PLOS Complex Systems

RESEARCH ARTICLE

The complexity of problem-solving human social systems: Structural vs dynamic complexity

Michael Roos*

Department of Management and Economics, Ruhr-Universität Bochum, Bochum, Germany

* Michael.Roos@rub.de

Abstract

Research in the social sciences often describes social complexity through a combination of structure, organization, and behavior within human social systems. In this paper, I argue that these aspects, while important, are conceptually distinct. Specifically, I distinguish between structural complexity—the organizational properties of a system—and dynamic complexity—the patterns of behavior and interaction within the system. To illustrate this distinction, I present three agent-based models of collective problem-solving: a hierarchical model, a random network model, and a hybrid of the two. These models are used to demonstrate how different forms of complexity can be measured and how they affect system performance. Several metrics are proposed to quantify structural and dynamic complexity, and model simulations show that the structurally complex hierarchical model is more efficient at solving problems than the dynamically complex network model. The simulations confirm the widespread intuition that systems with high structural complexity are effective for solving known problems, while systems with high dynamic complexity are more flexible. However, I also show that the hierarchical model is less robust against error than the network model. Finally, the proposed metrics provide a foundation for rigorous empirical research on the complexities of human social systems.

Author summary

Human social systems, like organizations or societies, are complex. This research distinguishes between two key forms of complexity: structural complexity, which refers to how a system is organized, and dynamic complexity, which describes the unpredictability of how it behaves. These concepts are explored using three simple formal models: a hierarchical system, like a company with a clear chain of command, a network system, where individuals interact more freely, and a combination of the two, that might come closest to a real-world organization. The results show that systems with higher structural complexity are more efficient at solving familiar problems. However, they may struggle with

OPEN ACCESS

Citation: Roos M (2025) The complexity of problem-solving human social systems: Structural vs dynamic complexity. PLOS Complex Syst 2(7): e0000055. https://doi.org/10.1371/journal.pcsy.0000055

Editor: Marcos Oliveira, University of Exeter Faculty of Environment Science and Economy, UNITED KINGDOM OF GREAT BRITAIN AND NORTHERN IRELAND

Received: September 18, 2024

Accepted: June 13, 2025

Published: July 11, 2025

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pcsy.0000055

Data availability statement: The program code of the models is available in the CoMSES library: Michael Roos (2025, March 01). "Simple models with different types of complexity" (Version 1.1.0). CoMSES Computational Model Library. Retrieved from: https://www.comses.net/codebases/68b87f5e-582b-481e-bff9-c7f3be36ad6e/releases/1.1.0/. The dataset produced with these models and ODDs of the models are provided there, too.

Funding: The author(s) received no specific funding for this work.

Competing interests: The author has declared that no competing interests exist.

new challenges due to their rigid structure. In contrast, more dynamic systems are flexible and adaptable, but less efficient and stable. This work highlights the trade-off between efficiency and robustness in human social systems. Systems designed for efficiency often lack adaptability, while those built for flexibility may face unpredictability. The findings help us understand how to design systems—whether organizations or societies—that can balance efficiency with the ability to cope with uncertainty and error.

1. Introduction

The complexity of human social systems (HSS) has long captivated researchers across various fields, including history and archaeology [1–3], sustainability research [4–7], sociology [8,9], economics [10], and business studies [11]. One interesting question is how the complexity of a HSS such as a national economy is related to some outcome the system produces, e.g., economic growth. Hausmann et al. posit that countries with higher economic complexity generate stronger economic growth due to the accumulation of knowledge [12]. Another question is how and why the complexity of human systems evolves over time. Joseph Tainter's influential theory of societal collapse frames human societies as problem-solving systems [1,4,13]. As societies confront mounting challenges, they develop more complex institutions and technologies. However, this escalating complexity eventually reaches a tipping point where its maintenance becomes more costly than beneficial, leading to a process of decomplexification.

The arguments put forth by Hausmann et al. and Tainter possess a degree of plausibility; however, they are deficient in terms of a precise characterization of complexity in HSS. The concept of "economic complexity" functions as a heuristic device, as measured by the Economic Complexity Index proposed by Hidalgo and Hausmann [14]. This index employs a predominantly mathematical approach aimed at reducing the dimensionality of large data sets. Tainter and Taylor [5] define "complexity in human social systems as differentiation in structure and behavior, and degree of organization" (p. 168). Differentiation in structure, behavior, and organization are elements of social complexity, but they do not always align or produce the same effects. I argue that it is imperative to distinguish between *structural complexity*, defined as the organizational properties of a system, and *dynamic complexity*, which refers to the behavioral patterns and interactions within the system. Human social systems are typically characterized by both structural complexity and dynamic complexity. The performance of these systems in various dimensions is contingent on the degree of each property.

