Karlsruher Institut für Technologie

Karlsruhe Reports in Informatics 2010,6
Edited by Karlsruhe Institute of Technology,
Faculty of Informatics
ISSN 2190-4782

Towards Efficient Equilibria of Combinations of Network-Formation and Interaction Strategies

Björn-Oliver Hartmann, Klemens Böhm, Christian Hütter

2010

KIT – University of the State of Baden-Wuerttemberg and National Research Center of the Helmholtz Association

Fakultät für Informatik

Towards Efficient Equilibria of Combinations of Network-Formation and Interaction Strategies

Björn-Oliver Hartmann
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
Email: bjoern-oliver.hartmann@kit.edu

Klemens Böhm
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
Email: klemens.boehm@kit.edu

Christian Hütter
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
Email: christian.huetter@kit.edu

***Abstract*—Agents in networks have two strategic choices: They can forward/process incoming service requests – or not, and they can establish additional contacts and maintain or terminate existing ones. In other words, an agent can choose both an action-selection and a link-selection strategy. So far, it is unclear which equilibria exist in such settings. We show that there are the following equilibria: First, an inefficient one where agents leave the network. Second, an equilibrium where agents process requests on behalf of others, i.e., they cooperate. In this second equilibrium, agents distribute their contacts uniformly, which is not efficient. We show that a strategy, we propose in this paper, yields an equilibrium that is *optimal*, i.e., that yields the highest sum of payoffs over all equilibria. If agents base their link-selection decisions on the processing times of their requests, optimal system states can be equilibria.**

***Keywords*-Artificial social systems, peer-to-peer coordination, game theory, distributed problem solving, social and organizational structure**

I. Introduction

In the recent past, network formation has become an important research topic. Agents[1] in networks may act as servers and provide services or information, e.g., process an incoming request, and at the same time they want to consume services provided by others, i.e., act as clients. In this paper, we consider networks where agents may be both client and server. While an agent is interested in consuming services from many different agents, it typically is linked to only few nodes in the network. Thus, an agent may have to send its service requests (*requests* for short) via intermediate nodes. Since forwarding agents can always drop requests, there is a general interest in short *forwarding chains*, i.e., those with few intermediate nodes. In combination with the processing of requests, we investigate another dimension of an agent's strategy space, link selection. While we assume that there is a given underlying network structure, i.e., each node has contacts it cannot choose, there are links/contacts an agent can choose, i.e., *additional contacts*. In this paper, a link between agents is bilateral: Both agents can use it and have to pay for it.

Strategic Choices An agent has certain strategic choices: (a) It can forward or process a request – or not. When forwarding, it can decide which agent to forward to. (b) It can establish additional contacts or terminate existing ones. We refer to the first choice as *action selection*, and to the second one as *link selection*. Networks with these features are ubiquitous, e.g., social networks, cooperation between companies, or P2P networks. Link selection is indeed an issue since maintaining contacts typically incurs costs. While agent behavior includes both link selection and action selection, existing work typically deals with only one aspect (see Section II). Given this, we study agent behavior under a strategic perspective, i.e., node behavior is driven by utility considerations, and address the fundamental question which equilibrium states arise.

Research Questions This paper addresses the following research questions: (a) Which link-selection strategies and action-selection strategies do agents use, and which equilibria exist? Equilibria are a fundamental concept in game theory, and finding them is necessary to analyze strategic choices. (b) A fast answer often is more valuable than a slow one, and we model this with a discount factor. How does this specific parameter affect the strategic choices of nodes? (c) If equilibria exist, how efficient are they? In *efficient networks*, the sum of the payoffs of all agents, i.e., the *social welfare*, is maximal, or, more practically, agents process almost all requests, and the number of intermediaries is small. (d) The relationship between nodes depends on their ability to estimate the degree of cooperation of other nodes. When this ability becomes weaker/stronger, how does this affect the equilibria?

Identifying equilibria in networks where more than two players process a request tends to be difficult [1]. Taking link selection into account makes the problem of identifying equilibria even more involved [2].

Contributions As a first step of our analysis, we specify a strategy space that is very general, i.e., subsumes strategies known from literature. While [3] has identified network structures that are optimal, it is unclear if they are equilibria. We now have conducted a theoretical analysis of the setting described before that shows the following: There are two equilibria. (I) The *inefficient network*: If agents do not cooperate enough, they are dissatisfied with the system

[1] In this article, we use 'agent' and 'nodes' as synonyms.

as a whole and leave the network. (II) The *cooperative equilibrium*: Uncooperative behavior does not yield high payoffs, since agents ignore requests from uncooperative nodes. Consequently, uncooperative strategies do not pay off, and thus, the degree of cooperation is high. If agents do not take the contact distributions of their contacts into account when processing requests, then the optimal strategy is distributing additional contacts uniformly over the network. The resulting network structure is not efficient [3]. A better outcome is possible if agents do not only distinguish between cooperative/uncooperative behavior of other nodes, but take the response times of requests forwarded into account. We propose a strategy, dubbed Drop-Slow-Contacts (DSC), as follows: Nodes drop additional contacts where response times are too high. As a core result of this paper, we show that the DSC is an equilibrium strategy that increases the social welfare, and yields high payoffs.

If agents use the DSC Strategy, then they give each of their contacts an incentive to distribute its additional contacts in a way that is beneficial for the whole system. Our proposed Strategy DSC is cheap, i.e., does not need complex computations or extra messages, and yields high payoffs.

Paper Outline Section II discusses related work. We describe some fundamentals in Section III. Our analysis is in Section IV, and Section V concludes.

II. Related Work

This section addresses related work regarding action selection and link selection.

Action Selection Game-theoretic models [4], [5], [6], [7] as well as behavioral models [8], [9] can help to understand action-selection strategies. First, there are motives for malicious behavior: [8] explains the evolution of ideologically motivated attacks, and describes a set of countermeasures. [9] shows how nodes propagate viruses in an email network, and describes how to immunize against this malicious propagation. The (Iterated) Prisoners' Dilemma [5] explains why free-riding may occur. On the contrary, [6], [7] show that cooperation can evolve through different strategies: Indirect reciprocity and reputation [6] as well as network reciprocity, and group selection [7]. Indirect reciprocity means that Node i only chooses the cooperative strategy towards Node j if j has done so towards other nodes. Reputation in [6] depends on the observations of the node itself as well as on third-party opinions, i.e., feedback. [10] proposes a payment scheme for feedback so that issuing truthful feedback is the optimal strategy. Network reciprocity is indirect reciprocity between members of a forwarding chain. The models used in [6], [7] leave aside intermediate nodes, which is important in real-world networks. Even though intermediate nodes are important in real-world networks, investigating them analytically is difficult [1]. [11] analyzes the cost of selfish routing compared to a centralized solution: The latency is close to optimal in a given network, even though nodes route based on utility considerations. Thus, no centralized instance is necessary for routing. In contrast to our approach, [11] assumes that a network structure is given. In general, however, the structure is the result of the behavior of the agents. Analyzing multi-player games is difficult if the network structure is taken into account [2]. All the approaches mentioned leave network formation aside. The network structure can influence the outcome of a game [12]. Thus, we investigate network formation as well.

Link Selection Network-formation models [13], [14], [15] describe which networks emerge from link selection of nodes. [16] shows which network topologies guarantee strong equilibria, i.e., states where no group of nodes can improve the utility of each of its members. [17] uses link blocking to minimize the propagation of undesirable data in the network. [13] proposes the *Connections Model*: A node benefits from nodes it is directly or indirectly linked with. The benefit decreases, the larger the path length between two nodes is. [13] shows that nodes form different networks contingent on contact-maintenance costs: If contact-maintenance costs are less than the decrease of benefit between a contact and the contact of the contact, *complete networks* where a node is linked to any other node are efficient. If contact-maintenance costs are higher than the expected benefit of a new contact, *empty networks*, i.e., those where no node has a contact, result. For other contact-maintenance costs, *star networks* are efficient, i.e., one node (the *center*) is linked to all other nodes. The star is not an equilibrium. The center of the star can increase its payoff by giving up the center position. Instead, we will focus on situations where all agents have about the same number of contacts. This setup is more likely since the motivation for having many more contacts than other agents is unclear. We are interested in finding equilibria for networks where agents choose their action-selection strategy and their link-selection strategy, in contrast to [13]. [14] comes to results that are similar to [13], but also points out which kinds of *hot spots*, i.e., nodes with high forwarding load, arise from different networks. Both [13], [14] leave out action selection, i.e., do not allow their nodes to drop requests or to use utility-based routing algorithms. Still, [13], [14] show the difference between efficient networks and the equilibria: Self-interested nodes do not form efficient networks in every situation.

Combined Models In simple scenarios with restrictive assumptions there are models that feature both link and action selection: [12] investigates two types of 2×2 games of players that form networks. There, the network structure depends on link costs and on the distribution of strategies among all nodes. [18] shows, in a behavioral experiment, that the network structure influences the payoff of players in network-trade games. There, equilibrium theory is a good predictor for human behavior. Thus, link selection and action selection should be investigated in combination.

III. Definitions and Assumptions

We investigate coordinator-free networks of self-interested agents. In the following, we describe network characteristics, measures used in our study, and the strategy space.

A. Network Characteristics

Fundamentals $N = \{0, ..., n-1\}$ is a set of identifiers of agents that form a network. We see two types of contacts in a network: contacts a node can choose, i.e., *additional contacts*, and those it is always linked to, i.e., *fixed contacts*. This is natural, e.g., relatives or neighbors are fixed, while a person is free to choose his friends or business contacts. Let K_i (K_i^+) be the set of all contacts (set of all additional contacts) of Agent i. Further, let $|\cdot|$ denote the cardinality of a set. E.g., $|K_i|$ is the number of contacts of Agent i. If we do not have a specific agent in mind, we leave the subscript aside. We refer to the network structure consisting of fixed contacts as *fixed network structure*. To keep the analysis manageable, the fixed network structure is a ring in our case, i.e., each agent $i \in N$ has two fixed contacts: the agent left of Agent i, Agent $(i-1)$ mod n and the one to the right, Agent $(i+1)$ mod n.

Kleinberg Distributions In [3] Kleinberg has investigated the routing complexity in random networks with an initial network structure. According to [3] the average number of hops in a ring plus a random matching is minimal if all pairs of nodes (u, v) are contacts with a probability proportional to $[d(u, v)]^{-r}$ (where $d(u, v)$ is the number of steps in the original network, and r the dimension of the network, e.g., $r = 1$ for a ring of nodes). We will refer to these networks as *networks with a Kleinberg distribution*. [3] contains a proof that there exists a constant β so that nodes deliver requests in $\beta \cdot log^2(n)$ steps with high probability. [3] assumes cooperative nodes and leaves action selection aside. Even though [3] is important from an algorithmic perspective, it is still necessary to investigate if agents have an *interest* to form such networks.

B. System Model

One-Shot vs. Repeated Interactions In large networks, we frequently have the following scenario: Endpoints of a forwarding chain interact only once, i.e., play a one-shot game, while contacts interact repeatedly. This scenario is interesting: Theory predicts uncooperative behavior in one-shot cooperation games [5]; on the other hand, cooperation can evolve if the game is played repeatedly [5]. We focus on the described scenario, instead of allowing repeated interactions between endpoints of a forwarding chain. Based on the scenario described, we analyze the payoffs after each agent issued one request to each other agent in the following. We do not investigate any intermediate states, since, otherwise, contacts would also interact only once (or few times), and this would change the nature of the game.

