}} = 1. \tag{C.46}$$

Solving equation (C.46) for v_{min} yields:

$$v_{min} = \frac{k}{k-1}.$$

Therefore, if $\frac{k}{k-1} \geq v_i$, $\forall i \in K$, all of the constraints for nodes v_j, $j \in N \setminus K$ are always satisfied. The constraints that define set Υ^P then can be restated as follows:

$$\frac{k}{k-1} \leq v_i \leq n-1 \qquad \forall i \in [1,k], \tag{C.47}$$

$$\frac{k-2}{\sum_{j\in K\setminus\{i\}} \frac{1}{v_j}} - v_i \leq 0 \qquad \forall i \in [1,k], \tag{C.48}$$

$$-\frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}} = v_j \qquad \forall j \in (k,n]. \tag{C.49}$$

It follows that similarly to the full support case, all non-convex regions of set Υ^P have extreme points defined by the intersection of one or several non-convex constraints (C.48) and the surface of hypercube defined by (C.47). We denote the set of extreme points of Υ^P as $E(\Upsilon^P)$. The set of nodes which pass the Attacker's support conditions with equality is denoted as Θ, $\Theta = N \setminus K$.

Before approaching the search of extreme points for the general partial support case, we first consider a special case in which $k = 2$.

Special case, k=2. Consider non-convex constraint (C.48) for nodes $i \in K = \{1, 2\}$:

$$v_i \geq \frac{1}{\frac{1}{v_i} + \frac{1}{v_{-i}}}.$$

Rearranging and simplifying yields:

$$v_i \leq v_i + v_{-i},$$

which is always satisfied.

Therefore, if $k = 2$, non-convex constraints do not influence the set of feasible values $\Upsilon^P_{k=2}$ and this set is always convex. It follows that all extreme points in this case are defined by linear constraints (C.47). In the next claim we derive the general form of all extreme points for the special case $k = 2$.

Claim C.13. *If $k = 2$, the set of feasible values for $i \in K = \{1, 2\}$ has three families of extreme points:*

$$\mathbf{w^1_{k=2}} = \left(2, 2, \underbrace{1, \cdots, 1}_{\times\theta}\right), \tag{C.50}$$

$$\mathbf{w^2_{k=2}} = \left(n-1, n-1, \underbrace{\frac{n-1}{2}, \cdots, \frac{n-1}{2}}_{\times\theta}\right), \tag{C.51}$$

$$\mathbf{w^3_{k=2}} = \left(n-1, 2, \underbrace{\frac{2(n-1)}{n+1}, \cdots, \frac{2(n-1)}{n+1}}_{\times\theta} \right). \tag{C.52}$$

where $\theta = |\Theta|$.

Proof of Claim C.13. First, observe that if $k = 2$, the minimum value that any node $i \in K$ can attain is $v_{min} = \frac{k}{k-1} = \frac{2}{2-1} = 2$.

We now consider three possibilities in which $v_i \in \{2, n-1\}$

1. $v_i = v_{-i} = 2$**,** $i \in \{1, 2\}$**.** Substituting $k = 2$, $v_i = v_{-i} = 2$ into (C.49) yields:

$$v_j = \frac{1}{\frac{1}{2} + \frac{1}{2}} = 2 \quad \forall j \in \Theta.$$

Therefore, this case yields the extreme point with composition of coordinates (C.50).

2. $v_i = v_{-i} = n-1$**,** $i \in \{1, 2\}$**.** Substituting $k = 2$, $v_i = v_{-i} = n-1$ into (C.49) yields:

$$v_j = \frac{n-1}{2} \quad \forall j \in \Theta.$$

Therefore, this case yields the extreme point with composition of coordinates (C.51).

3. $v_i = n-1$**,** $v_{-i} = 2$**,** $i \in \{1, 2\}$**.** Substituting $k = 2$, $v_i = n-1$ and $v_{-i} = n-1$ into (C.49) yields:

$$v_j = \frac{2(n-1)}{n+1} \quad \forall j \in \Theta.$$

Therefore, this case yields the extreme point with composition of coordinates (C.52). ■

From this point we focus on a more general case in which $k \geq 3$. As in the full support case, we distinguish two types of extreme points of non-convex regions: *Type 1 extreme points* which result from intersection of a single non-convex constraint and linear constraints; and *Type 2 extreme points* which result from intersection of several non-convex constraints and linear constraints.

Type 1 extreme points. We demonstrate that similarly to full support case, there are two families of *Type 1 extreme points* that belong to set $E(\Upsilon^P)$: lower family and upper family.

First, consider a non convex constraint (C.48) for the value of some node $x \in K$:

$$v_x \geq \frac{k-2}{\sum_{j \in K \setminus \{x\}} \frac{1}{v_j}} = A_x. \tag{C.53}$$

There are two cases to consider:

Case 1c. Intersections in which node x takes extreme values $v_x \in \{\frac{k}{k-1}, n-1\}$,

Case 2c. Intersection in which nodes indexed $i \in K \setminus x$ take values $v_i \in \{1, n-1\}$.

Case 1c. $v_x = A_x \in \{1, n-1\}$. We have already demonstrated in Claim C.1 that coordinate v_x cannot attain value $n-1$ in any extreme point of a non-convex region.