This paper answers the following research questions: 1) How can we characterize the structural and dynamic complexity of problem-solving HSS? 2) What is the relation between the two complexity types and the performance dimensions solution completeness, efficiency and robustness? 3) How can we measure structural complexity and dynamic complexity of HSS? To answer these questions, I use three simple agent-based models that are designed to illustrate the meaning of structural complexity and dynamic complexity as clearly as possible. While these models are conceptual in nature and do not aim to represent actual systems, they could serve as the theoretical underpinnings for more sophisticated, empirically validated models that are applicable to specific contexts. The models allow us to explore and understand precisely why structural and dynamic complexity lead to different outcomes. Furthermore, I propose several new metrics designed to capture key features of structural complexity (hierarchy, differentiation of roles, specialization, decentralization of decisions) and of dynamic complexity (volatility in the problem-solving process, diversity of the solution). Since measurement requires conceptual clarity, the metrics contribute to our understanding of the various properties of complexity and their roles in HSS. The agent-based models are used to validate the metrics.

The distinction between dynamic and structural complexity is mirrored in two distinct conceptions of complex systems, reflecting different disciplinary origins. The first conception of complex systems is the one commonly used in complex systems theory with roots in physics, chemistry, and some parts of biology. Ladyman et al. [15] describe this approach: “A complex system is an ensemble of many elements which are interacting in a disordered way, resulting in robust organisation and memory” (p. 57). The focus here is on “disordered” *interactions* that result in some sort of order or pattern. The alternative conception is characterized by Carlson and Doyle [16]: “In engineering and biology, complex systems are almost always intrinsically complicated, and involve a great deal of built in or evolved structure and redundancy in order to make them behave in a reasonably predictable fashion in spite of uncertainties in their environment” (p. 1412). The focus here is on the complicated structure of the system that leads to predictable or orderly behavior. The key difference between these characterizations of complex systems is that the first one emphasizes decentralized interaction and self-organization as sources of complexity, while the second emphasizes designed (or, in the case of biological systems, evolved) complicatedness as central to complexity. HSS, especially large-scale ones such as societies, are characterized by both features. For this reason, Andersson et al. [17] describe human societies as wicked, that is, characterized by self-organized complexity and intentionally designed complicatedness. To align their approach with the contrasting definitions of complex systems given above, I propose to call the (self-organized) complexity in [17] *dynamic complexity* and the designed complicatedness *structural complexity*.

Andersson and Törnberg [18] list the main characteristics of structural complexity (or complicatedness) and (dynamic) complexity. Structural complexity, which is typical for technology, but is also a property of human organizations, is characterized by specialized system components organized in hierarchies. There may be many component classes with few components in each class. The components are “slaved” in the sense that they do not make sense separately and have aligned goals and functions or roles. Such a “system may pack very large numbers of components into delineable compartments organized in a level hierarchy. This strongly structures the patterns of permitted interactions” ([18] p. 121). In other words, structural complexity refers to the inherent properties of a system, such as the number and role of its components, the relationships among them, the hierarchy within the system, and the degree of specialization. Conversely, dynamic complexity is characterized by many components (but few component categories) at the same level of organization. There is high redundancy, because components can replace each other due to high similarity. There are only loose exogenous constraints on the formation and dissolution of interactions between components, but strong endogenous structuring of interactions occurs. In contrast to the static properties of a system described by structural complexity, dynamic complexity is concerned with how a system behaves over time. This distinction is crucial: while structure provides the framework, dynamic complexity emerges from how the system operates within this framework, especially in response to external or internal changes.

The purpose of this paper is to apply the framework proposed by Andersson and Törnberg to problem-solving HSS, i.e., organizations. Andersson and Törnberg consider human organizations as “trans-complicated”, which means that they have an element of dynamic complexity added to the pure structural complexity. This adds the complication that components can have “an agenda of their own” such that the alignment of the components with the system goals must be actively maintained.

In particular, I translate the verbal characterizations of dynamic and structural complexity into features of formal agent-based models, which has several advantages. First, terms like “slaved components”, “redundant components”, or “agenda of their own” obtain precise meanings. Second, a formal model can be used to analyze how the output of the system depends on its features. If we consider human organizations as problem-solving systems, then the output is a solution to a problem. We can evaluate the ”goodness” of a solution along several dimensions: its completeness, i.e., how many aspects of the problems it addresses correctly, its efficiency, i.e., how much input was needed to achieve the solution, and its robustness, i.e., how strongly it is affected by errors. Again, what these terms mean can be made precise with models. Third, formal models can be used to derive metrics for the different aspects of structural and dynamic complexity. If we want to compare the complexity of different organizations empirically or assess how the complexity of a specific organization (or HSS) has changed over time, we must be able to measure complexity with metrics that are well-understood.

In Section 2, I introduce the agent-based models designed to capture the relevant features of structural and dynamic complexity. Section 3 presents simulation results that relate problem-solving performance to aspects of the complexity of the system. Section 4 proposes several metrics of structural and dynamic complexity. Finally, in Section 5, I conclude with reflections on the broader implications for understanding social complexity.

2. Models and methods

In this section, I present three simple models that serve two purposes: first, I use them to demonstrate what structural and dynamic complexity can mean in the context of problem-solving human social systems. I will discuss how aspects of the two types of complexity can be measured in the models, and how metrics for real-world systems might be developed. Second, I apply the models to simulate problem-solving performance and analyze how it relates to structural and dynamic complexity. Since clarity is the main point of an illustration [19], the models are highly simplified and do not claim to represent specific real-world systems.