Requests We want to calculate the expected payoff of an agent. To simplify the analysis, we assume that for each request r there exists only one agent that can answer it. Structured P2P systems [19], to give an example, meet this condition. We refer to this agent as the *destination of request r* (short form: $dest(r)$). Note that the issuer does not have to know the destination of a request, but a contact that is closer to the destination. E.g., if someone wants to know 'When did the Mayflower reach Cape Cod?', he does not have to know a history professor – someone who probably knows a professor, e.g., a history student, is sufficient as well. We assume that destinations of requests issued are uniformly distributed. We refer to the creator of a request as the *issuer of request r* (short form: $iss(r)$). If the destination of a request processes the request, it sends the answer directly to the issuer. (We will explain how agents can estimate the degree of cooperation of other agents in Section III-C.) We refer to the first agent that receives a request r as *the first forwarder* (short form *ff(r)*). Next to the first forwarder, there normally are further forwarders, and the average number of hops if forwarding is only along fixed contacts is $\frac{n-1}{4}$: This is because the maximal distance in the fixed network structure is half of the size of the ring ($\frac{n-1}{2}$), and the expected distance is half of it.

Leaving the Network Issuing requests is either beneficial or not. Thus, agents either issue requests, or they leave the network, i.e., use Strategy Dropout (cf. Section III-E).

C. Assumptions

Network We assume that nodes have some address in an address space. This can be their IP-address, a physical address such as a city, or some knowledge domain as in the history example above. Further, we investigate networks where an agent knows which one of its contacts is closest to the destination of a given request. This implies the existence of a distance function in the address space. In our case this is the number of hops in the fixed network structure. Having an intuition about which contact is closest to a given destination is a feature of many networks [20], [21].

Time We assume that agents process requests immediately, and that each message hop lasts a fixed amount of time. Still, the processing time of a request depends on the number of hops during the processing: It is proportional to the number of hops it needs.

Uncertainty We assume that real-world agents use some kind of reputation system [22] to estimate the degree of cooperation of other agents, e.g., eigentrust [22], or feedback payments [10]. The perception of the behavior of other agents may be erroneous.

Utility We assume that an agent's utility increases and decreases with its payoff.

D. Measures

We now introduce measures used in this paper.
The payoff is the difference of income and expenditure.

Definition 3.1: The *payoff* of Agent i is:

$$payoff(i) = \text{income}(i) - \text{expenditure}(i)$$

An agent benefits from the system every time it receives an answer to one of its requests. Let R_i be the set of requests Agent i creates, and let $R_{i \to j} \subseteq R_i$ be those requests that Agent i forwards to Agent j. For some requests Agent i receives an answer: Let $A_i \subseteq R_i$ be the set of requests that have been answered, and let $A_{i \to j} \subseteq R_{i \to j}$ be the set of requests that have been answered and that Agent i had issued to Agent j. The faster an agent i receives an answer, the higher is the benefit, i.e., the benefit of an answer depends on a discount factor $\delta \in [0, 1]$ and on the number of hops $h(req)$ a request $req \in A_i$ needs to be answered. The benefit of an answer is multiplied with cost factor a.

Definition 3.2: The income of an agent i when it receives an answer to its request $req \in A_i$ is:

$$\text{income}(req) = a \cdot \delta^{h(req)}$$

Definition 3.3: The *income of Agent i* is the sum of the incomes corresponding to the answers Agent i recieved:

$$\text{income}(i) = a \cdot \sum_{req \in A_i} \delta^{h(req)}$$

Expenditures are processing costs and contact-maintenance costs. We start by explaining the former.

An agent can process requests on behalf of others. Let F_i be the set of requests Agent i forwards (i.e., $R_i \cap F_i = \emptyset$), and let W_i be the set of requests Agent i answers (W for Work). Let $F_{i \to j} \subseteq F_i$ be the set of requests that Agent i forwards to Agent j, and let $W_{i \to j}$ be the requests that Agent i forwarded to Agent j and that Agent j answers. The cost of issuing (forwarding, answering) a request is q (f, w). Next to processing costs there are contact-maintenance costs. To define them, we need the two following auxiliary functions:

$$g(i, j) = \begin{cases} \textbf{true} & \text{if Node j is additional contact of Node i} \\ \textbf{false} & \text{else} \end{cases} \quad (1)$$

$$nr(i, j) = \text{number of time units when g(i, j) holds} \quad (2)$$

The cost of maintaining an additional contact per time unit is c. $r(i)$ is the number of additional contacts of Node i: $r(i) = \sum_{n \in N} nr(i, n)$

Definition 3.4:

$$\text{expenditure}(i) = q \cdot |R_i| + f \cdot |F_i| + w \cdot |W_i| + c \cdot r(i)$$

Table I lists the abbreviations.

Social Welfare Besides the payoff of an agent, we want to quantify the success of the system as a whole, i.e., the *value function v* [13], which is also known as *social welfare* [23].

Action/Event	**Number of Events**	**Payoff Factor**
Issuing a request	$\|R\|$	q
Forwarding a request	$\|F\|$	f
Answering a request	$\|W\|$	w
Receiving an answer	$\|A\|$	a
Maintaining additional contact	$r(\cdot)$	c

Table I
BENEFITS AND COSTS OF ACTIONS/EVENTS

Definition 3.5: The value function v of the system is:

$$v = \sum_{i \in N} payoff(i)$$

Definition 3.6: A system is *efficient* if its value function v is maximal.

Algorithm 1: isCooperative(Agent i, Threshold t)

Input: Agent i, Threshold t
Output: {TRUE, FALSE}
double trueCooperation = C(i);
if $C(i) \geq t$ **then**
  return $getRandomValue[0, 1] \geq reportingError$;
else
  return $getRandomValue[0, 1] < reportingError$;
end

Cooperation The efficiency of a system depends on the degree of cooperation of its agents. We refer to the *degree of cooperation* of Agent i as $C(i)$. $C(i)$ is the ratio of all requests Agent i processes among the ones received. Following our assumption **Uncertainty**, agents use some reputation system. To abstract from the concrete method to estimate the degree of cooperation, we propose having (I) an abstract method and (II) an unreliability factor. (I) With Method 'isCooperative(Agent j, Threshold t)' an agent i can test whether the degree of cooperation $C(j)$ of an agent j is higher than or equal to a threshold t_i chosen by i (cf. Algorithm 1). (II) Unreliability factor $reportingError \in [0, 1]$ models the extent of errors in the estimation, i.e., $reportingError$ is the ratio of the cases when the result is incorrect.

Connectivity Two auxiliary functions are the *distance between agents* and the *normalized distance*. The former is the number of intermediaries between two agents in the fixed network structure, and the latter is the normalization of it.

Definition 3.7: The *distance* between two Agents i, j is: $d(i, j) = min(|i - j|, n + min(i, j) - max(i, j))$

Example: Figure 1 shows a ring with six agents. To determine the distance between Agent 4 and 0, one has

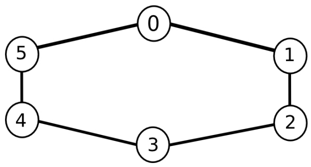

Figure 1. Ring with Six Agents

Dimension	Type	Description
$sendRequest$	binary	If true, then the agent issues requests.
$contactC$	double $\in [0,1]$	Minimal degree of cooperation to consider to process a request.
$ownCoop$	double $\in [0,1]$	Ratio of processed requests from issuers and predecessors that are deemed cooperative.
$maxC$	integer $\in [0,\|N\|]$	Maximal number of additional Contacts.
$additionalCC$	double $\in [0,1]$	Minimal degree of cooperation, to become a contact.
$maxProcessingTime$	double $\in [0,\infty)$	Maximal request-processing time.
$distribution$	arbitrary distribution	Distribution of contacts.

Table II
STRATEGY-SPACE PARAMETER

to count clockwise from 4 to 0. The difference is two: $d(4,0) = min(|4-0|, 6 + min(4,0) - max(4,0)) = min(|4-0|, 6+0-4) = min(4,2) = 2$. The distance between Agent 1 and 3 is: $d(1,3) = min(|1-3|, 6+1-3) = min(2,4) = 2$. ■

Definition 3.8: The *normalized distance* $\Delta(\cdot,\cdot)$ between Agent $i \in N$ and Agent $j \in N$ is: $\Delta(i,j) = \frac{4}{n-1} \cdot d(i,j)$
One measure that describes how efficiently agents use the system is the average number of hops $\overline{h}$. A smaller average number of hops leads to less forwarding load (if all other parameters remain the same).

Definition 3.9:

$$\overline{h} = \frac{1}{|A|} \cdot \sum_{req \in A} h(req)$$

Measure $\overline{h}_\Delta(i)$ is the number of hops relative to the distance to the destination for Agent i's requests:

Definition 3.10:

$$\overline{h}_\Delta(i) = \frac{1}{|A_i|} \cdot \sum_{req \in A_i} \frac{h(req)}{\Delta(i, dest(req))}$$

$\overline{h}_\Delta(i|j)$ is the average number of hops of Agent i's requests that it has forwarded to its contact j:

Definition 3.11:

$$\overline{h}_\Delta(i|j) = \frac{1}{|A_{i\to j}|} \cdot \sum_{req \in A_{i\to j}} \frac{h(req)-1}{\Delta(j, dest(req))}$$

E. Strategy Space

Agent behavior can be manifold. There are aspects of agent behavior that are well known and do not change. For instance, [24] shows that humans use threshold strategies when dealing with requests on behalf of others. Consequently, we describe these known aspects of the behavior by means of pseudocode. Still, the respective threshold value is unclear, and we allow agents to choose this value, as we will explain in the remainder of the section. Apart from threshold strategies, most aspects of agent behavior are unknown, and we model them by means of a vector of parameters called *strategy vector* S. We start with action selection and continue with link selection. Table II serves as a summary.

Issuing An agent may issue requests or may choose not to. (We do not look at strategies where agents sometimes issue and sometimes do not issue requests.) Only if binary parameter $sendRequests$ is true, an agent issues requests.

Processing Humans in coordinator-free environments use *threshold strategies* when processing a request on behalf of others [24]: If a player i issues requests to a contact j, and the fraction of these requests answered is less than a certain threshold value, player i ignores requests from contact j. In other words, human players hold a contact responsible for the whole forwarding chain (even though someone else in the forwarding chain might have dropped the requests). This makes sense – a contact is responsible for selecting cooperative contacts. Agents in our scenario use threshold strategies as well, i.e., behave as described in Algorithm 2. For each request, an agent tests whether the predecessor in the forwarding chain qualifies (Line 1). 'Qualifies' means that the forwarder has a degree of cooperation that is at least as high as the expectation of the agent: An agent ignores requests from contacts which have a lower degree of cooperation than threshold $contactC$ (contact-cooperation threshold). An agent can use a timeout scheme to detect whether a request has been dropped. Further, from an economic point of view, it does not make a difference whether a contact drops a request, or whether it is lost due to a technical defect etc. In both cases, a contact that would have processed the request is more useful for the issuer.

Algorithm 2: Dealing with Requests

```
   Input: Request r, Predecessor p
1  if isCooperative(p, contactC) and
   getRandomValue[0,1] < ownCoop then
2  |  process r;
3  else
4  |  drop r;
5  end
```

An agent can decide whether it is willing to process a request even if the predecessor in the forwarding chain qualifies (Line 1). Thus, in our model, an agent processes a request with probability $ownCoop$ (Line 1).

Link Selection The link-selection strategy of an agent specifies under which conditions the agent chooses and maintains additional contacts. We use four parameters/conditions to model link selection. Note that all four conditions have to hold, otherwise the agent drops the respective additional contact.

1) An agent can specify how many additional contacts it wants to have at most, by parameter $maxC$.
2) The degree of cooperation of a potential contact has to be higher than threshold $additionalCC$ (additional contact cooperation threshold).
3) The average processing time of requests an agent i forwards to a contact j, i.e., $\overline{h}_{\triangle}(i|j)$, has to be less than or equal to threshold $maxProcessingTime$. While parameter $additionalCC$ defines which degree of cooperation an agent demands from its additional contacts, $maxProcessingTime$ is an additional condition on the average processing times.
4) An agent can specify the distribution of its contacts: Parameter $distribution$ is a probability distribution over the underlying address space (or a density function in case of a continuous address space). If an agent i fulfills Conditions 1 to 3, then a potential additional contact chooses i with a probability proportional to the specified value in the probability distribution (cf. [3]).