We now search for an extreme point in which coordinate $v_x = A_x = \frac{k}{k-1}$, some coordinate $v_y \in [\frac{k}{k-1}, n-1]$ and the rest coordinates $v_i \in \{\frac{k}{k-1}, n-1\}$, $\forall i \in K \setminus \{x, y\}$. The corresponding composition of coordinates can be written in the following form:

$$\mathbf{w}_b^q = \left(\underbrace{n-1}_{\times q}, \underbrace{\frac{k}{k-1}}_{\times k-q-2}, v_y, \underbrace{\frac{k}{k-1}}_{=A_x}, \underbrace{A^k}_{\times \theta} \right), \tag{C.54}$$

where q is a number of nodes that attain value $v^q = n-1$, A^k is an expected attack on top k nodes, and $\theta = |\Theta|$. Also note that in any sequence (C.54) it must be the case that $q \in [1, k-2]$.

We now demonstrate that the only case in which $\mathbf{w}_b^q \in E(\Upsilon^P)$ is when $q = 1$ and $v_y = \frac{k(n-1)}{kn-2k-n+1}$.

Claim C.14. $\mathbf{w}_b^q \in E(\Upsilon^P)$ *IFF* $q = 1$ *and* $v_y = \frac{k(n-1)}{kn-2k-n+1}$

Proof of Claim C.14. First, we demonstrate that if $q = 0$, $\mathbf{w}_b^0 \notin E(\Upsilon^P)$. Substituting sequence (C.54) with $q = 0$ into (C.53) yields:

$$A_x = \frac{k}{k-1} = \frac{k-2}{\frac{k-2}{\frac{k}{k-1}} + \frac{1}{v_y}}. \tag{C.55}$$

By rearranging and simplifying equation (C.55) we obtain:

$$\frac{k}{(k-1)\left(k^2 v_y - 3k v_y + k + 2\text{vy}\right)} = 0,$$

which does not have a solution for $v_y \in \mathbb{R}$.

Now consider the case in which $q \geq 1$. Substituting composition (C.54) in (C.53):

$$A_x = \frac{k}{k-1} = \frac{k-2}{\frac{q}{n-1} + \frac{k-q-2}{\frac{k}{k-1}} + \frac{1}{v_y}}. \tag{C.56}$$

Solving (C.56) for v_y yields:

$$v_y = \frac{k(n-1)}{q(kn-2k-n+1)}. \tag{C.57}$$

We now demonstrate that in (C.57) $v_y < \frac{k}{k-1}$ for any $q \geq 2$.

Consider the first derivative of the RHS of equation (C.57) w.r.t. q:

$$\frac{\partial}{\partial q}\frac{k(n-1)}{q(kn-2k-n+1)} = -\frac{k(n-1)}{q^2(kn-2k-n+1)} < 0,$$

which is always negative as $n(k-1) > 2k$. Therefore, it is sufficient to consider (C.57) when $q = 2$. Substituting $q = 2$ in (C.57) yields:

$$v_y = \frac{k(n-1)}{2k(n-2)-2n+2}. \tag{C.58}$$

We now show that:

$$\frac{k(n-1)}{2k(n-2)-2n+2} < \frac{k}{k-2}. \tag{C.59}$$

Rearranging and simplifying (C.59) yields:

$$(k-1)k(k(n-3)-n+1)(k(n-2)-n+1) > 0,$$

or:

$$\underbrace{(n(k-1)-3k+1)}_{\iota_1}(n(k-1)-2k+1) > 0.$$

Observe that $\frac{\partial \iota}{\partial k} = n-3 > 0$ for any $n > 3$. Thus, consider ι when $k = 4$ (the minimum value that k can attain in this composition is 4, as the composition has two nodes of value v^q, one node of value v^q and one node of value $\frac{k}{k-1}$):

$$\iota = 3n - 12 + 1 > 0,$$

which is always positive since $n > k = 4$.

It follows that inequality (C.59) is always satisfied and the intersection happens beyond the hypercube defined by linear constraints (C.47) for any $q \geq 2$. Thus, any family of points with composition $\mathbf{w}_b^{q\geq 2} \notin E(\Upsilon^P)$.

The only possibility that is left to verify is when $q = 1$. Substituting sequence (C.54) with $q = 1$ into (C.57) yields:

$$v_y = \frac{k(n-1)}{kn - 2k - n + 1}. \tag{C.60}$$

We now demonstrate that:

$$\frac{k(n-1)}{kn - 2k - n + 1} > \frac{k}{k-1}. \tag{C.61}$$

Rearranging and simplifying (C.61) yields:

$$(k-1)k^2(k(n-2) - n + 1) > 0,$$

or

$$k(n-2) - n + 1 > 0,$$

which is always satisfied for any $n > k \geq 3$. ■

Therefore, *Case 1c* yields only one family of extreme points with the following composition of coordinates:

$$\mathbf{w}_b^q = \left(n-1, \frac{k(n-1)}{kn - 2k - n + 1}, \underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{\times n-2} \right). \tag{C.62}$$

We denote the family of extreme points with composition (C.62) as the *the lower family of extreme points of the partial support case.*

Case 2c. $v_i \in \{\frac{k}{k-1}, n-1\}$, $\forall i \in K \setminus \{x\}$. The general composition of coordinates for this case can be written as:

$$\mathbf{w}_u^q = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{\times k-q-2}, A_x, \underbrace{A^k, \cdots, A^k}_{\theta} \right), \tag{C.63}$$

We demonstrate that *Case 2c* yields only one extreme point family with composition of coordinates $\mathbf{w}_u^q$ in which $q = k - 1$.