The first model represents a hierarchical system (HS) with a clear structure consisting of a manager and specialists. Its real-world analog could be a company or a government agency. This model embodies high structural complexity, characterized by multiple levels of hierarchy, a significant degree of specialization, and well-defined decision rights. The second model represents a non-hierarchical system (NS) - a random network of generalists. We could imagine a network of loosely cooperating professionals or firms, or a cooperative. This model has a lower structural complexity, with minimal hierarchy and specialization, allowing for more flexible and decentralized interaction among its components. Both systems face a problem to be solved. The two models are extreme cases designed to illustrate the characteristics of structural and dynamic complexity as clearly as possible. As extreme cases they are likely to be very rare in reality. In reality, most HSS or human organizations are hybrid systems – [18] call them trans-complex, trans-complicated, or wicked systems - that are both structurally and dynamically complex. As the most realistic model of a human organization such as a firm, I present the hierarchical network system (HNS), which is a combination of the HS and the NS. In real organizations, there typically exist formal hierarchical reporting structures, but also informal collaboration networks based on informal status, experience, or sympathy.

The systems are implemented as agent-based models using the software NetLogo [20]. The code as well as ODD protocols [21] and the output data are available in the CoMSES library [22].

2.1. Hierarchical system (H-system)

The model of the HS consists of *n* agents: 1 manager and *n-1* specialists. The manager has no special expertise and is responsible for coordinating the problem-solving process. Each specialist has a number *s* actual skills, randomly selected from *S* possible skills. The skill level for each skill of a specialist is set to 1, representing full proficiency in that skill. I assume that specialists have one unique skill ($s = 1$). Hence an S-dimensional vector with one 1 and *s-1* 0s represents the skill profile of a specialist.

The agents are arranged in a tree or star network. Fig 1 shows an example ($n = 7$), with the manager (in red) at the top connected to each specialist (blue circles). The numbers at the blue circles indicate the non-zero element of the skill vectors (here with $s = 10$), which represents the unique skill of each specialist. The specialists are connected only to the manager and not to each other, reflecting a typical hierarchical organization where communication flows through a central authority. The star network is the simplest way to represent a 2-level hierarchy.

The problem facing the system is represented as a σ-dimensional vector with π entries set to 1 and the remaining $\sigma - \pi$ elements set to 0. I call $\sigma \in \mathbb{N}$ the *problem space* and $\pi \in \mathbb{N}$ with $\pi \leq \sigma$ is the *problem size*. The idea is that there can be σ different aspects or dimensions of a problem, but a given problem involves only some π of all possible dimensions. The problem is randomly generated at the beginning of the simulation.

The manager perceives the problem and distributes it to the relevant specialists according to their skills. Each specialist receives a copy of the problem and tries to solve it by setting any 1 corresponding to his skill to 0. After the specialists have modified their respective copies of the problem, they are returned to the manager. The manager consolidates the modified problems from all specialists into a final solution. The consolidation process involves merging the modified problem vectors from each specialist to create a single problem vector that reflects all modifications. Because the manager knows the capabilities of all the specialists, the problem is delegated to the specialists only once. If a problem dimension cannot be addressed because no specialist has the required skill, the final solution will be incomplete. Nevertheless, the manager stops the problem-solving process because a second delegation cannot lead to a better solution. In this way, each problem is solved in the best possible manner in a single simulation round. Note that the manager is the only decision maker, while the specialists only perform the tasks assigned to them. In this sense, the specialists are “enslaved”. The manager makes two decisions: 1) to whom to assign a task, and 2) to stop the problem-solving process after the final solution has been assembled. We consider these decisions to be the two actions of the manager, while each specialist performs the single action of solving a particular aspect of the problem.

Each action taken by the manager or specialist has a cost of 1. The total cost is the sum of all individual costs incurred in the problem-solving process. The completeness of the final solution is calculated as

$$CS = \pi - \pi',$$

where π is the initial problem size and π' is the remaining problem size after the problem-solving process has ended. This metric provides an indication of how effectively the problem was solved, since π is the number of initially active problem dimensions and π' is the number of dimensions that could not be addressed due to a lack of relevant specialists.

The H-system is an idealized system that describes how one would organize a problem-solving process with limited resources such that it produces the best possible solution in a cost-minimizing way. The problem-solvers are specialized

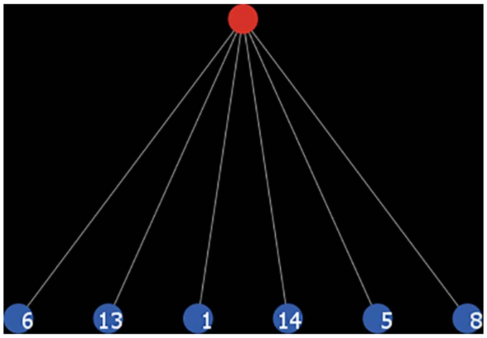

Fig 1. Representation of the hierarchical system.

https://doi.org/10.1371/journal.pcsy.0000055.g001

to the maximum degree: they have the highest possible expertise in just one dimension of the problem space. Furthermore, they do not make any decisions, but simply work on the aspect that is assigned to them. To free the problem-solvers from any kind of transaction cost, a coordinator (or manager) is needed that analyzes the problem, decides on how the resources are used and integrates the single elements of the solution. Hence problem analysis and solution integration are centralized to minimize organizational costs.