Common Strategies Having introduced the strategy space, we now explain that one can model strategies known from literature:

- An *Action-Selection Altruist* [25] does not use the system, but processes each request it receives ($ownCoop = 1$, $sendRequest = false$ and $contactC = 0$).
- A *Link-Selection Altruist* [25] accepts arbitrarily cooperative contacts ($additionalCC = 0$, and $maxC$ is a large constant).
- An *Action-Selection Free-Rider* does not process requests on behalf of others: $ownCoop = 0$ [4].
- A *Link-Selection Free-Rider* does not contribute to the network structure, i.e., $maxC = 0$ [26].
- A *Dropout* neither uses the network nor contributes to it ($sendRequest = false$, $maxC = ownCoop = 0$).
- *Threshold Strategies:* Humans cooperate with others only if they process a certain number of their requests, i.e., humans set $contactC$ to some value v. The same is known for choosing contacts [27], i.e., humans normally set $additionalCC$ to some value.

Thus, we can describe strategies known from literature by our strategy set, as well as many other strategies.

To ease the analysis to some degree, we assume that $maxC$ is an exogenous parameter. In Appendix E, we experimentally identify values for $maxC$ in different equilibria using evolutionary algorithms.

To sum up, the strategy space S of an agent is as follows:

$$S = \{sendRequest, contactC, ownCoop, additionalCC, maxProcessingTime, distribution\} \quad (3)$$

Further, let $S(i)$ be the strategy vector of Agent i.

IV. Network Efficiency

Our objective is finding network equilibria. In our analysis, we start with equilibria for pairs of contacts. Afterwards, we use these results to identify equilibria in the general case, i.e., $n \geq 2$.

Nash Equilibrium A Nash equilibrium is a state where no player i can increase its payoff $payoff(i)$ by changing its strategy $S(i)$ [28].

Cooperative Equilibrium and Inefficient Network In the following, we will show that if all agents have the same strategy, and if the strategy is as follows, then they are in an equilibrium:

$$S_c = \{true, 1.0, 1.0, 1.0, \overline{h}_{\triangle}(i), kleinberg\} \quad (4)$$

We refer to this equilibrium as the cooperative equilibrium: All agents send requests, cooperate and have Kleinberg-distributed additional contacts, i.e., all agents have Strategy S_c. Further, they do not accept agents as additional contacts that process their requests slower than their additional contacts do on average. This strategy yields high social welfare, as we describe in Section IV-B2. Next to the cooperative equilibrium there is a second one. We refer to it as the inefficient network, i.e., have Strategy S_i: Agents leave the network and consequently do not have any distribution of additional contacts, nor any threshold for the processing times of their requests, both denoted as '–'.

$$S_i = \{false, 0.0, 0.0, 0.0, -, -\} \quad (5)$$

Altruism We exclude Strategy Action-Selection Altruist. This strategy has only expenditures, but never income. Strategy Dropout always is more successful. Thus, being an Action-Selection Altruist is never part of an equilibrium.

In the following, we illustrate the situation only for one agent, Player i, when investigating symmetric cases.

A. Contact Relations

We start with a simple setup: We analyze the relation between two players, Player i and Player j. This setup allows studying the relationship between two contacts. In this section, we show that there are (at least) two equilibria in the two-player setup: the inefficient network and the cooperative one. While the inefficient network always is an equilibrium, cooperation is contingent on certain conditions, as we explain in the following.

Inefficient Network Both players are dropouts (cf. Section II; $sendRequest = false$). None of them has an advantage by joining the network again, since the other one is not contributing. Strategy Dropout (with zero payoff) dominates action-selection free-riding: action-selection free-riders have expenditures but no income.

Cooperative Equilibria We now find exogenous parameter values so that full cooperation, i.e., $C(i) = 1.0$, is an equilibrium.

Lemma 4.1: Two contacts, Player i and Player j, cooperate to the highest possible extent if the following conditions hold. (Note that the conditions also have to hold in the symmetric case, where i is exchanged with j and j with i.)

$$isCooperative(i, additionalCC_j = 1)$$

and

$$isCooperative(i, contactC_j = 1)$$

and

$$\frac{ownCoop_i}{|R_{i\to j}|} < \frac{a \cdot ownCoop_j - q - c \cdot r(i)}{w \cdot |W_{j\to i}| + f \cdot |F_{j\to i}|}$$

Appendix B contains a proof.

The third condition describes the relation of the cost factors and the number of requests Agent i creates and forwards to Agent j ($|R_{i\to j}|$). As long as the benefit for receiving an answer (a) is high compared to the cost of issuing requests (q) or maintaining additional contacts (c) as well as the cost of processing requests ($w \cdot |W_{j\to i}| + f \cdot |F_{j\to i}|$), the third condition holds. Thus, the first two conditions are much more important: A player has to be fully cooperative, since players accept only fully cooperative contacts.

Note that Lemma 4.1 has an important implication: In the cooperative equilibrium, contacts cooperate at least as much as their contacts expect from them, and vice versa. Thus, only *bilateral contacts*, i.e., Agent i trusts Agent j, and vice versa, are stable.

In the efficient state the degree of cooperation is maximal, i.e., $ownCoop_i = contactC_j = additionalCC_j = 1$. Here, the efficient state is only an equilibrium if the error when estimating the degree of cooperation is less than 50%, see Appendix F.

B. n-Player Game

In the n-Player Game, agents can choose actions and contacts. Solving games with n players where players can modify the structure of the system is difficult [1], [2]. Thus, we take a statistical view. We assume that agents choose the position of their additional contacts following a probability distribution.

As we have seen in Section IV-A, agents cooperate with their contacts. Thus, we can assume that agents choose the closest contact to the destination of a request when forwarding requests.

With n players, agents have to forward many requests, and the structure of the system, i.e., link selection, plays a greater role. In the following, we identify the link-selection choices of an agent in an equilibrium.

Compared to the 2-Player Game, the network structure might change in n-Player Games, and forwarding costs play a role. While the income is as in Definition 3.3, the expected value of the expenditure $E[expenditure(i)]$ changes. In particular, cost of forwarding and contact-maintenance costs increase.

$$\begin{aligned} &E[expenditure(i)] = \\ &\quad q|R_i| + ownCoop_i(f|F_i| + w|W_i|) + c \cdot r(i) \end{aligned} \tag{6}$$

For ($E[income(i)] > E[expenditure(i)]$), Strategy Dropout again is not beneficial. Formula (6) depends on Player i's forwarding load $|F_i|$. The forwarding load itself depends on the network structure.

1) Contact Selection – Simplistic Case: The network structure results from link-selection strategies. In the following, we investigate two cases: In Case A, agents base their decision of processing a request on the contact structure of their contacts, in Case B, they take the distribution of the agents in a forwarding chain (over the address space) into account. As we will see, the strategies for both cases differ. We start with Case A. Case B follows in Section IV-B2.

As a prerequisite for our analysis, we have to understand the forwarding load in the fixed network structure. In such a network, i.e., a ring of players, the forwarding load is likely to be high. Let $\overline{C}$ be the average degree of cooperation. Then the probability $p(h)$ that a request is processed over h hops is: $p(h) = \overline{C}^h$

Example: Figure 2 graphs $p(h)$. It shows $p(h)$ for a small population ($40 + 1$ agents). Every agent processes 80% of the requests. Since the initial structure is a ring, there are at most 20 agents in the forwarding chain. With each step the probability that the request will be processed successfully decreases. ■

For large n, $p(h)$ converges against a continuous distribution. Hence, we show the continuous case here and in the following.

No Additional Contacts If an agent does not have any additional contacts, the *expected benefit of sending a request over h hops E(h)* is: $E(h) = p(h) \cdot \delta^h \cdot a - q$. The *expected benefit of issuing a set of $n - 1$ requests such that the destinations of the requests are in the address space of a different agent* is:

$$E(n) = 2 \cdot \sum_{i=1}^{(n-1)/2} \big(p(i) \cdot \delta^i \cdot a - q\big) \tag{7}$$

We multiply the sum with two, since there are $\frac{n-1}{2}$ agents *left and right* on the ring structure. Note that we omit

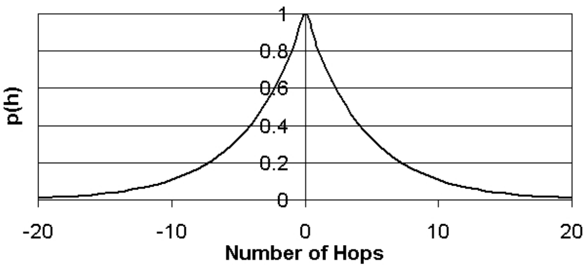

Figure 2. Request-Processing Prob.: No Contacts

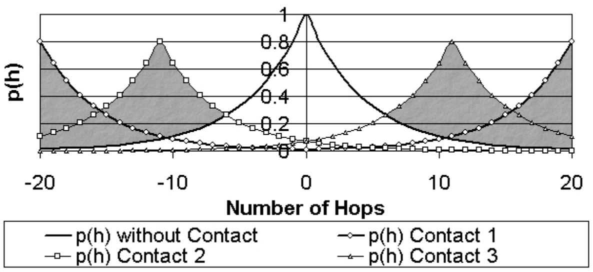

Figure 3. Request-Processing Probability: Three Contacts

rounding operators for non-integer values of positions or numbers of nodes to ease presentation. – Even though Formula (7) gives us the expected benefit, we will also introduce an approximation of it that is easier to work with:

$$E_a(n) = 2 \cdot \int_1^{\frac{n-1}{2}} \left(p(h) \cdot \delta^h \cdot a - q\right) \text{ dh} \tag{8}$$

$E_a(n)$ converges against $E(n)$ for large numbers of agents.

Expected Benefit with Additional Contacts With additional contacts the expected benefit $E(n)$ changes.

Example: For an illustration see Figure 3. In contrast to the situation illustrated in Figure 2, the agent has three additional contacts. One is exactly 20 hops away, the other ones 11 hops (one right, one left). The grey area is the probability mass won, i.e., where the agent had failed to get answers before it has added the contacts. ■

The benefit of having additional contacts is maximal if the probability mass won is maximal. If agents do not take the contacts of contacts into account, the distribution of additional contacts is approximately *optimal*, i.e., brings the maximal benefit, if they all have about the same distance to the next contact. For the exact positions see Lemma 4.2.

Optimal Positions An agent that wants to maximize its payoff has to find optimal positions $p_i, i \in [1, ..., |K^+|]$ for its $|K^+|$ additional contacts. (For the sake of simplification we assume that the number of additional contacts is odd.) If an agent does not take the distribution of contacts of contacts into account, the optimal positions of its contacts are as follows:

Lemma 4.2: If agents process requests independently from the contact distribution of their contacts, then the optimal positions of the contacts of an agent with position $p_0 = 0$ are as follows: One of the contacts is on the other side of the ring, i.e., $p_{|K^+|} = \frac{n-1}{2}$; every other contact with position p has a counterpart with position $(n-p)$. For $i = 1 ... \frac{|K^+|-1}{2}$ the optimal positions are as follows:

$$\frac{n-1}{2} \cdot \frac{2 \cdot i}{|K^+|+1} + 2 \cdot \frac{|K^+|-i}{|K^+|+1} - 4 \cdot \frac{|K^+|-i}{|K^+|+1} \cdot \frac{ln(\frac{1+\delta}{2})}{ln(C \cdot \delta)} \tag{9}$$

Appendix A contains a proof.