Claim C.15. $\mathbf{w}_u^q \in E(\Upsilon^P)$ *IFF* $q = k - 1$.

Proof of Claim C.15. We first demonstrate that $\mathbf{w}_u^q \notin E(\Upsilon^P)$ if $q \in \{0, 1\}$.

Substituting composition (C.63) with $q = 0$ in (C.53) yields:

$$A_x = \frac{k-2}{\frac{(k-1)^2}{k}} = \frac{(k-2)k}{(k-1)^2} = 1 - \frac{1}{(k-1)^2} < \frac{k}{k-1}.$$

It follows that this intersection happens beyond the hypercube defined by (C.47) and $\mathbf{w}_u^0 \notin E(\Upsilon)$.

Now consider the case in which $q = 1$. Substituting composition (C.63) with $q = 1$ in (C.53) yields:

$$A_x = \frac{k-2}{\frac{(k-2)(k-1)}{k} + \frac{1}{n-1}}.$$

We now demonstrate that:

$$\frac{k-2}{\frac{(k-2)(k-1)}{k} + \frac{1}{n-1}} < \frac{k}{k-1}. \tag{C.64}$$

Rearranging and simplifying (C.65) yields:

$$(k-1)k^2\left(k^2(n-1) + k(4-3n) + 2(n-1)\right) > 0,$$

or:

$$\underbrace{k^2(n-1) + k(4-3n) + 2(n-1)}_{\tau} > 0. \tag{C.65}$$

Consider the first derivative of τ w.r.t to k:

$$\frac{\partial \tau}{\partial k} = 4 - 2k - 3n + 2kn > 0,$$

which is always positive since $2n(k-3) \geq 2k$ for any $k \geq 3$. Thus, it is sufficient to verify inequality (C.65) at $k = 3$. Substituting $k = 3$ into (C.65) yields:

$$2n + 1 > 0,$$

which always holds.

It follows that $\mathbf{w}_u^1 \notin E(\Upsilon^P)$.

Now we demonstrate that any family with composition (C.63) with $q \in [2, k-2]$ and $k \geq 4$ cannot belong to $E(\Upsilon^P)$. To see that, consider the gain from an expected attack on $q \geq 2$ nodes:

$$A^q = \frac{(n-1)(q-1)}{q}.$$

Since $\frac{\partial A^q}{\partial q} = \frac{n-1}{q^2}$, it follows that the minimum value of A^q is achieved whenever $q = 2$:

$$A^2 = \frac{n-1}{2}.$$

Now observe that:

$$\frac{n-1}{2} > \frac{k}{k-1},$$

for any $n > k \geq 4$.

It follows that in any composition (C.63) with $q \in [2, k-2]$ nodes of value $v^q = n-1$, and $k-q-1$ nodes of value $v^k = \frac{k}{k-1}$ with $n > k \geq 4$, any node of value v^k is not included in the Attacker support. Thus, extreme points of this composition do not belong to Υ^P.

The only possibility left to verify is when $q = k-1$. In this case, there is no nodes of value v^k and every node of value $n-1$ must be included in the Attacker's support. Substituting composition (C.63) in (C.53) with $q = n-1$ yields:

$$A_x = \frac{(n-1)(q-1)}{q} > \frac{k}{k-1}.$$

■

It immediately follows from Claim (C.15) that the only family of extreme points that *Case 2c* yields is:

$$\mathbf{w}_u = \left(\underbrace{n-1, \cdots, n-1}_{\times k-1}, \underbrace{\frac{(n-1)(q-1)}{q}, \cdots, \frac{(n-1)(q-1)}{q}}_{\times n+1-k} \right). \tag{C.66}$$

We denote the family which have composition of coordinates (C.66) as the *upper family of extreme points of the partial support case.*

Type 2 extreme points. This type of extreme points results from the intersection of a surface of a hypercube defined by (C.47) and $t \geq 2$ non-convex constraints (C.48).

Utilising the results of Claim (C.6) we can write down the general form of intersection of t constraints straight away. Let T denote the set of nodes whose constraints intersect and $M = K \setminus T$ denote the set of nodes whose constraints do not intersect. Without loss of generality we relabel nodes in set T as $\mathbf{v}_z = (v_{z_1}, \cdots, v_{z_t})$ and nodes in set M as $\mathbf{v}_z = (v_{x_1}, \cdots, v_{z_{k-t}})$. Then any *Type 2* extreme point family can be written as:

$$\mathbf{w}_i = \left(v_{x_1}, \cdots, v_{x_{k-t}}, v_{z_1}, \cdots, v_{z_t}, \underbrace{A^k}_{\times \theta} \right), \tag{C.67}$$

where $v_{z_i} = \frac{k-1-t}{\sum_{j \in M} \frac{1}{v_j}}$.

As previously, there are two cases to consider:

Case 1d. Each node indexed $i \in T$ attains value $v_i \in \left\{\frac{k}{k-1}, n-1\right\}$;

Case 2d. Each node indexed $j \in M$ attains value $v_j \in \left\{\frac{k}{k-1}, n-1\right\}$.