To make the model slightly more realistic, I assume that the manager can make mistakes when analyzing the problem. If the manager is not a specialist in every aspect of the problem, a misperception of the actual problem might occur. This misperception is modeled by a random variable ε that is a random floating-point number with $0 \leq \varepsilon < m \leq 1$. The parameter m regulates the maximum degree of misperception. Instead of the actual problem, the manager perceived each element i of the problem as $|k_i - \varepsilon_i|$, where k_i is either 0 or 1. The manager believes that an element i of the problem is present (i.e., equal to 1), if $|k_i - \varepsilon_i| \geq 0.5$ and assigns it to the relevant specialist. Hence, as long as $m \leq 0.5$, the manager's beliefs are correct, and no erroneous assignment takes place. If $0.5 < m < 1$, some errors will occur, and with $m = 1$, the manager has a completely random perception of the problem.

2.2. Non-hierarchical system (N-system)

The second model represents a decentralized network of n generalist agents working together to solve a problem in a distributed way. Each generalist has s skills randomly chosen from a set of S possible skills. Since several generalists can have overlapping skills, they can be seen as "redundant components". The skill level $\lambda \in]0,\ 1]$ for each skill is uniform across all agents.

Agents are connected in a random network in which each agent has, on average, γ links to other agents. The network is generated by randomly connecting each agent to γ other agents, ensuring that no agent is connected to itself and avoiding duplicate links. Fig 2 shows an example for $n = 7$, $s = 3$, $S = 10$,and $\gamma = 3$. The vectors in Fig 2 indicate the non-zero skills of each agent.

The problem is represented in the same way as before as a σ-dimensional vector with π entries randomly set to 1 and the remaining $\sigma - \pi$ entries set to 0. The problem-solving process begins with a randomly selected agent perceiving the problem and attempting to solve it by reducing the values of the corresponding entries in the problem list according to its skill level. Specifically, for each 1 in the problem list that corresponds to a skill possessed by the agent, the agent reduces the value by the amount specified by the skill level λ. The modified problem is then passed to a randomly chosen

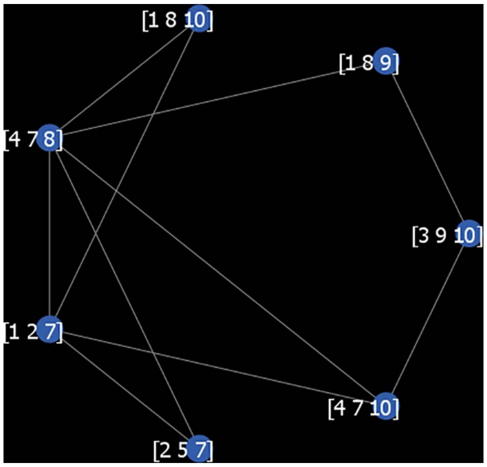

Fig 2. Representation of the non-hierarchical system.

https://doi.org/10.1371/journal.pcsy.0000055.g002

neighboring agent (connected via the network), which continues the process in the next simulation round. This modeling choice implies that the agents do not know the skills of their network neighbors and thus use other criteria for choosing to whom to pass on the problem. The problem continues to be passed on among agents until it is either completely solved (i.e., all problem entries have been reduced to zero) or until an agent reaches a predefined maximum number of problem-solving attempts, *max-work*.

When the simulation stops, the total cost and completeness of the solution are calculated and displayed. Each time an agent deals with the problem and passes it on, a cost of 1 is incurred. The total cost is the sum of all such costs for all agents over all rounds. The completeness of the solution is calculated in the same way as before.

Like the H-model the N-model is stylized. Since it is meant to represent a minimal degree of coordination, the assignment of skills to the problem occurs in a random fashion, which is a built-in inefficiency. However, without any kind of coordination, the problem-solving process would not come to an end, so each agent decides to stop the process after working on the problem a certain number of times. Note that there is a partial redundancy in the models: the skills of the generalists overlap, because each agent has several skills and there is no constraint that precludes an overlap. Depending on the parameterization, the sum of all skill levels λ over the skills *s* of a generalist, could be higher than the total skill level of a specialist. Furthermore, all *n* generalists might have a larger set of skills than the same number of specialists. This implies that the group of generalists in principle can solve the problem as completely as the specialists or even better, but potentially at higher costs, because a single generalist cannot solve an aspect of a problem completely in one working round if $\lambda < 1$. Regarding the costs, note that I do not model any decision costs. A cost of 1 is incurred, if a generalist receives the problem, no matter whether the generalist can work on it, passes it on or stops the process. Hence the N-model does not have costs that are equivalent to the costs of the manager in the H-model.