Example: Agent i with position $p_i = 0$, $|K_i^+| = 3$ calculates the optimal positions of its three additional contacts. For an average degree of cooperation of 80% ($\overline{C} = 0.8$), 40 agents, and a discount factor of 99% ($\delta = 0.99$), the following positions are optimal (cf. Figure 3): 1. One agent is on the other side of the ring: $p_2 = 20$. 2. The other two agents are almost between the other two, i.e., have position $p_1 = 11$ and $p_3 = 29$. The exact position(s) for p_1 [and p_3] are: $roundToInt\left([n-]\left(20 \cdot \frac{2}{4} + 2 \cdot \frac{2}{4} - 4 \cdot \frac{ln(\frac{1+0.99}{2})}{ln(0.8 \cdot 0.99)}\right)\right)$ ■

Lack of Efficiency The situation in Figure 3 is not efficient, because contacts and contacts of contacts have almost the same positions, i.e., contacts of contacts lead only to little benefit. E.g., the agents in the previous example would be additional contacts of each other, i.e., they would not benefit from contacts of contacts.

We find the implications of Lemma 4.2 surprising: Without an incentive to form efficient network structures, payoff-maximizing agents distribute their contacts uniformly, and we are not aware of a real-world system that gives such an incentive.

2) Contact Selection – General Case: We now investigate networks where agents process requests of their contacts dependent on how fast their contacts (and the forwarding chains) process their requests, i.e., we are in Case B.

Contact relations have two aspects: First, an agent wants to have additional contacts so that it can get answers to its requests. Second, an agent wants to be chosen as an additional contact: Namely, since contact relations are bilateral, this is a prerequisite for the first aspect. The first aspect leads to the following: Agents try to choose agents as additional contacts that have Kleinberg-distributed contacts (because then request processing is fastest). Because of the second aspect, an agent is motivated to have Kleinberg-distributed additional contacts himself, because otherwise the agent is not attractive as an additional contact for other agents. While Kleinberg et al. [3] show that their distribution is optimal, it is unclear whether payoff-maximizing agents form such networks. Our contribution is to show that a Kleinberg distribution is indeed an equilibrium. To do so, we take a Kleinberg distribution for each agent as a starting point and check whether an agent has a reason to change its contact distribution. Note that we already know from Section IV-A that additional contacts cooperate.

Beneficial Contacts Due to the discount factor, the benefit of an agent decreases with the number of hops/amount of time between issuing a request and receiving an answer. Two points influence the number of hops: (i) The number of contacts of an agent and (ii) the distribution of additional contacts of an agent's contacts. For (i) an agent needs additional contacts and has to be attractive for other agents. An agent is attractive if its contacts process requests quickly (ii). According to [3], agents route requests optimally if they use a Kleinberg distribution.

Evaluation of Contacts Since an agent cannot control directly how its contacts distribute their contacts, it has to rely on another technique. One option is observing the number of hops (or the time) until it obtains an answer for requests forwarded to the contact in question. If requests routed over a contact are forwarded over too many hops (and the discount factor is high), the contact is not useful enough.

Drop-Slow-Contacts (DSC) Strategy: *An agent i measures the time its requests need, i.e., $\overline{h}_\triangle(i)$. It drops an additional contact j if it processes requests slower than an average contact, i.e., if $\overline{h}_\triangle(i) < \overline{h}_\triangle(i|j)$ holds.*

Appendix G proves that DSC is an cooperative equilibrium.

This strategy gives agents an incentive to process requests as fast as possible if they want to have additional contacts themselves, since they compete with other contacts. Note that this strategy is cheap, i.e., does not need extra messages or complex computations.

Example: Agent i has three additional contacts: Contacts j, k, l. Suppose that the requests of Agent i have an average processing time of $\overline{h}_\triangle(i) = 2.\overline{6}$, and the average processing times of requests that Agent i forwarded to its contacts are: $\overline{h}_\triangle(i|j) = 2$, $\overline{h}_\triangle(i|k) = 2$, and $\overline{h}_\triangle(i|l) = 4$. Since $\overline{h}_\triangle(i) < \overline{h}_\triangle(i|l)$, Agent i should drop Contact l, or try to exchange l against an agent with better performance. ■

We now look at the implications of DSC. First, we look at a special case, i.e., is there an incentive to change from Kleinberg-distributed additional contacts to a uniform distribution? Then we deal with the general case.

Let us assume that all agents have distributed their contacts according to a Kleinberg distribution, except for Agent u. u has uniformly distributed contacts and is additional contact of Agent i. Should i replace u with another Agent k? To answer this question, we look at the expected number of hops of Agent i's requests, depending on the contact.

Contact with Uniform Contacts: If Agent i keeps Contact u, the expected value of the number of hops is $(1 + \beta \cdot log^2(n))$ (with a fixed constant β from the respective proof in [3]): The uniformly distributed contacts of Contact u are not beneficial to Agent i, when i wants to forward a request to an agent in the neighborhood of u, because the additional contacts of u are not close to u itself. Since Contact u does not have additional contacts close to itself, it can only give the request to one of its fixed contacts. If its contacts follow a Kleinberg distribution, the expected value for the number of hops is $\beta \cdot log^2(n)$ [3]. This means that there is an extra step due to Contact u's lack of contacts in its neighborhood.

Contact with Kleinberg Distributed Contacts: If Agent i exchanges Contact u with uniformly distributed additional contacts against Agent k with Kleinberg-distributed additional contacts, the expected value for the number of hops is $\beta \cdot log^2(n)$ [3] and not $1 + \beta \cdot log^2(n)$.

Lemma 4.3: The payoff of Agent i increases for every request it forwards to Agent k with Kleinberg-distributed additional contacts, compared to the case where i uses Contact u with uniformly distributed additional contacts, by the following amount:

$$a \cdot \overline{C}^{\beta \cdot log^2(n)} \cdot \delta^{\beta \cdot log^2(n)} \cdot (1 - \overline{C} \cdot \delta)$$

Appendix C contains a proof. – The consequence is as follows: Agent i exchanges u against Agent k, since it leads to a higher payoff. This means that agents prefer contacts with a Kleinberg distribution. In other words, an agent in a Kleinberg network cannot increase its payoff by switching to a uniform distribution: Its additional contacts would replace it with other contacts, i.e., this agent would loose its contacts.

Other Distributions: Contacts that have a Kleinberg distribution with $r \neq 1$ (cf. Section II), i.e., with an inefficient r-value, or any other distribution are not stable either. If an additional contact has not distributed its additional contacts in an optimal way, then the number of hops increases, i.e., there exists an $\epsilon > 0$ so that the number of hops increases by about ϵ steps. Consequently, the payoff of an agent that has an additional contact with a suboptimal distribution of additional contacts is lower.

Lemma 4.4: The payoff of Agent i increases for every request it forwards to Agent k with Kleinberg-distributed additional contacts, compared to the case where i uses Contact u with a distribution of additional contacts that is different from a Kleinberg distribution.

Appendix D contains a proof. – Due to Lemma 4.4, an agent exchanges a contact with non-optimal additional contacts against an agent with Kleinberg-distributed additional contacts.

Summary In Lemma 4.1 we show under which conditions contacts cooperate. Further, agents either use the DSC Strategy, or they do not. For the second case, Lemma 4.2 shows how agents should choose their additional contacts. Note that this situation is inefficient (cf. Section IV-B1). The social welfare is higher in the first case where agents use the DSC Strategy. Lemmata 4.3 and 4.4 show that all agents using the strategy and having Kleinberg-distributed additional contacts are in equilibrium.

V. Conclusions

Agents in networks typically can choose both their action-selection as well as their link-selection strategy. Understanding the behavior of agents is essential to identify promising strategies, or to design mechanisms to increase the social welfare. We have identified two equilibria: (I) Contacts do not cooperate. In this case it is rational to leave the network. (II) Agents use the DSC Strategy proposed in this article. It is as follows: An agent measures the time its requests passed on to an additional contact need to be answered. If this time is larger than the average processing time, the agent drops the additional contact. Further, contacts cooperate with each other, and all agents have Kleinberg-distributed additional contacts. The DSC Strategy is cost-free, and agents can increase the social welfare by using it.

References

[1] G. Gottlob, G. Greco, and F. Scarcello, "Pure Nash Equilibria: Hard and Easy Games," in *ACM Theoretical Aspects of Rationality and Knowledge*, 2003.

[2] G. Valiant and T. Roughgarden, "Braess's Paradox in Large Random Graphs," in *ACM EC*, 2006.

[3] J. Kleinberg, "Complex Networks and Decentralized Search Algorithms," in *International Congress of Mathematicians (ICM)*. ACM Press, 2006.

[4] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust Incentive Techniques for Peer-to-Peer Networks," in *ACM EC*, 2004.

[5] R. M. Axelrod, *The Evolution of Cooperation*. Perseus Books Group, 2006.

[6] M. Nowak and K. Sigmund, "Evolution of Indirect Reciprocity," *Nature*, 2005.

[7] M. Nowak, "Five Rules for the Evolution of Cooperation," *Science*, 2008.

[8] N. Stakhanova, O. Stakhanov, and A. Ghorbani, "A Behavioral Model of Ideologically-motivated "Snowball" Attacks," *ARES*, 2008.

[9] J. Liu, C. Gao, and N. Zhong, "Virus Propagation and Immunization Strategies in Email Networks," in *ADMA*, 2009.

[10] R. Jurca and B. Faltings, "Robust Incentive-Compatible Feedback Payments," in *TADA/AMEC*, 2006.

[11] T. Roughgarden and E. Tardos, "How Bad Is Selfish Routing?" *J. ACM*, 2002.

[12] S. Berninghaus and B. Vogt, "Network Formation and Coordination Games," *Advances in Understanding Strategic Behaviour: Game Theory, Experiments, and Bounded Rationality*, 2004.

[13] M. Jackson, "A Survey of Models of Network Formation: Stability and Efficiency," in *Group Formation in Economics: Networks, Clubs, and Coalitions*. Cambridge University Press, 2003.

[14] N. Christin and J. Chuang, "A Cost-Based Analysis of Overlay Routing Geometries," in *IEEE Conference on Computer Communications*, 2005.

[15] J. Larrosa and F. Tohme, "Network Formation with Heterogenous Agents," EconWPA, Microeconomics 0301002, 2003.

[16] A. Epstein, M. Feldman, and Y. Mansour, "Strong Equilibrium in Cost Sharing Connection Games," in *ACM EC*, 2007.

[17] M. Kimura, K. Saito, and H. Motoda, "Minimizing the Spread of Contamination by Blocking Links in a Network," in *AAAI'08*, 2008.

[18] J. S. Judd and M. Kearns, "Behavioral Experiments in Networked Trade," in *EC*. ACM, 2008.

[19] S. Androutsellis-Theotokis and D. Spinellis, "A Survey of Peer-to-Peer Content Distribution Technologies," *ACM Comput. Surv.*, 2004.

[20] P. S. Dodds, R. Muhamad, and D. J. Watts, "An Experimental Study of Search in Global Social Networks," *Science*, 2003.

[21] S. Matsuyama and T. Terano, "Analyzing Dynamics of Peer-to-Peer Communication -From Questionnaire Surveys to Agent-Based Simulation," in *Multi-Agent-Based Simulations*, 2006.

[22] A. Jøsang, R. Ismail, and C. Boyd, "A Survey of Trust and Reputation Systems for Online Service Provision," *Decision Support Systems*, 2007.

[23] R. Cavallo, "Social Welfare Maximization in Dynamic Strategic Decision Problems," Ph.D. dissertation, Computer Science, Harvard University, 2008.

[24] S. Schosser, K. Böhm, R. Schmidt, and B. Vogt, "Incentives Engineering for Structured P2P Systems - a Feasibility Demonstration using Economic Experiments," in *ACM EC*, 2006.

[25] E. Fehr and U. Fischbacher, "The nature of human altruism," *Nature*, 2003.