Case 1d. $v_i \in \left\{\frac{k}{k-1}, n-1\right\}$**,** $\forall i \in T$**.** First, note that similarly to the full support case, none of nodes indexed $i \in T$ can attain value $v_{z_i} = v^q = n-1$, since the maximum gain from an attack is $\frac{(n-1)^2}{n} < n-1$.

Second, if nodes $v_i = \frac{k}{k-1}$, $\forall i \in T$ attain value $v_i = v^k = \frac{k}{k-1}$, then it yields intersection point which has composition (C.62).Therefore, this case does not yield any new extreme points.

Case 2d. $v_j \in \left\{\frac{k}{k-1}, n-1\right\}$**,** $\forall j \in M$**.** We demonstrate that $\mathbf{w_i} \in E(\Upsilon^P)$ only if all nodes indexed $j \in M$ take value $v_j = v^q = n-1$.

Claim C.16. $\mathbf{w_i} \in E(\Upsilon^P)$ *IFF* $v_j = v^q = n-1$, $\forall j \in M$.

Proof of Claim C.16. First, consider the case in which all the nodes $j \in M$ take value $v_j = v^k = \frac{k}{k-1}$. In this case any node indexed $i \in M$ must attain value:

$$v_{z_i} = \frac{k-1-t}{\frac{k-t}{\frac{k}{k-1}}} = \frac{k(k-t-1)}{(k-1)(k-t)}.$$

Now observe that:

$$\frac{k-1-t}{\frac{k-t}{\frac{k}{k-1}}} = \frac{k(k-t-1)}{(k-1)(k-t)} < \frac{k}{k-1}. \tag{C.68}$$

Simplifying (C.68) yields:

$$\frac{k-t-1}{k-t} < 1,$$

which is always satisfied for any $t \geq 2$.

Second, consider the case in which $q = 1$. In this case, any node indexed $i \in M$ must attain value:

$$v_{z_i} = \frac{k-t-1}{\frac{(k-1)(k-t-1)}{k} + \frac{1}{n-1}}.$$

Similarly observe that:

$$\frac{k-t-1}{\frac{(k-1)(k-t-1)}{k} + \frac{1}{n-1}} < \frac{k}{k-1},$$

which can be rearranged and simplified as

$$\underbrace{k^2(n-1) + k(-n(t+2)+t+3) + (n-1)(t+1)}_{\psi} > 0. \tag{C.69}$$

Consider the first derivative of ψ w.r.t to t:

$$\frac{\partial \psi}{\partial t} = k - 1 - (k-1)n < 0,$$

which is always negative.

Now consider condition (C.69) when t attains its maximum value $t = k - 2$:

$$(k-1)n + 1 > 0,$$

which is always satisfied.

Therefore, the case in which $q = 1$ does not yield any extreme points.

Third, consider the case in which $q \in [2, k-t)$ and $k > q$. In this case, any node that attains value $v^k = \frac{k}{k-1}$ cannot be included in the Attacker's support as $\frac{(n-1)(q-1)}{q} > \frac{k}{k-1}$ for any $k \geq 4$.

Therefore, the only feasible extreme point has the composition of coordinates in which $v_j = v^q = n - 1$, $\forall j \in M$ since, in this case, all the nodes $i \in M$ are included in the Attacker's support and:

$$v_{z_i} = \frac{(n-1)(q-1)}{q} > \frac{k}{k-1},$$

for any $k \geq 4$. ■

It follows from Claim C.16 that the only family of extreme points that *Case 2d* yields must have the following composition of coordinates:

$$\mathbf{w}_i^q = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{\frac{(n-1)(q-1)}{q}, \cdots, \frac{(n-1)(q-1)}{q}}_{\times (n-q)} \right), \qquad \text{(C.70)}$$

where $q \in [2, k-2]$.

We denote extreme points which have composition of coordinates (C.70) as the *intermediate family of extreme points of the partial support case.* Also observe that by allowing $q \in [2, k-1]$ and $k \in [2, n-1]$, the composition (C.70) also covers the upper family of extreme points of the partial support case (C.66) and the family (C.51). We combine those three types of families.

Therefore set Υ^P with $k \geq 3$ has two families of extreme points:

1. lower family of extreme points of the partial support case:

$$\mathbf{w}_b^q = \left(n-1, \frac{k(n-1)}{kn-2k-n+1}, \underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{\times(n-2)} \right). \tag{C.71}$$

2. intermediate family of extreme points of the partial support case:

$$\mathbf{w}_i^q = \left(\underbrace{n-1, \cdots, n-1}_{\times q}, \underbrace{\frac{(n-1)(q-1)}{q}, \cdots, \frac{(n-1)(q-1)}{q}}_{\times(n-q)} \right), \tag{C.72}$$

where $q \in [2, k-1]$.

Note that any point of family (C.72) can be characterised by the composition (C.28) found in the full support case.

C.4 Convex hull construction for the partial support case

As in the full support case, in order to construct a convex hull of Υ^P, it is sufficient to build a series of hyperplanes spanned by all the extreme points of non-convex regions and derive the corresponding constraints.