2.3. Hierarchical network system (HN-system)

Real human organizations often have features of both the H-system and the N-system. The HN-system consists of a manager and *n-1* generalists. The generalists are grouped into two teams. Each team has a team leader who is the only team member that is linked to the manager. Each team forms a network, in which each member on average is linked to γ other members. Fig 3 shows an example of the system with $n-1=14$ generalists. The team leaders are represented by blue circles, the normal team members are green, and the manager is red.

The manager has no problem-solving skills but allocates the initial problem to one of the two team leaders. To do this, the manager compares the aggregated skill of each team with the problem and chooses the team that can solve the problem more completely in one round. More precisely, the manager first sums up the skill vectors of all the members of a team and subtracts the resulting vector from the problem vector. The problem is assigned to the team with the smaller resulting vector sum. Like in the N-model, each team member (including the team leader) has *s* random skills out of the

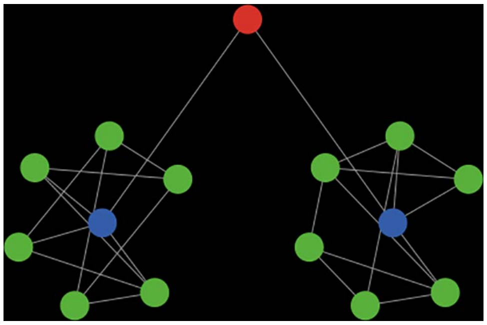

Fig 3. Representation of the hybrid HN-system.

https://doi.org/10.1371/journal.pcsy.0000055.g003

set of S possible skills, each skill with a skill level of λ. The problem is again a vector with π randomly assigned 1s and $\sigma - \pi$ 0s.

In terms of skills, the team leaders do not differ from the other team members, i.e., they are also generalists that can work on the problem if they possess relevant skills. However, the team leaders are the only team members that are linked with the manager. They receive the problem from the manager and decide to turn it back after a while. When the manager receives a solution from a team leader, the process stops. The team leader returns the (partially) solved problem to the manager after receiving the problem *max-leader* times (in total from the manager and from the team members). The team members work on the problem and then pass it on to one randomly chosen link-neighbor (except the one from whom they received it in the previous round). There is no constraint on how often they can receive the problem, since they do not decide to stop the process.

The main idea behind the HN-model is that organizations often have both formal structures - here the links between the manager and the team leaders and the decision power of these agents - and informal ones – here the networks among the team members. There is no formally given protocol for how the teams deal with a problem. Apart from the team leader, nobody has decision power to stop the process, but the team leader does not have enough formal authority to tell the team members what to do. In this sense, the team members can be seen to have an "agenda of their own". The random network among the team members might represent social relationships, e.g., who is willing to collaborate with whom. The team networks might have cliques without the team leader, in which the problem can circulate for a long time. An alternative design could be that the team leader is linked to all team members and that the problem must be passed on to the team leader in regular intervals such that there is closer monitoring of the process. This design would make the model more like the H-model, but with two hierarchy levels (manager and team leader). I chose the present design to have a hybrid model of the H-model and the N-model.

As the HN-model has been presented so far, there is no need for an organization with several teams. If the skills are distributed with identical probability across all generalists, the teams on average will have similar aggregated skills. To increase the realism of the model slightly, I assume that the parameter *skill-bias* determines the probability, with which team 1 has skills in the first half of the skill space S and team 2 has skills in the second half. If, e.g., *skill-bias* = 0.7, then skills in the first (second) half of the skill space are selected with probability 0.7 in team 1 (team 2). The bias in the skill distribution could be interpreted as some general specialization of the team, e.g., team 1 is more skilled in technical topics and team 2 in legal aspects of the problem space. Team specialization, in turn, is not useful if the non-zero elements of the problems are distributed with equal probability across the problem space. Therefore, I also introduce the parameter *problem-bias* that determines the probability of the problem elements in the first half of the space being equal to one. If the problem has more ones in the first half than in the second, we could interpret it as a primarily technical problem, for example.

Analogous to the H-model, I also allow that the manager misperceives the problem and hence assigns it to the team that is not the best to solve it. As before, the manager perceives the problem elements as $\left|k_i - \varepsilon_i\right|$, where $\varepsilon_i < m$ is the error term and m is the degree of misperception. When assigning the problem to one of the teams, the manager compares the perceived problem with the aggregated skills of the teams and hence may give the problem to the wrong team, if the misperception is large.

3. Simulation results

As discussed in the Introduction, the problem-solving performance of a HSS can be evaluated along three dimensions: completeness, efficiency, and robustness. The completeness of the final solution is calculated as

$$CS = \pi - \pi',$$

where π is the initial problem size and π' is the number of non-zero problem elements after the problem-solving process has ended. Efficiency is simply measured as the inverse of total costs per completeness.

$$E = \frac{CS}{TC},$$

i.e., efficiency is high, when completeness is high and total costs are low.

Robustness refers to how completeness and efficiency depend on the degree of misperception in the H-model and the HN-model. Since there is no manager that analyzes and assigns the problem to the problem solvers in the N-model, robustness is not applicable there.