[26] B.-O. Hartmann, K. Böhm, A. Khachatryan, and S. Schosser, "The Dangers of Poorly Connected Peers in Structured P2P Networks and a Solution Based on Incentives," in *WI*, 2007.

[27] B.-O. Hartmann, K. Böhm, and B. Vogt, "The Network Game: Analyzing Network-Formation and Interaction Strategies in Tandem," in *IAT*, 2008.

[28] J. F. Nash, "Non-Cooperative Games," *Annals of Mathematics*, vol. 54, 1954.

[29] J. Maynard Smith, "The Theory of Games and the Evolution of Animal Conflicts," *Journal of Theoretical Biology*, 1974.

APPENDIX

A. Appendix – Structure

We now calculate the optimal positions $p_1, ..., p_{n_c}$ of n_c additional contacts formally. Without loss of generality, we assume that the additional contacts have positions $p_1, ..., p_{n_c/2}$ left of the agent on the ring, and $p_{n_c/2+1}, ..., p_{n_c}$ right on the ring. Further, we assume that $p_1 > p_2 > ... > p_{n_c/2} > 0 < p_{n_c/2+1} < ... < p_{n_c}$ holds. If so, the functions that describe the probability that a contact successfully processes a request have intersection points at the following positions:
1. Intersection point for the closest agent left and right of the agent: $\{\frac{p_{n_c/2}-1}{2}, \frac{1+p_{n_c/2+1}}{2}\}$
2. For all other agents i, j with $p_i < p_j$: $\frac{p_i+p_j}{2}$
3. For p_1 and p_{n_c}: $\frac{p_{n_c}+p_1}{2}$

Thus, the expected benefit for all requests is:

$$
\begin{aligned}
&E(p_1, ..., p_{n_c}) \\
&= \int_{\frac{p_{n_c}+p_1}{2}}^{\frac{p_1+p_2}{2}} p(|h-p_1|+1)\cdot\delta^{|h-p_1|}\cdot a \text{ dh} \\
&+ ... \\
&+ \int_{\frac{p_{n_c/2-2}+p_{n_c/2-1}}{2}}^{\frac{p_{n_c/2-1}+p_{n_c/2}}{2}} p(|h-p_{n_c/2-1}|+1)\cdot\delta^{|h-p_{n_c/2-1}|}\cdot a \text{ dh} + \int_{\frac{p_{n_c/2-1}+p_{n_c/2}}{2}}^{\frac{p_{n_c/2}-1}{2}} p(|h-p_{n_c/2}|+1)\cdot\delta^{|h-p_{n_c/2}|}\cdot a \text{ dh} \\
&+ \int_{\frac{p_{n_c/2}-1}{2}}^{\frac{1+p_{n_c/2+1}}{2}} p(h)\cdot\delta^{h}\cdot a \text{ dh} \\
&+ \int_{\frac{1+p_{n_c/2+1}}{2}}^{\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}} p(|h-p_{n_c/2+1}|+1)\cdot\delta^{|h-p_{n_c/2+1}|}\cdot a \text{ dh} + \int_{\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}}^{\frac{p_{n_c/2+2}+p_{n_c/2+3}}{2}} p(|h-p_{n_c/2+2}|+1)\cdot\delta^{|h-p_{n_c/2+2}|}\cdot a \text{ dh} \\
&+ ... \\
&+ \int_{\frac{p_{n_c-1}+p_{n_c}}{2}}^{\frac{p_{n_c}+p_1}{2}} p(|h-p_{n_c}|+1)\cdot\delta^{|h-p_{n_c}|}\cdot a \text{ dh}
\end{aligned} \tag{10}
$$

To find a maxima for $E(p_1, ..., p_{n_c})$ we calculate the derivative of it, i.e., the gradient. Due to lack of space, we only show the partial derivative according to the first parameter p_1, i.e., $\frac{\partial E(p_1,...,p_{n_c})}{\partial p_1}$. The other parameters can be calculated analogously. The partial derivative of the parameter p_1 is:

$$
\begin{aligned}
\frac{\partial E(p_1, ..., p_{n_c})}{\partial p_1} &= \underbrace{\frac{\partial}{\partial p_1}\int_{\frac{p_{n_c}+p_1}{2}}^{\frac{p_1+p_2}{2}} p(|h-p_1|+1)\cdot\delta^{|h-p_1|}\cdot a \text{ dh}}_{\text{Case (A)}} \\
&+ \underbrace{\frac{\partial}{\partial p_1}\int_{\frac{p_1+p_2}{2}}^{\frac{p_2+p_3}{2}} p(|h-p_2|+1)\cdot\delta^{|h-p_2|}\cdot a \text{ dh}}_{\text{Case (B)}} + \underbrace{\frac{\partial}{\partial p_1}\int_{\frac{p_{n_c-1}+p_{n_c}}{2}}^{\frac{p_{n_c}+p_1}{2}} p(|h-p_{n_c}|+1)\cdot\delta^{|h-p_{n_c}|}\cdot a \text{ dh}}_{\text{Case (C)}} \\
&= a\cdot(\int_{\frac{p_{n_c}+p_1}{2}}^{\frac{p_1+p_2}{2}} \frac{\partial}{\partial p_1} p(|h-p_1|+1)\cdot\delta^{|h-p_1|} \text{ dh} + \frac{1}{2}\cdot p(|\frac{p_1+p_2}{2}-p_1|+1)\cdot\delta^{|\frac{p_1+p_2}{2}-p_1|} \\
&- \frac{1}{2}\cdot p(|\frac{p_{n_c}+p_1}{2}-p_1|+1)\cdot\delta^{|\frac{p_{n_c}+p_1}{2}-p_1|} - \frac{1}{2}\cdot p(|\frac{p_1+p_2}{2}-p_2|+1)\cdot\delta^{|\frac{p_1+p_2}{2}-p_2|} \\
&+ \frac{1}{2}\cdot p(|\frac{p_{n_c}+p_1}{2}-p_{n_c}|+1)\cdot\delta^{|\frac{p_{n_c}+p_1}{2}-p_{n_c}|}) \\
&= a\cdot(\frac{1}{2}\cdot p(|\frac{p_1+p_2}{2}-p_2|+1)\cdot\delta^{|\frac{p_1+p_2}{2}-p_2|} - \frac{1}{2}\cdot p(|\frac{p_{n_c}+p_1}{2}-p_{n_c}|+1)\cdot\delta^{|\frac{p_{n_c}+p_1}{2}-p_{n_c}|})
\end{aligned} \tag{11}
$$

We derive Case A as follows:

$$\frac{\partial}{\partial p_1}\int_{p_1\cdot d+e}^{p_1\cdot b+c} f(h,p_1)\ \mathrm{dh} = \underbrace{\frac{\partial}{\partial p_1}\int_{0}^{p_1\cdot b+c} f(h,p_1)\ \mathrm{dh}}_{\text{We follow this case only.}} - \frac{\partial}{\partial p_1}\int_{0}^{p_1\cdot d+e} f(h,p_1)\ \mathrm{dh} \tag{12}$$

The first part of Equation (12) can be derived as follows:

$$\begin{aligned}
&\frac{\partial}{\partial p_1}\int_{0}^{p_1\cdot b+c} f(h,p_1)\ \mathrm{dh}\\
&= \lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0+\Delta x)\ \mathrm{dh} - \int_0^{x_0\cdot b+c} f(h,x_0)\ \mathrm{dh}}{\Delta x}\\
&= \lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0+\Delta x)\ \mathrm{dh} - \int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0)\ \mathrm{dh} - \int_0^{x_0\cdot b+c} f(h,x_0)\ \mathrm{dh} + \int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0)\ \mathrm{dh}}{\Delta x}\\
&= \lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0+\Delta x)\ \mathrm{dh} - \int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0)\ \mathrm{dh}}{\Delta x}\\
&+ \lim_{\Delta x\to 0}\frac{-\int_0^{x_0\cdot b+c} f(h,x_0)\ \mathrm{dh} + \int_0^{(x_0+\Delta x)\cdot b+c} f(h,x_0)\ \mathrm{dh}}{\Delta x}\\
&= \underbrace{\lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c}(f(h,x_0+\Delta x) - f(h,x_0))\ \mathrm{dh}}{\Delta x}}_{\text{We follow this case only.}} + \underbrace{\lim_{\Delta x\to 0}\frac{\int_{x_0\cdot b+c}^{(x_0+\Delta x)\cdot b+c} f(h,x_0)\ \mathrm{dh}}{\Delta x}}_{b\cdot f(p_1\cdot b+c,p_1)}
\end{aligned} \tag{13}$$

Let $\epsilon \in [x_0, x_0+\Delta x]$ hold. Then, the first summand of Equation (13) is:

$$\begin{aligned}
&\lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c}(f(h,x_0+\Delta x) - f(h,x_0))\ \mathrm{dh}}{\Delta x} = \lim_{\Delta x\to 0}\frac{\int_0^{(x_0+\Delta x)\cdot b+c} f'(h,\epsilon)\Delta x\ \mathrm{dh}}{\Delta x}\\
&= \lim_{\Delta x\to 0}\int_0^{(x_0+\Delta x)\cdot b+c}\frac{\partial f}{\partial x}(h,\epsilon)\ \mathrm{dh} = \int_0^{p_1\cdot b+c}(\frac{\partial}{\partial p_1} f(h,p_1))\ \mathrm{dh}
\end{aligned} \tag{14}$$

To sum up, the derivative of the first summand is:

$$\frac{\partial}{\partial p_1}\int_{p_1\cdot d+e}^{p_1\cdot b+c} f(h,p_1)\ \mathrm{dh} = \int_{p_1\cdot d+e}^{p_1\cdot b+c}\frac{\partial}{\partial p_1} f(h,p_1)\ \mathrm{dh} + b\cdot f(p_1\cdot b+c,p_1) - d\cdot f(p_1\cdot d+e,p_1) \tag{15}$$

Case B from Equation (11) can be derived as follows:

$$\frac{\partial}{\partial p_1}\int_{p_1\cdot b+c}^{a} f(h)\ \mathrm{dh} = \frac{\partial}{\partial p_1}(F(a) - F(p_1\cdot b+c)) = \underbrace{\frac{\partial}{\partial p_1}F(a)}_{=0} - \frac{\partial}{\partial p_1}F(p_1\cdot b+c) = -b\cdot f(p_1\cdot b+c) \tag{16}$$

We calculate Case C from Equation (11) as follows:

$$\frac{\partial}{\partial p_1}\int_{a}^{p_1\cdot b+c} f(h)\ \mathrm{dh} = \frac{\partial}{\partial p_1}(F(p_1\cdot b+c) - F(a)) = \frac{\partial}{\partial p_1}F(p_1\cdot b+c) - \underbrace{\frac{\partial}{\partial p_1}F(a)}_{=0} = b\cdot f(p_1\cdot b+c) \tag{17}$$