From Claim C.11 we know that hyperplane that spans through all extreme points of some family can be expressed in the following form:

$$\sum_{j \in K \setminus x} v_j + C v_x + D = 0,$$

where C and D are coefficients.

We now demonstrate that all the extreme points of the intermediate families of extreme points lie on the same hyperplane.

Claim C.17. *There exists a hyperplane that spans through all extreme points of all intermediate families. The corresponding hyperplane equation is:*

$$\sum_{j \in K \setminus x} v_j - (k-1)v_x - (n-1) = 0.$$

Proof of Claim C.17. Assume that there exists a hyperplane which spans through extreme points of two intermediate families which have exactly q and $q+1$ nodes of value $v^q = n-1$ correspondingly. Then the corresponding hyperplane equation must satisfy the following system of equations:

$$\begin{cases} q(n-1) + (k-q-1)\frac{(n-1)(q-1)}{q} + \frac{(n-1)(q-1)}{q}C + D = 0, \\ (q+1)(n-1) + (k-q-2)\frac{(n-1)q}{q+1} + \frac{(n-1)q}{q+1}C + D = 0. \end{cases} \tag{C.73}$$

Solving system (C.73) for C and D yields:

$$C = -(k-1), \qquad D = -(n-1).$$

Since neither C nor D depend on q, it is possible to build a hyperplane that spans through extreme points of all intermediate families simultaneously. The corresponding hyperplane equation is then:

$$\sum_{j \in N \setminus x} v_j - (k-1)v_x - (n-1) = 0.$$

■

We now characterise a hyperplane which spans through nodes of the lower family of extreme points and the closest to that family extreme points of the intermediate family (which have the composition of coordinates $\mathbf{w}_i^2$).

Claim C.18. *The hyperplane which spans through extreme points of the lower family and the closest intermediate family of extreme points ($\mathbf{w}_i^2$) can be described by the following hyperplane equation:*

$$\sum_{j \in K \setminus i} v_j - \frac{k^2(n-2) + k(5-2n) + n - 1}{k(n-2) - n + 1} v_i - \frac{(n-1)(kn - 3k - n + 1)}{kn - 2k - n + 1} = 0. \tag{C.74}$$

Proof of Claim C.18. To derive a corresponding hyperplane the following system of equations must be solved:

$$\begin{cases} 2(n-1) + (k-3)\frac{(n-1)}{2} + \frac{(n-1)}{2}C + D = 0, \\ (n-1) + \frac{k(n-1)}{kn-2k-n+1} + (k-3)\frac{k}{k-1} + \frac{k}{k-1}C + D = 0. \end{cases} \tag{C.75}$$

Solving system (C.75) for C and D yields:

$$C = -\frac{k^2(n-2) + k(5-2n) + n - 1}{k(n-2) - n + 1}, \qquad D = -\frac{(n-1)(kn - 3k - n + 1)}{kn - 2k - n + 1}.$$

The corresponding hyperplane equation is then:

$$\sum_{j \in K \setminus i} v_j - \frac{k^2(n-2) + k(5-2n) + n - 1}{k(n-2) - n + 1} v_i - \frac{(n-1)(kn - 3k - n + 1)}{kn - 2k - n + 1} = 0. \quad \text{(C.76)}$$

■

Therefore, in order to obtain (Υ^P) each non-convex constraint $x \in [1, k]$ must be replaces with two linear constraints of the form:

$$\begin{cases} \sum_{j \in K \setminus \{x\}} v_j - (k-1)v_x - (n-1) \leq 0, \\ \sum_{j \in K \setminus \{x\}} v_j - \frac{k^2(n-2)+k(5-2n)+n-1}{k(n-2)-n+1} v_x - \frac{(n-1)(kn-3k-n+1)}{kn-2k-n+1} \leq 0. \end{cases}$$

Thus, (Υ^P) is defined by the following set of constraints:

$$\begin{aligned} &\frac{k}{k-1} \leq v_i \leq n-1 && \forall i \in [1, k], \\ &\sum_{j \in K \setminus \{i\}} v_j - (k-1)v_i - (n-1) \leq 0 && \forall i \in [1, k], \\ &\sum_{j \in K \setminus \{i\}} v_j + Cv_i + D \leq 0 && \forall i \in [1, k], \\ &-\frac{(k-1)}{\sum_{i=1}^{k} \frac{1}{v_i}} = v_j && \forall j \in (k, n], \end{aligned}$$

where $C = -\frac{k^2(n-2)+k(5-2n)+n-1}{k(n-2)-n+1}$ and $D = -\frac{(n-1)(kn-3k-n+1)}{kn-2k-n+1}$.

C.5 Extreme points of convex regions

Apart from extreme points of non-convex regions, sets Υ^C and Υ^P have extreme points defined solely by linear constraints (C.1) and (C.47). In this subsection, we characterise these points. Full and partial support cases are considered separately.

Full support Any extreme points of a hypercube defined by linear constraints (C.1) can have coordinates of value either $v^q = n - 1$ or $v^u = 1$. Any family of those points must have the following composition:

$$\mathbf{v}_f^q = \left(\underbrace{n-1, \cdots, n-1}_{q}, \underbrace{1, \cdots, 1}_{n-q} \right), \tag{C.77}$$

where $q \in [0, n]$.