To compare the performance of the models, I simulate each model 1000 times with the parameters shown in Table 1. A sensitivity analysis, which is presented in the online appendix, shows that the model responds in plausible ways to variations of the parameters. The analysis proceeds in two steps: first, I analyze all models without skill bias, problem bias, and misperception. In the second step, I introduce the biases and misperception in the H-model and the HN-model.

3.1. Completeness and efficiency without misperception and biases

Table 2 shows the mean and the standard deviation of the measures of completeness and efficiency in the three models. On average, the H-system generates more complete and more efficient solutions than the N-system. This is not surprising, since it is the purpose of the hierarchical design to solve problems effectively and efficiently. Interestingly, the HN-system generates on average more complete solutions than the H-system but has a very low mean efficiency. Both findings for the HN-system are related, because the average duration of the process is much longer than in the other model (17.1 rounds in the HN-system vs. 8.1 rounds in the N-model vs. 1 round in the H-model). With many rounds, the HN-system can solve the problem quite well, because it has a large expected number of skills which will be used several times. The many rounds, however, also generate large total costs. The differences in average completeness and efficiency are statistically significant across all three models (one-side t-test, unequal variances, $p < 0.001$).

The boxplots in Figs 4 and 5 provide an impression of the distribution of outcomes across the 1000 simulation runs. The most remarkable results is that there is a lot of variation across in the HN-model. This system can produce extreme

Table 1. Parameters used for performance simulations.

Parameter	H-model	N-model	HN-model
N	8	7	15
S	15	15	15
S	1	3	3
λ	1	0.3	0.3
σ	15	15	15
π	10	10	10
γ	–	3	3
max-work/max-leader	1	3	3

https://doi.org/10.1371/journal.pcsy.0000055.t001

Table 2. Performance of the models.

		H-model	N-model	HN-model
Completeness	Mean	4.682	4.248	5.615
	Std. Dev.	0.926	1.160	1.353
Efficiency	Mean	0.695	0.530	0.202
	Std. Dev	0.045	0.107	0.091

https://doi.org/10.1371/journal.pcsy.0000055.t002

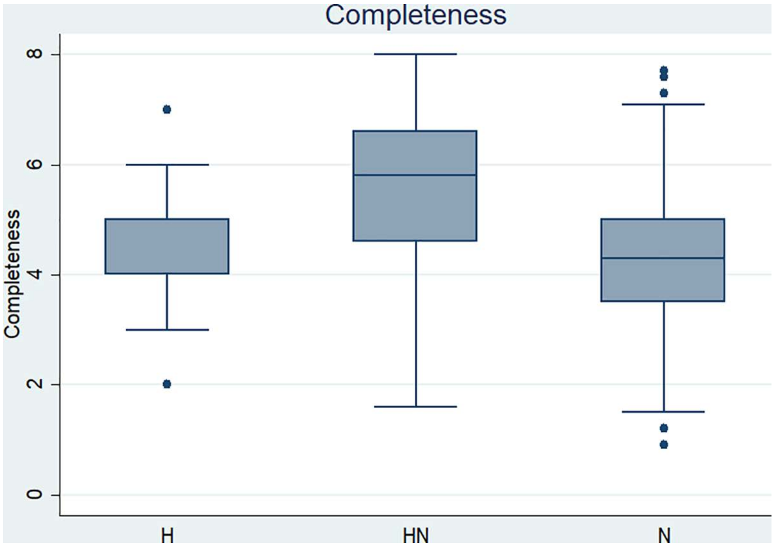

Fig 4. Boxplots of completeness in the three models.

https://doi.org/10.1371/journal.pcsy.0000055.g004

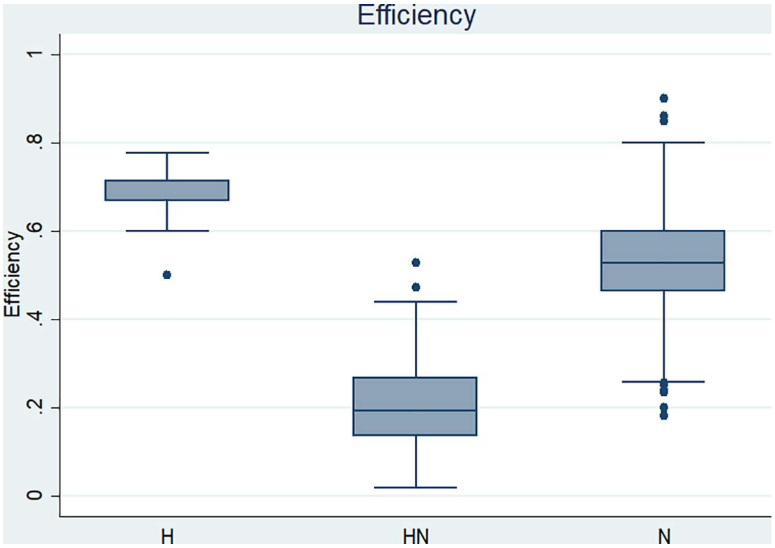

Fig 5. Boxplots of efficiency in the three models.

https://doi.org/10.1371/journal.pcsy.0000055.g005

outcomes with very high or low completeness and efficiency. Again, this is due to the large variation in the duration of the problem-solving process which ranges from 3 rounds to 205 rounds. In some cases, the problems for a long time do not come back to the team leader who is the only one who can decide to stop the process.