In analogy the gradient, i.e., $\frac{\partial E(p_1,\ldots,p_{n_c})}{\partial p_1,\ldots,\partial p_{n_c}}$, can be calculated. It is:

$$
\begin{pmatrix}
\frac{1}{2} aC^{1+\left|\frac{p_1+\mathrm{p2}}{2}-p_1\right|} d^{\left|\frac{p_1+\mathrm{p2}}{2}-p_1\right|} - \frac{1}{2} aC^{1+\left|\frac{p_1+\mathrm{p2}}{2}-\mathrm{p2}\right|} d^{\left|\frac{p_1+\mathrm{p2}}{2}-\mathrm{p2}\right|} \\
-\frac{1}{2} aC^{1+\left|-p_1+\frac{p_1+p_{n_c}}{2}\right|} d^{\left|-p_1+\frac{p_1+p_{n_c}}{2}\right|} + \frac{1}{2} aC^{1+\left|-p_{n_c}+\frac{p_1+p_{n_c}}{2}\right|} d^{\left|-p_{n_c}+\frac{p_1+p_{n_c}}{2}\right|} \\
+\int_{\frac{p_1+p_{n_c}}{2}}^{\frac{p_1+\mathrm{p2}}{2}} \left(-\frac{aC^{1+|h-p_1|} d^{|h-p_1|}(h-p_1)\mathrm{Log}[C]}{|h-p_1|} - \frac{aC^{1+|h-p_1|} d^{|h-p_1|}(h-p_1)\mathrm{Log}[d]}{|h-p_1|}\right) dh, \\
\ldots \\
\frac{1}{2} aC^{1+\left|-p_1+\frac{p_1+p_{n_c/2-1}}{2}\right|} d^{\left|-p_1+\frac{p_1+p_{n_c/2-1}}{2}\right|} - \frac{1}{2} aC^{1+\left|-p_{n_c/2-1}+\frac{p_1+p_{n_c/2-1}}{2}\right|} d^{\left|-p_{n_c/2-1}+\frac{p_1+p_{n_c/2-1}}{2}\right|} \\
+\frac{1}{2} aC^{1+\left|-p_{n_c/2-1}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} d^{\left|-p_{n_c/2-1}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} - \frac{1}{2} aC^{1+\left|-p_{n_c/2}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} d^{\left|-p_{n_c/2}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} \\
+\int_{\frac{p_1+p_{n_c/2-1}}{2}}^{\frac{p_{n_c/2-1}+p_{n_c/2}}{2}} \left(-\frac{aC^{1+|h-p_{n_c/2-1}|} d^{|h-p_{n_c/2-1}|}(h-p_{n_c/2-1})\mathrm{Log}[C]}{|h-p_{n_c/2-1}|} - \frac{aC^{1+|h-p_{n_c/2-1}|} d^{|h-p_{n_c/2-1}|}(h-p_{n_c/2-1})\mathrm{Log}[d]}{|h-p_{n_c/2-1}|}\right) dh, \\
\frac{1}{2} aC^{1+\left|\frac{1}{2}(-1+p_{n_c/2})-p_{n_c/2}\right|} d^{\left|\frac{1}{2}(-1+p_{n_c/2})-p_{n_c/2}\right|} + \frac{1}{2} aC^{1+\left|-p_{n_c/2-1}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} d^{\left|-p_{n_c/2-1}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} \\
-\frac{1}{2} aC^{1+\left|-p_{n_c/2}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} d^{\left|-p_{n_c/2}+\frac{p_{n_c/2-1}+p_{n_c/2}}{2}\right|} \\
+\int_{\frac{p_{n_c/2-1}+p_{n_c/2}}{2}}^{\frac{1}{2}(-1+p_{n_c/2})} \left(-\frac{aC^{1+|h-p_{n_c/2}|} d^{|h-p_{n_c/2}|}(h-p_{n_c/2})\mathrm{Log}[C]}{|h-p_{n_c/2}|} - \frac{aC^{1+|h-p_{n_c/2}|} d^{|h-p_{n_c/2}|}(h-p_{n_c/2})\mathrm{Log}[d]}{|h-p_{n_c/2}|}\right) dh \\
+\frac{a\left(-\frac{1}{2}C^{p_{n_c/2}/2} d^{p_{n_c/2}/2}\mathrm{Log}[C] - \frac{1}{2}C^{p_{n_c/2}/2} d^{p_{n_c/2}/2}\mathrm{Log}[d]\right)}{\sqrt{C}\sqrt{d}(\mathrm{Log}[C]+\mathrm{Log}[d])}, \\
-\frac{1}{2} aC^{1+\left|-p_{n_c/2+1}+\frac{1+p_{n_c/2+1}}{2}\right|} d^{\left|-p_{n_c/2+1}+\frac{1+p_{n_c/2+1}}{2}\right|} + \frac{1}{2} aC^{1+\left|-p_{n_c/2+1}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} d^{\left|-p_{n_c/2+1}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} \\
-\frac{1}{2} aC^{1+\left|-p_{n_c/2+2}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} d^{\left|-p_{n_c/2+2}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} \\
+\int_{\frac{1+p_{n_c/2+1}}{2}}^{\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}} \left(-\frac{aC^{1+|h-p_{n_c/2+1}|} d^{|h-p_{n_c/2+1}|}(h-p_{n_c/2+1})\mathrm{Log}[C]}{|h-p_{n_c/2+1}|} - \frac{aC^{1+|h-p_{n_c/2+1}|} d^{|h-p_{n_c/2+1}|}(h-p_{n_c/2+1})\mathrm{Log}[d]}{|h-p_{n_c/2+1}|}\right) dh \\
+\frac{a\left(\frac{1}{2}C^{1+\frac{p_{n_c/2+1}}{2}} d^{1+\frac{p_{n_c/2+1}}{2}}\mathrm{Log}[C] + \frac{1}{2}C^{1+\frac{p_{n_c/2+1}}{2}} d^{1+\frac{p_{n_c/2+1}}{2}}\mathrm{Log}[d]\right)}{\sqrt{C}\sqrt{d}(\mathrm{Log}[C]+\mathrm{Log}[d])}, \\
\frac{1}{2} aC^{1+\left|-p_{n_c/2+1}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} d^{\left|-p_{n_c/2+1}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} - \frac{1}{2} aC^{1+\left|-p_{n_c/2+2}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} d^{\left|-p_{n_c/2+2}+\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}\right|} \\
+\frac{1}{2} aC^{1+\left|-p_{n_c/2+2}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} d^{\left|-p_{n_c/2+2}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} - \frac{1}{2} aC^{1+\left|-p_{n_c}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} d^{\left|-p_{n_c}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} \\
+\int_{\frac{p_{n_c/2+1}+p_{n_c/2+2}}{2}}^{\frac{p_{n_c/2+2}+p_{n_c}}{2}} \left(-\frac{aC^{1+|h-p_{n_c/2+2}|} d^{|h-p_{n_c/2+2}|}(h-p_{n_c/2+2})\mathrm{Log}[C]}{|h-p_{n_c/2+2}|} - \frac{aC^{1+|h-p_{n_c/2+2}|} d^{|h-p_{n_c/2+2}|}(h-p_{n_c/2+2})\mathrm{Log}[d]}{|h-p_{n_c/2+2}|}\right) dh, \\
\ldots \\
-\frac{1}{2} aC^{1+\left|-p_1+\frac{p_1+p_{n_c}}{2}\right|} d^{\left|-p_1+\frac{p_1+p_{n_c}}{2}\right|} + \frac{1}{2} aC^{1+\left|-p_{n_c}+\frac{p_1+p_{n_c}}{2}\right|} d^{\left|-p_{n_c}+\frac{p_1+p_{n_c}}{2}\right|} \\
+\frac{1}{2} aC^{1+\left|-p_{n_c/2+2}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} d^{\left|-p_{n_c/2+2}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} - \frac{1}{2} aC^{1+\left|-p_{n_c}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} d^{\left|-p_{n_c}+\frac{p_{n_c/2+2}+p_{n_c}}{2}\right|} \\
+\int_{\frac{p_{n_c/2+2}+p_{n_c}}{2}}^{\frac{p_1+p_{n_c}}{2}} \left(-\frac{aC^{1+|h-p_{n_c}|} d^{|h-p_{n_c}|}(h-p_{n_c})\mathrm{Log}[C]}{|h-p_{n_c}|} - \frac{aC^{1+|h-p_{n_c}|} d^{|h-p_{n_c}|}(h-p_{n_c})\mathrm{Log}[d]}{|h-p_{n_c}|}\right) dh
\end{pmatrix}
\tag{18}
$$

For optimal positions the gradient is zero. For simplification we assume that the number of additional contacts is odd. The position of one of the contacts is on the other side of the ring, i.e., $p_{n_c} = \frac{n-1}{2}$. Every other contact with position p has a sibling with position $n-p$ on the other side of the ring. For $i = 1 \ldots \frac{|K^+|-1}{2}$ and $j = \frac{|K^+|-1}{2} \ldots 1$ the optimal positions are as follows:

- For p_i:

$$\frac{n-1}{2} \cdot \frac{2 \cdot i}{|K^+|+1} + \frac{2 \cdot j}{|K^+|+1} - \frac{4 \cdot j}{|K^+|+1} \cdot \frac{ln(\frac{1+\delta}{2})}{ln(C \cdot \delta)}$$

- For $p_{i+\frac{|K^+|-1}{2}}$:

$$n - \left(\frac{n-1}{2} \cdot \frac{2 \cdot i}{|K^+|+1} + \frac{2 \cdot j}{|K^+|+1} - \frac{4 \cdot j}{|K^+|+1} \cdot \frac{ln(\frac{1+\delta}{2})}{ln(C \cdot \delta)} \right)$$

We checked all calculations using symbolic calculation tools.

B. Appendix – Proof of Lemma 4.1

In this section we investigate efficient equilibria for contact relations.
In the inefficient network agents do not send requests. Here, we assume that both players send requests ($sendRequest = true$), i.e., are not dropouts. Still it is unclear if they cooperate (and how much) (cf. parameter $ownCoop$) and how much they expect from their contact (cf. parameter $contactC$).

Following the definition of an equilibrium we look at the payoff of Player i and j. If both cannot increase their own payoff by changing their strategy, they are in an equilibrium. Note that we assume that agents understand that the inefficient network leads to no payoff, and that they avoid strategies that lead to an inefficient network.

We have three cases: Player i and j fulfill the expectation of each other (Case (1)), only one fulfills the others expectation (Case (2)), as well as Player i and Player j do not cooperate as much as their contact demands (Case (3)).

Case (1): The degree of cooperation of Player i ($ownCoop_i$) is higher than the expectation of Player j, and vice versa. Then the following formulas hold:

$$isCooperative(i, additionalCC_j) \tag{19}$$

$$isCooperative(i, contactC_j) \tag{20}$$

Further, the income that Agent i gains from Contact j has to be higher than the expenditures:

$$a \cdot |A_{i\to j}| > q \cdot |R_{i\to j}| + c \cdot r(i) + (w \cdot |W_{j\to i}| + f \cdot |F_{j\to i}|) \cdot ownCoop_i \tag{21}$$

We use two estimations to simplify Formula 21. First, Agent i does not receive more answers for requests that it forwarded to Contact j than j processes, i.e., $|A_{i\to j}| \leq |R_{i\to j}| \cdot ownCoop_j$. Second, $c \leq c \cdot |R_{i\to j}|$, i.e., Agent j is only an additional contact of Agent i, if i forwards at least one own request to j. It follows:

$$a \cdot |R_{i\to j}| \cdot ownCoop_j > q \cdot |R_{i\to j}| + c \cdot |R_{i\to j}| \cdot r(i) + (w \cdot |W_{j\to i}| + f \cdot |F_{j\to i}|) \cdot ownCoop_i \tag{22}$$

Lemma 4.1 directly follows from Equations (19), (20) and (22).

In Case (2) the degree of cooperation of Player i is lower than the expectation of Player j ($isCooperative(i, contactC_j)$ and/or $isCooperative(i, additionalCC_j)$), but the degree of cooperation of Player j is higher than the expectation of Player i ($isCooperative(j, contactC_i)$ and $isCooperative(j, additionalCC_i)$). Thus, Player j drops Player i or its requests, and consequently Player i does not have any income: $E[income(i)] = 0$. Instead Player i has costs if i does not choose Strategy Dropout. It follows that Strategy Dropout is the equilibrium.

In Case (3) the degree of cooperation of Player i is lower than the expectation of Player j ($isCooperative(i, contactC_j)$ and/or $isCooperative(i, additionalCC_j)$), and vice versa:

$$E[income(i)] = 0 \wedge E[expenditure(i)] = q \cdot |R_i| \tag{23}$$

In this case Strategy Dropout is the equilibrium. As we have seen, only Case (1) leads to a cooperative equilibrium.