We now demonstrate that only families in which $q = 0$ (sequence of unit values), $q = 1$ (star network sequence), and $q = n$ (complete network sequence) are included in $E(\Upsilon^C)$.

Claim C.19. $\mathbf{v}_f^q \in E(\Upsilon^C)$ *IFF* $q \in \{0, 1, n\}$.

Proof of Claim C.19. First consider composition (C.77) with $q = 0$. Each coordinate in this composition takes value $v^u = 1$. Since all the coordinates in this sequence attain the same value, then by Claim (2.4) they must all be included in the Attacker's support. It follows that $\mathbf{v}_f^0 \in E(\Upsilon^C)$.

By the same argument, it must be the case that any family with a composition of coordinates (C.77) with $q = n$ must be included in the set of extreme points too, $\mathbf{v}_f^0 \in E(\Upsilon^C)$.

We now analyse families of nodes with composition (C.77) in which $q \geq 1$. We consider two cases:

Case 1. $q = 1$. This composition yields a star network degree sequence, which must be always included in $E(\Upsilon^C)$.

Case 2. $q \geq 2$. In this case, sequence (C.77) must have q coordinates of value $v^q = n - 1$ and $n - q$ coordinates of unit value $v^u = 1$. However, any node which attains a unit value is not included in the Attacker's support if $q \geq 2$. To see that observe that the expected gain from an attack on q nodes of values $n - 1$ is:

$$A^q = \frac{(n-1)(q-1)}{q} > 1,$$

which is larger than 1 for any $n > q \geq 2$.

It follows that $\mathbf{v}_f^{q \geq 2} \notin E(\Upsilon^C)$. ■

Therefore, three extreme point families are defined purely by linear constraints in the full support case: a star network, a complete network, and a sequence of unit values. However, we are not considering the latter as an eligible solution for the maximisation problem. Sequence $\mathbf{v}_f^0$ appears as a maximiser for the relaxed maximisation problem at extreme values of $c \to 2$, outperforming a star network. However, the sequence is clearly not-graphic as the minimum possible number of connections that a connected network can have is $(n-1)$, and this sequence has only $\frac{1}{2}n$. Moreover, by Lemma 3.1 we know that a star network must always be an optimal choice for the Defender for the sufficiently high cost of an edge, $c > c_u$.

Partial support Similarly, we now determine extreme points of a hypercube defined by linear constraints (C.47). As case with $k = 2$ was already considered in Subsection C.3, we consider a case in which $k \geq 3$. Any family of extreme points defined by linear constraints (C.47) have coordinates either of value $v^q = n-1$ or $v^k = \frac{k}{k-1}$. Thus, they can be represented by the following composition of coordinates:

$$\mathbf{w}_p^q = \left(\underbrace{n-1, \cdots, n-1}_{q}, \underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{k-q}, \underbrace{A_p^q}_{\times n-k} \right), \tag{C.78}$$

where $q \geq 2$, and

$$A_p^q = \frac{k-1}{\frac{(k-1)(k-q)}{k} + \frac{q}{n-1}}.$$

We now demonstrate that $\mathbf{w}_p^q \in E(\Upsilon^P)$ IFF $q \in \{0, 1, k\}$.

Claim C.20. $\mathbf{w}_f^q \in E(\Upsilon^P)$ *IFF* $q \in \{0, 1, k\}$.

Proof of Claim C.20. First, consider the case in which $q = 0$. The composition of coordinates then has k coordinates which attain value $v^k = \frac{k}{k-1}$ and $n-k$ coordinates of value $v_u = 1$. Since every of the top k nodes in the composition attains the same value, then by Claim 2.4 they must be included in the Attacker's support. It follows that $\mathbf{w}_0^q \in E(\Upsilon^C)$.

By the same argument, the composition in which $q = k$ is included in the set of extreme values of the set Υ^C, $\mathbf{w}_0^k \in E(\Upsilon^C)$.

Now consider sequence (C.20) in which $q \geq 1$.

Case 1. $q = 1$. In this case, the composition has one coordinate of value $v^q = n-1$, $k-1$ coordinates of value $v^k = \frac{k}{k-1}$ and $n-k$ coordinates that attain the value equal to the expected gain from an attack on the top k nodes. Now consider constraint (C.44) for some node y of value v^k.

$$v_y \geq \frac{k-2}{\frac{(k-2)(k-1)}{k} + \frac{1}{n-1}}. \tag{C.79}$$

Substituting $v_y = \frac{k}{k-1}$ into (C.79):

$$\frac{k}{k-1} \geq \frac{k-2}{\frac{(k-2)(k-1)}{k} + \frac{1}{n-1}}. \tag{C.80}$$

Rearranging and simplifying (C.80) yields:

$$(k-1)k^2\left(k^2(n-1) + k(4-3n) + 2(n-1)\right) \geq 0,$$

Which is satisfied whenever:

$$\underbrace{k^2(n-1) + k(4-3n) + 2(n-1)}_{\iota} \geq 0. \tag{C.81}$$

Consider the first derivative of ι w.r.t. n:

$$\frac{\partial \iota}{\partial n} = k^2 - 3k + 2 > 0,$$

which is larger than zero for any $n \geq 3$.