3.2. Completeness and efficiency with misperception and biases

Robustness against error is the third performance criterion I want to consider. In the H-model, error is caused by the misperception of the problem by the manager, $|k_i - \varepsilon_i|$ with $0 \leq \varepsilon < m \leq 1$. Strong misperception lets the manager assign some elements of the problem to the wrong specialists whose skills are not needed to solve the problem. Erroneous problem assignment creates inefficiency, because some specialists invest time on the problem despite not being able to contribute. Furthermore, it makes the problem-solving process less effective, because some aspects of the problem are not dealt with, even though the system has the relevant skills available.

Fig 6 shows how the relative completeness – the percentage of initial elements of the problem that have been solved by the process – in the H-model depends on the degree of misperception m. The graph is a local polynomial smoothed line (with a 95% confidence band) that represents the 1000 simulations for each $m = \{0.4,\ 0.45,\ 0.5,\ \ldots 0.95\}$. We can see the average relative completeness declines from 0.47 at $m = 0.4$ to 0.25 at $m = 0.95$ which is a decrease by 47%. Note that the average relative completeness in the N-model of 0.4248 (4.248 from Table 2 divided by 10 initial elements) is already reached at $m = 0.55$. Already at relatively low levels of misperception (i.e., $m \geq 0.55$), the H-System does not solve the problem more completely than the N-model. Misperception causes the efficiency (completeness relative to costs) to decline, too (Fig 7). The highest degree of efficiency of 0.69 is reached when misperception does not matter (i.e., $m < 0.5$). Remember that the manager assigns a problem element to the specialist with skill i, if $|k_i - \varepsilon_i| \geq 0.5$. With $m = 0.95$ average efficiency falls to about 0.45. The H-system's efficiency falls below 0.53, which is the N-model's average efficiency, at $m \geq 0.75$. Hence, the H-system with misperception quickly becomes less effective than the N-model but remains more efficient somewhat longer.

Above, I argued that the organization of the generalists in two teams in the HN-model makes sense only in the presence of problem-bias and skill-bias, i.e., if the non-zero elements on are unevenly distributed in the problem vector and if the skills are unevenly distributed across the teams. Absent skill-bias, the misallocation of the problem to the wrong team due to misperception reduces the relative completeness of the solution (see left column of Fig 8). The average relative completeness falls from about 0.53 to 0.52 if m increases from 0 to 0.9. The effects are small and not significantly different, because if there is no systematic bias in the skills between the teams, there are only small random differences in the aggregate skills of the teams. If the teams differ systematically in their skills, then misperception matters more (see second and third column in Fig 8). A more biased skill distribution leads to more complete solutions, when there is more problem bias unless the misperception is very high ($m = 0.9$). Note that even with very high degrees of misperception, the average relative completeness remains well above 0.42 (or above) which is the relative completeness in the N-model.

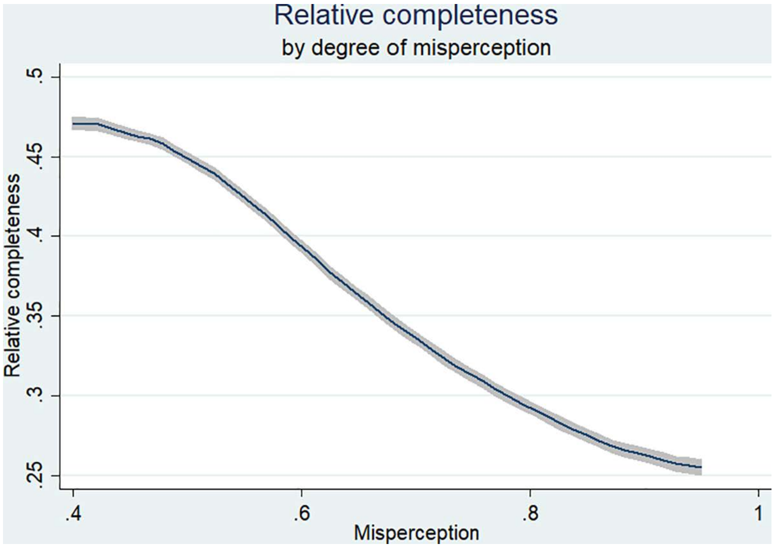

Fig 6. Relative completeness of the solution as a function of the degree of misperception in the H-model.