C. Appendix – Prof of Lemma 4.3

Proof: (I) The expected benefit of a request that has been forwarded over $\beta \cdot log^2(n)$ *hops is:*

$$p(\beta \cdot log^2(n)) \cdot \delta^{\beta \cdot log^2(n)} \cdot a - q$$

(II) The expected benefit when forwarding it one additional hop further is:

$$p(1 + \beta \cdot log^2(n)) \cdot \delta^{1+\beta \cdot log^2(n)} \cdot a - q$$

The difference of (I) and (II) is:

$$a \cdot \overline{C}^{\beta \cdot log^2(n)} \cdot \delta^{\beta \cdot log^2(n)} \cdot (1 - \overline{C} \cdot \delta)$$

The increased payoff follows from one additional forwarder less that might drop the request ($\overline{C}$)*, and that would prolong the processing* (δ)*. Lemma 4.3 follows.* □

D. Appendix – Prof of Lemma 4.4

Proof: (I) The expected benefit of a request that has been forwarded over $\beta \cdot log^2(n)$ hops is:

$$p(\beta \cdot log^2(n)) \cdot \delta^{\beta \cdot log^2(n)} \cdot a - q$$

(II) The expected benefit when forwarding it $\epsilon > 0$ additional hops further is:

$$p(\epsilon + \beta \cdot log^2(n)) \cdot \delta^{\epsilon + \beta \cdot log^2(n)} \cdot a - q$$

The difference of (I) and (II) is:

$$a \cdot \overline{C}^{\beta \cdot log^2(n)} \cdot \delta^{\beta \cdot log^2(n)} \cdot (1 - \overline{C}^{\epsilon} \cdot \delta^{\epsilon})$$

Lemma 4.4 follows. □ Note that, in the presence of a discount factor or of uncooperative agents, the previous term has a value that is positive, since $1 - \overline{C}^{\epsilon} \cdot \delta^{\epsilon} \in (0, 1]$. Thus, an agent exchanges a contact with non-optimal additional contacts against an agent with Kleinberg-distributed additional contacts.

E. Appendix – Simulations

What are evolutionarily stable equilibria in information systems that allow nodes to choose their actions and their contacts? This is the main question for system designers that seek for efficient networks. If the equilibrium is not efficient, mechanism design is necessary to change the situation, otherwise costly efforts are not necessary.

Answering this question is difficult. Simulations are only meaningful if their design does not imply the result. Simulations of evolutionary processes avoid this effect: They use a well known algorithm, i.e., evolution, and treat the strategy choices of agents as a genome. Thus, the system designer does not make any assumptions about the strategies or the distribution of strategies in the system, but about the actions an agent can take principally. Theoretical analyses, on the other hand, are more precise, but often lead to formulas that cannot be solved analytically.

In this section we have two contributions: (I) We validate the identified equilibria experimentally. (II) We show that (partial) cooperation arises from a network of agents with randomly chosen strategies. Note that the second contribution is different from the first one: To show that a state is an equilibrium, all agents start already with the equilibrium strategy (and then have no incentive to leave it); the second contribution is different: agents start with a random strategy, and evolve. Thus, the second contribution shows likely equilibria.

Evolutionary Stable Evolutionarily Stable Strategies (ESS) are a subset of Nash equilibria [29]. Still, they have another interpretation. Strategies are described by a gene G. A strategy S(G) is an ESS if it cannot be invaded by mutation of itself S(G'). An invasion is defined as follows: A strategy S(G') invades strategy S(G) if its expected payoff is higher. We use this interpretation of Nash equilibria in Appendix E where we investigate different strategies under an evolutionary process.

Game Setup We believe agents in a decentralized search system act as described in the following game:
Players play rounds, i.e., time is discrete: (1) Each agent can issue one request per round (but is not forced to do so) and send it to one of his contacts. (2) At the beginning of each round, an Agent i obtains a set of requests issued or forwarded by agents which have i as a contact. Agent i decides for each request whether to drop or process (i.e., forward or answer) it. (3) An agent can drop additional contacts at his discretion and can select players as contacts with whom it had interacted before. For instance, Agent i can select Agent j as an additional contact if j had issued at least one request that has reached Agent i. Selecting and dropping is possible in each round.

General Setup We simulate 500 agents in a dynamic network, i.e., we assume that strategies that lead to less payoff than other strategies have a higher chance of dying out, and that new strategies come up with a certain probability. The motivation is that agents/humans in decentralized search systems use strategies that are useful to them, i.e., that lead to high payoffs. Thus, after 1,000 rounds in the game, we start a new meta-round: Strategies with a high payoff $z(i)$ remain, unsuccessful strategies die out. More precisely, we assume an evolutionary process with (a) a single point crossover and a crossover-rate of 20%, and (b) a mutation probability of 10%. (a) means that out of 500 agents in a system, 100 combine their strategy, i.e., split their strategy-vector at a random position and exchange one fraction. (b) specifies that 50 agents out of 500 randomly change their strategy. The intuition behind this is that some agents have new ideas about how they should behave. Note that it is equivalent whether agents change their strategy or whether we exchange agents against new agents with a new strategy: No agent keeps a history about events in a previous meta-round. This is natural, since these events are the result of a distribution of strategies that is no longer valid.

Location In contrast to our theoretic analysis, we do not assume that an agent chooses the closest peer to a destination of a request when forwarding. Instead we relax this assumption: Agents can choose any contact that is closer to the destination than they are. When forwarding a request, there may be a tradeoff between the position of a contact and its degree of cooperation. For example a medium cooperative contact that is very close to the destination might be a better choice than a very cooperative contact that is far away. Thus, strategy-space dimension $locationWeight$ defines how important the location is compared to the degree of cooperation. E.g., $locationWeight = 1$ means that an agent chooses the closest contact no matter how uncooperative it is. Algorithm 3 shows how an agent chooses a contact for forwarding a request:

Algorithm 3: Influence of $locationWeight$

Input: List of Contacts L, RequestDestination d, LocationWeight LW

1 max(Δ) = $max_{c \in L}$ Δ(c, d); min(Δ) = $min_{c \in L}$ Δ(c, d);
2 max(C) = $max_{c \in L}$ C(c); min(C) = $min_{c \in L}$ C(c);
3 double rating = -1.0; Contact contact = null;
4 **foreach** *Contact* $c \in L$ **do**
5   double r = $\frac{max(\Delta) - \Delta(c,d)}{max(\Delta) - min(\Delta)} \cdot LW + \frac{C(c) - min(C)}{max(C) - min(C)} \cdot (1 - LW)$;
6   **if** $r > rating$ **then**
7   rating = r; contact = c;
8   **end**
9 **end**
10 return contact;

Dependent on parameter $locationWeight$ it computes a rating for each contact. Algorithm 3 identifies the contact with the largest (smallest) distance to the destination and stores this distance (Line 1), analogously for the degree of cooperation (Line 2). The algorithm then computes a rating for each contact (Lines 4-7). At the end, the agent forwards the request to the contact with the highest rating.

Exogenous Parameters Next to dynamic strategies, we varied exogenous parameters: (a) the cost of maintaining contacts c, (b) the reliability of information on the degree of cooperation of other agents, i.e., $reportingError$, and (c) the discount factor δ. There are further exogenous parameters: the cost factors for forwarding and answering requests as well as receiving responses (cf. Table I). We do not change these, here, because we focus on the relation between the cost of maintaining contacts and the cost of processing requests, and not on the costs themselves. In other words, we vary the relation between cost of contacts and processing costs, i.e., change the costs for contacts and leave processing costs constant.

Base-Line Set We start with a base-line set of exogenous parameters. Later, we will describe how the results change if a parameter changes. The base-line set is as follows: For each additional contact an agent loses 1 point per round ($c = 1$). When determining the degree of cooperation of a contact, an agent has a reporting error of 5% ($reportingError = 0.05$). If it still receives an answer, a discount factor of 90% comes into the play ($\delta = 0.9$).

Payoff Factors In the internet, forwarding, answering, issuing requests as well as maintaining contacts are cheap operations. For instance, maintaining 1,000 contacts is not problematic for a modern pc. A user, on the other hand, is happy, i.e., receives a high benefit, for each answer to its requests. Thus, we have chosen a high benefit for receiving an answer. It is 1,000 points in our experiments. The cost of issuing a request is 2 points, cost of forwarding 1 point, and the cost of answering a request 5 points.

Contact Distribution Since we know from the previous section that agents choose a Kleinberg distribution, an agent i selects a contact j with probability $d(i,j)^{-r}$ (and selects it if parameter $maxC$ and $additionalCC$ allow to do so) in our simulations. For an illustration of the effect of parameter r see Figure 4. It shows $d(i,j)^{-r}$ for common values of r: The higher r, the faster decreases $d(i,j)^{-r}$. This means that agents with a higher r prefer close contacts over distant ones.

1) Equilibrium – Results: In this section, we show that cooperation is an equilibrium, i.e., we are in Case (I).

Initial Strategy Agents use the following strategy at the begin of the evolutionary process: Agents participate and cooperate

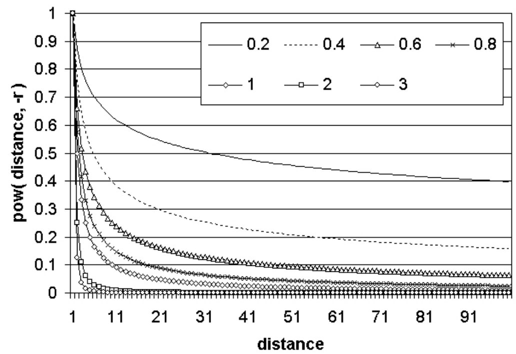

Figure 4. Impact of Parameter r

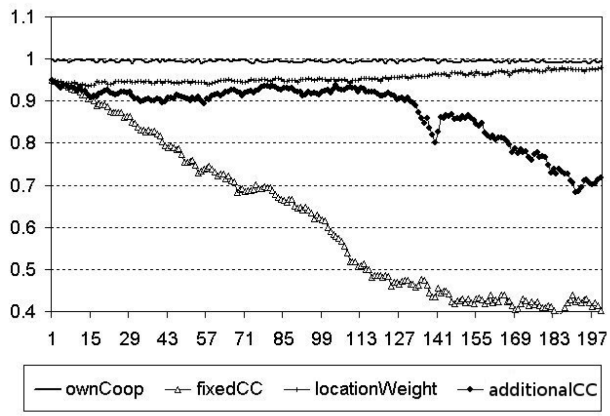

Figure 5. Action-Selection Strategy

($sentRequest = true$, $ownCoop = 100\%$), they use a threshold-strategy and defect on contacts that have a degree of cooperation of less than 95% ($contactC = additionalCC = 95\%$), the while issuing/forwarding agents overweight the location of the next receiver over its degree of cooperation ($locationWeight = 95\%$), agents have five additional contacts and weight them by their distance to them ($maxC = 5$, $r = 1$).

Baseline-Set Figures 5 and 6 illustrate the results from the baseline-set. Figure 5 shows action-selection parameters, and Figure 6 the additional-contact distribution. Both show the average over all agents. For Figure 6 we have grouped an agent's additional contacts by their distance (to the agent) into eight equidistant classes. The class that is closest to the agent (B1) is shown on the bottom of Figure 6, the next closest (B2) is the one above, and so forth. For these plots alone, we simulated 100,000 agents issuing 100,000,000 requests. Note that, although we show only a representative result here, we repeated all simulation 25 times.

We start with describing action-selection choices (cf. Figure 5): Agents participate ($sendRequests = 1$, $z(i) > 0$, both not shown in the figures), have a high degree of cooperation ($ownCoop \approx 1$), and use high thresholds for their contacts ($contactC \approx 0.5$, $additionalCC \approx 0.85$). The $contactC$-threshold is insignificant: In all simulations it had a large variance (whereas all other results are representative). An explanation is that an agent with a lot of additional contacts does not rely on its fixed contacts. When forwarding requests, agents weight the location of a contact over its degree of cooperation ($locationWeight \approx 0.95$). This is natural, since they have cooperative contacts only.