Thus, we verify inequality (C.81) when $n = 4$ (since it is the minimum possible number of nodes that network can have if $n > k \geq 3$):

$$3k^2 - 8k + 6 > 0,$$

which is always satisfied.

It follows that $\mathbf{w}_f^1 \in E(\Upsilon^P)$.

Case 2. $q \geq 2$. We demonstrate that any node which attains value v^k is not included in the Attacker's support in this case. To see that observe that an expected gain from an attack on q nodes of value v^q is always larger than v^k:

$$\frac{(n-1)(q-1)}{q} > \frac{k}{k-1},$$

which is always satisfied for any $k > q \geq 2$ as the minimum value that the LHS of inequality can attain is 2 (when $n = 4$ and $q = 2$), and the RHS of inequality is strictly smaller than 2 for any $k \geq 3$. Thus, $\mathbf{w}_f^{q\geq 2} \notin E(\Upsilon^P)$. ■

To conclude, partial support case yields three families of extreme points defined purely by linear constraints (C.47):

(a) k-family:

$$\mathbf{w}_p^0 = \left(\underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{\times k}, \underbrace{1, \cdots, 1}_{\times (n-k)} \right),$$

where $k \in [2, n-1]$. Note that when, $k = 2$ it yields family (C.50) found in the convex case of partial support extreme points analysis;

(b) Quasi-star family:

$$\left(n-1, \underbrace{\frac{k}{k-1}, \cdots, \frac{k}{k-1}}_{\times (k-1)}, \underbrace{\frac{k-1}{\frac{(k-1)^2}{k} + \frac{1}{n-1}}, \cdots, \frac{k-1}{\frac{(k-1)^2}{k} + \frac{1}{n-1}}}_{\times (n-k)} \right),$$

where $k \in [2, n-1]$. Note that when $k = 2$, it yields family (C.52) found in the convex case of partial support extreme points analysis;

(c) Intermediate family:

$$\mathbf{w}_p^k = \left(\underbrace{n-1, \cdots, n-1}_{\times k}, \underbrace{\frac{(n-1)(k-1)}{k}, \cdots, \frac{(n-1)(k-1)}{k}}_{\times (n-k)}, \right)$$

which is equivalent to to intermediate families of extreme points (C.28) and (C.72) that were discovered in both full and partial support cases.

D Feasibility of intermediate family sequences

In order for an intermediate family sequence to yield a maxi-core network, three conditions must be satisfied:

(a) $v^b = \frac{(n-1)(q-1)}{q}$ must be integer, $v^b \in \mathbb{Z}$;

(b) the sequence must pass Erdos-Gallai graphicality conditions (3.5);

(c) the sum of all degrees in the sequence must be even.

We now consider each of those conditions separately.

Integrality of the sequence The value of peripheral nodes indexed $j > q$ can be restated as follows:

$$v^b = \frac{(n-1)(q-1)}{q} = n - 1 - \underbrace{\frac{n-1}{q}}_{\tau}.$$

Thus, peripheral nodes' value is integer IFF $\tau \in \mathbb{Z}$.

Claim D.1. *Peripheral nodes' value $v^b \in \mathbb{Z}$ IFF $\frac{n-1}{q} \in \mathbb{Z}$.*

Sequence graphicality We now demonstrate that any sequence of an intermediate family with $q \in [2, n-2]$ always satisfies graphicality constraints.

Claim D.2. *Erdos-Gallai graphicality constraints* (3.5) *are always satisfied for any sequence of intermediate family* (3.18) *with $q \in [2, n-2]$.*

Proof of Claim D.2. As intermediate sequence (3.18) has only two types of nodes, it is sufficient to check Erdos-Gallai graphicality conditions for nodes indexed q and n (Tripathi & Vijay, 2003).

First, observe that:

$$q < \frac{(n-1)(q-1)}{q},$$

if $n > \frac{q^2+q-1}{q-1} = q + \frac{1}{q-1} + 2$ or, since $n \in \mathbb{Z}$, when $q < n-2$.

Consider two cases:

Case 1. $q < n-2$. In this case, Erdos-Gallai conditions for node indexed q can be written as follows:

$$q(n-1) \leq q(q-1) + (n-q)\underbrace{\min(q, \frac{(n-1)(q-1)}{q})}_{=q},$$

which is always satisfied as:

$$q(n-1) - q(q-1) - (n-q)q \leq 0,$$

$$0 \leq 0.$$

We first verify the condition for node indexed n:

$$q(n-1) + (n-q)\frac{(n-1)(q-1)}{q} \leq n(n-1),$$

which can be rearranged and simplified as:

$$\frac{(n-1)(n-q)}{q} \geq 0.$$

Thus, in *Case 1*, Erdos-Gallai conditions are always satisfied.