On the link-selection choice, we have the following results (cf. Figure 6): Agents have 40 additional contacts. At the beginning of the dynamic process, agents favor close contacts: In meta-round 37, an agent has twenty additional contacts close to its own position ($B1$), but only five in meta-round 200. At the end of the measurement, agents distribute their additional contacts almost uniformly: $B1$ to $B8$ are almost the same. One exception is that agents prefer close or (outmost) distant contacts a little bit more than random ones. This is similar to a Kleinberg distribution.

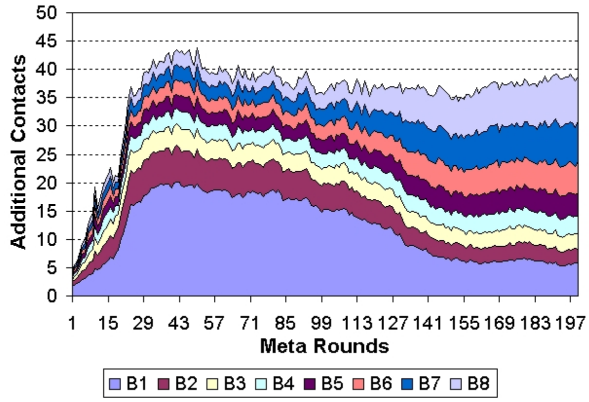

Figure 6. Additional-Contact Distribution

Contact-Maintenance Costs We now look at each exogenous parameter and investigate its influence on the equilibrium states. We start with the cost of maintaining contacts c. We vary the parameter in 10 steps from $c = 0$, $c = 0.5$, ... to $c = 4.5$.
1. The higher the costs, the less additional contacts are selected. Whereas an agent connects to many agents if contact cost are low $c = 0.5 \Rightarrow maxC > 60$, an agent has only around 25 contacts for $c = 2$, and around 10 for $c = 4$.
2. The less contacts cost, the more do agents expect from their additional contacts. Threshold $additionalCC$ is around 83% for $c = 0$, and about 90% for $c = 0.5$. For higher contact-maintenance costs the expectation falls to 81% ($c = 2$), 72% ($c = 3$), and to 70% for $c = 4$.

Reporting Error Next to the cost of maintaining contacts, we look at the reliability of information on the degree of cooperation of other agents, i.e., the $reportingError$. Clearly, agents cannot know the behavior of other agents in reality

Algorithm 4: isCooperative(Agent i, Threshold t)

Input: Agent i, Threshold t
Output: $\{$TRUE, FALSE$\}$

```
double trueCooperation = C(i);
double offset = ((getRandomValue[0,1] - trueCooperation) · reportingError);
double adjustedCooperation = trueCoop + offset;
return adjustedCooperation ≥ t;
```

without making mistakes. There is an unreliability factor that models this: Parameter $reportingError \in [0, 1]$ describes how reliable information on the degree of cooperation of other agents is. In contrast to the simple mechanism used in Section III-C, we use $reportingError$ as described in Algorithm 4: We add an offset that differs by the $reportingError$ parameter at most. The higher it is, the less are agents able to determine the degree of cooperation of other agents. We vary it in 5 steps from 0%, 5%, ... to 20%. The results are as follows:
1. If the reporting error is too high, the network breaks down, i.e., all agents use strategy 'Dropout'. This especially happens when $reportingError$ is greater than 15%. (In the following, we only mention results from systems that did *not* break down.)
2. In systems with a higher reporting error, agents tend to select more additional contacts.
3. The higher the reporting error, the more do agents favor close additional contacts and distant contacts. The first effect is more distinctive than the second, i.e., agents have more close contacts than distant ones.
4. The degree of cooperation slightly decreases if the reporting error rises (until the network breaks down).

We have seen that a low reporting error is crucial for efficiency. The lower it is, the more do agents cooperate, and the less additional contacts are necessary. The distribution of agents changes from almost uniform to a distribution that favors close contacts, and that slightly favors distant agents. One explanation is that agents need close contacts to forward requests on behalf of others effectively, and distant agents to improve the processing of their own requests.

Discount Factor The last exogenous parameter we vary is the discount factor δ. The lower it is, the more does the benefit

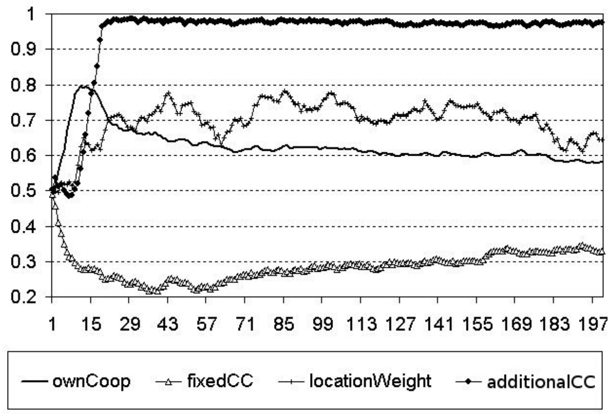

Figure 7. Action-Selection Strategy

for receiving an answer decrease per forwarder. We set δ to $0.5, 0.55, 0.6, ..., 0.95$ and 1. The results are:
1. Low discount factors (but still greater than 50%) do *not* lead to broken systems. This is surprising, since for a discount factor of 50% the benefit of receiving an answer decreases 50% per hop.
2. A low discount factor leads to more additional contacts. This is natural, since less hops means more benefit.
3. In a system with a discount factor of 100%, i.e., the value of an answer does not depend on the number of hops, agents choose only few additional contacts ($maxC \approx 5$).

From the three exogenous parameters we have investigated, each influences evolutionary equilibria in a different way. The cost of maintaining contacts influences the number of contacts per agent as well as the expectations in the degree of cooperation of contacts. The reporting error influences the degree of cooperation, and is crucial for efficient systems. The discount factor has the lowest impact on evolutionarily stable equilibria. Still, it can change the average number of contacts per agent.

2) Emerging Cooperation: Unlike the previous section, we do not assume any initial strategy here: Every agent flips its genome-bits randomly at the beginning of the first meta-round. Later, the evolutionary process selects and evolves successful strategies.

The question is now if cooperation can arise out of random strategies. For much simpler settings, e.g. iterated prisoner's dilemma, it is known that cooperation can arise [5]. In contrast to [5] we do not use human-written strategies, but random ones. Further, link-selection and intermediaries in the forwarding chain make the problem much harder [1], [2], and less predictable.

Answering this question is important beyond computer science, since it gives an explanation for human cooperation in similar settings, for instance, cooperation among friends, or even societies. Note that altruism does not play a role here. If the answer is 'yes', cooperation emerges out of self-interest. Note that a single simulation that leads to cooperation is sufficient to prove that cooperation can emerge.

The short answer to this research question is 'yes'. Still, there are some differences to the results from the previous section that we state out now.

Cooperation Agents show a lower degree of cooperation than in Case (I). Figure 7 shows average action-selection choices for the base-line parameter-set (cf. Figure 5). In contrast to Case (I) the average degree of cooperation ($ownCoop$) is lower, i.e., around 50%. Even though the degree of cooperation is not efficient, agents cooperate, i.e., they generate positive payoff for other agents. Further, their expectation in additional contacts is higher, i.e., almost 100% (cf. $additionalCC$). One interesting result is that the stability of the system is much more robust against errors in the information on the degree of cooperation of other agents: Systems with a reporting error of 60% or less do not break down. One reason for this robustness might be the heterogeneity of the strategies.

Link-Selection Agents tend to select more additional contacts than in Case (I). Figure 8 shows average link-selection choices for the base-line parameter-set: Agents choose around 100 additional contacts. This makes sense, since the degree of cooperation in the system is lower than in the efficient system. The distribution of additional contacts is as follows: Agents

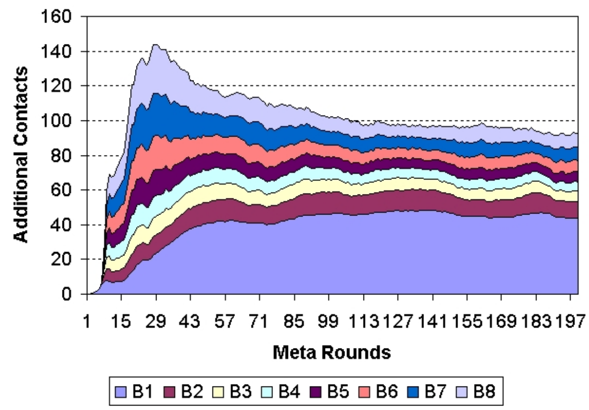

Figure 8. Additional-Contact Distribution

prefer close contacts over random ones. They almost meet a Kleinberg distribution.

Discount Factor In contrast to the Case (I), the discount factor plays a much greater role: Systems with a discount factor of less than 85% tend to break down. It seems to be important for the emergence of cooperation that agents benefit from their requests even in systems with a random strategy. If the discount factor is too large, this is not true. Whereas the discount factor influences action-selection strategies, it does not have a significant influence on link-selection.

Discussion Our simulations prove that the cooperation can arise out of self-interested agents with random strategies. This is an important result, since it gives an explanation why networks that feature action and link selection are actually cooperative. Many systems meet these conditions, and we now have a model that can explain cooperation in these systems.

F. Appendix – Reporting Error

Modern networks use reputation systems to increase cooperation and social welfare in the system. We assume that a reputation system exists and that it reports whether the degree of cooperation $C(i)$ of an agent i is lower than a threshold t. Further, we assume that these reports are wrong in $reportingError\%$ of the cases (cf. Algorithm 1).

Now we analyze if the cooperative equilibrium is stable depending on exogenous parameter $reportingError$? To answer this question, we start with four observations. Keep in mind that $t = 1.0$ holds in the cooperative equilibrium.

1) Cooperative agents lose $reportingError\%$ of their additional contacts.
2) Uncooperative agents lose $1 - reportingError\%$ of their additional contacts.
3) Cooperative agents lose about $\left(1 - (1 - reportingError)^{\overline{h}}\right)\%$ of their requests.
4) Uncooperative agents lose about $\left(1 - (reportingError)^{\overline{h}}\right)\%$ of their requests.

If having additional contacts is beneficial at all, then Observation 1) leads to a higher payoff than Observation 2), if $reportingError < 0.5$ holds. The same holds, for Observations 3) and 4). Consequently, for $reportingError < 0.5$ the cooperative equilibrium is stable, and the inefficient network otherwise.

Obviously, a system can only be efficient, if parameter $reportingError$ is zero. Otherwise, agents drop (some) requests of cooperative agents because they deem them uncooperative.

G. Appendix – Utility of DSC

In this paper, we propose that agents use the DSC Strategy. In this appendix, we show that it is beneficial for the agents to do so.

We assume that the system is in the cooperative equilibrium, i.e., additional contacts cooperate to the fullest extent. If an agent i has the possibility to exchange a contact s with an agent f that processes requests faster, i.e., $\overline{h}_\triangle(i|f) < \overline{h}_\triangle(i|s)$, then DSC would propose to do so, and not using DSC the opposite. While the cost of maintaining s or f are equal, the benefit incurred by Agent f is higher than the benefit incurred by Agent s due to the discount factor: Since the average number of hops for requests $req \in A_{i\to f}$ forwarded to Agent f is smaller than for those forwarded to s ($A_{i\to s}$) the benefit is higher, i.e., the following formula holds:

$$a \cdot \sum_{f_r eq \in A_{i \to f}} \delta^{h(f_r eq)} < a \cdot \sum_{s_r eq \in A_{i \to s}} \delta^{h(s_r eq)} \tag{24}$$