Case 1. $q \geq n-2$. Erdos-Gallai conditions for node indexed q are:

$$q(n-1) \leq q(q-1) + (n-q)\underbrace{\min(q, \frac{(n-1)(q-1)}{q})}_{=\frac{(n-1)(q-1)}{q}},$$

$$q(n-1) \leq q(q-1) + (n-q)\frac{(n-1)(q-1)}{q},$$

which can be rearranged as:

$$q(n-q) \leq \frac{(n-1)(q-1)(n-q)}{q},$$

or:

$$q \leq \frac{(n-1)(q-1)}{q},$$

which is never satisfied for any $q \in \{n-2, n-1\}$. ■

Sum of degrees parity The last thing left to verify is the parity of the sum of all degrees in the intermediate families that satisfy Erdos-Gallai and integrality constraints. First, observe that a q-maxi-core network is always feasible if the Defender has an odd number of nodes. For instance, it is always possible to create a network with $q = \frac{n-1}{2}$ and $v_b = n-3$, which satisfies both Erdos-Gallai and integrality constraints. If the Defender has an even number of vertices, q-maxi-core networks are not feasible since the value of peripheral nodes is either $v^b = \frac{(q-1)(n-1)}{q} \notin \mathbb{Z}$ for any $q \in [2, n-2]$ or the sum of all degrees is not even.

Claim D.3. *If the Defender has an odd number of vertices, maxi-core networks are always feasible. If the Defender has an even number of nodes, maxi-core networks are never feasible.*

Proof of Claim D.3.

The Defender has an odd number of nodes. Suppose that Claim D.1 holds and n is odd. Then peripheral nodes of value v^b in the intermediate family sequence can be either odd or even depending on the parity of q. We consider two cases: (1) q is even, and (2) q is odd.

Case 1. q is even. In this case, the value of the peripheral node can be stated as follows:

$$v^b = \underbrace{n-1}_{\text{even}} - \underbrace{\frac{n-1}{q}}_{\text{even/odd}} .$$

Since $\frac{n-1}{q}$ can be either even or odd, we consider two subcases.

(1.1) $\frac{n-1}{q}$ is even.

Suppose $\frac{n-1}{q}$ is even then: $v^b = \underbrace{n-1}_{\text{even}} - \underbrace{\frac{n-1}{q}}_{\text{even}} = \text{even}$. Then the overall sum of degrees must be:

$$\underbrace{q(n-1)}_{\text{even}} + \underbrace{(n-q)\frac{(n-1)(q-1)}{q}}_{\text{even}} = \text{even}.$$

Therefore, if both q and $\frac{n-1}{q}$ are even, then the resulting degree sequence is graphic.

(1.2) $\frac{n-1}{q}$ is odd.

Suppose $\frac{n-1}{q}$ is odd then: $v^b = \underbrace{n-1}_{\text{even}} - \underbrace{\frac{n-1}{q}}_{\text{odd}} = \text{odd}.$

The overall degree sum is then:

$$\underbrace{q(n-1)}_{\text{even}} + \underbrace{(n-q)\frac{(n-1)(q-1)}{q}}_{\text{odd}} = \text{odd}.$$

Thus, if the sequence with an even number of vertices q of value v^q, but $\frac{n-1}{q}$ is odd, such a sequence is not graphic.

Case 2. q is odd. Suppose $\frac{n-1}{q}$ is odd then:

$$v^b = \underbrace{n-1}_{\text{even}} - \underbrace{\frac{n-1}{q}}_{\textbf{even}} = \text{even}.$$

The sum of degrees is then:

$$\underbrace{q(n-1)}_{\text{even}} + \underbrace{(n-q)\frac{(n-1)(q-1)}{q}}_{\text{even}} = \text{even}.$$

Thus, if q is odd and Claim D.1 holds, the resulting sequence is always graphic.

The Defender has an even number of nodes. Suppose that n is even. Then in order for $v^b = n - 1 - \frac{n-1}{q}$ to be an integer, number of nodes of value $v^q = n-1$ must be odd. It then follows that v^b is always even:

$$v^b = \underbrace{n-1}_{\text{odd}} - \underbrace{\frac{n-1}{q}}_{\text{odd}} = \text{even}.$$

Then the overall sum of degrees must be:

$$\underbrace{q}_{\text{odd}} \cdot \underbrace{(n-1)}_{\text{odd}} + \underbrace{(n-q)}_{\text{odd}} \cdot \underbrace{\frac{(q-1)(n-1)}{q}}_{\text{even}} = \text{odd}.$$

It immediately follows that a sequence like that is not graphic.

Therefore, maxi-core networks are not feasible if the Defender has an even number of nodes. ■

Therefore, if the Defender has an even number of vertices, the relaxed maximisation problem provides only two feasible solutions: a star and complete networks.

E Supplementary material for Subsection 3.4

#	Complete/star
3	1.62353
4	1.55

Table 2: Estimations of edge cost threshold between complete and star formations for networks with 3 and 4 nodes

#	Complete/MC	MC/Star	q	v^s
5	1.6	1.64706	2	2
7	1.71429	1.72482	3	4
9	1.77778	1.78462	2	4
11	1.81818	1.82026	5	8
13	1.84615	1.84828	2	6
15	1.86667	1.86741	7	12
17	1.88235	1.88327	2	8
19	1.89474	1.89521	3	12
∞	2	2	n/a	n/a

Table 3: Estimations of edge cost thresholds between complete, maxi-core (MC), and star formations for networks with odd number of nodes and $n \leq 19$, where q stands for the core size, v^s is a value of a peripheral node in an optimal maxi-core network

#	Complete/star
6	1.67949
8	1.755
10	1.80244
12	1.8347
14	1.85798
16	1.87555
18	1.88927
20	1.90028
∞	2

Table 4: Estimations of edge cost threshold between the star and complete formations for networks with an even number of nodes, $n \leq 20$