https://doi.org/10.1371/journal.pcsy.0000055.g006

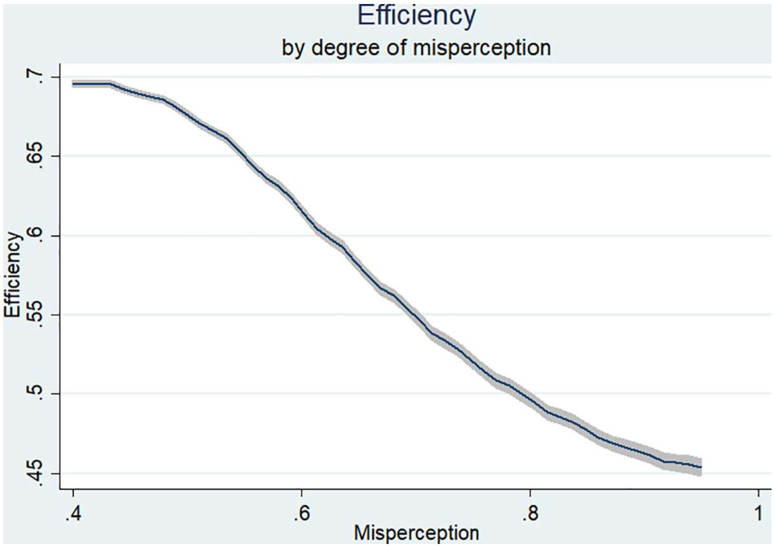

Fig 7. Efficiency as a function of the degree of misperception in the H-model.

https://doi.org/10.1371/journal.pcsy.0000055.g007

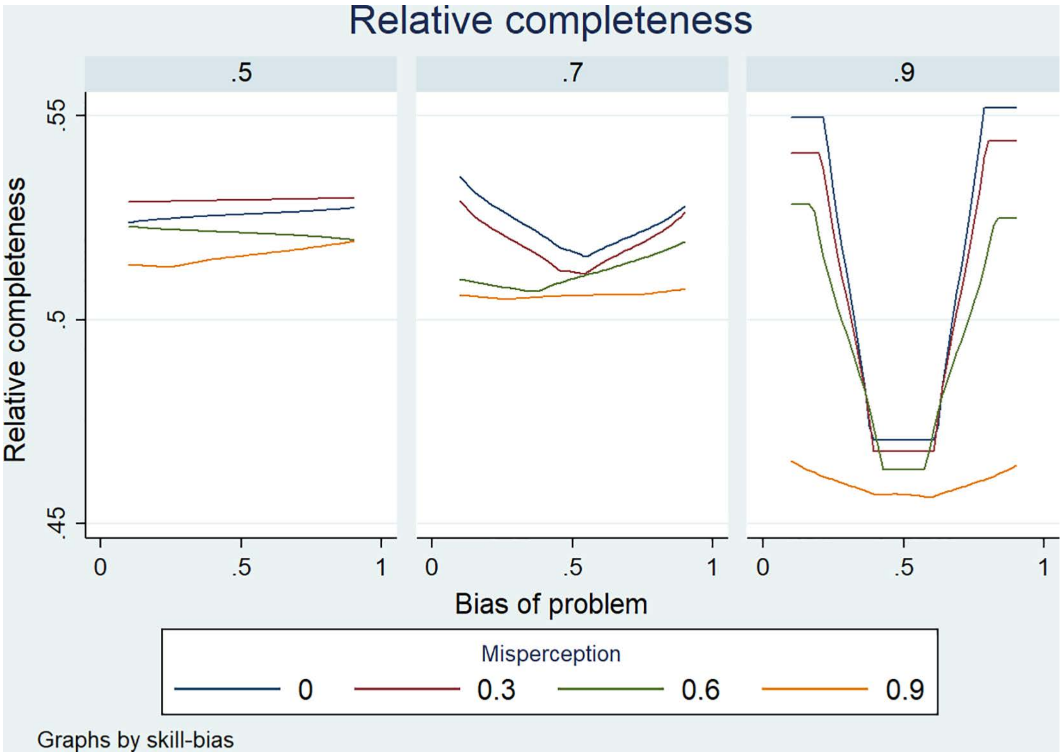

Fig 8. Relative completeness of the solution as a function of the degree of misperception in the HN-model.

https://doi.org/10.1371/journal.pcsy.0000055.g008

I find no systematic relationship between the total costs of the problem-solving process and misperception. This finding is not surprising, because once the problem has been assigned to a team, the duration of the problem-solving process and hence the total cost is largely random in the HN-model.

Since there is no systematic pattern in total costs, efficiency largely shows the same pattern as completeness. Fig 9 compares the efficiency without misperception and the efficiency at $m = 0.9$. The degree of efficiency differs only between no misperception and high misperception, if both problem-bias and skill-bias are large (right column of Fig 9). As before, the overall degree of efficiency in the HN-model is very low due to the often long duration of the problem-solving process.

4. Complexity metrics

In this section I propose several metrics that can be used to measure the degree of structural and dynamic complexity. Such measures are useful to compare different real-world systems and to track their evolution over time. I use the models from the previous sections to illustrate the metrics.

4.1. Measuring structural complexity

Building on the previous discussion, the following interrelated features are key elements of structural complexity:

- Hierarchy
- Differentiation of roles
- Specialization
- Dependence (Centralization of decisions)

For each of these elements, it seems plausible that we can measure its degree, with higher degrees indicating greater structural complexity of the system.

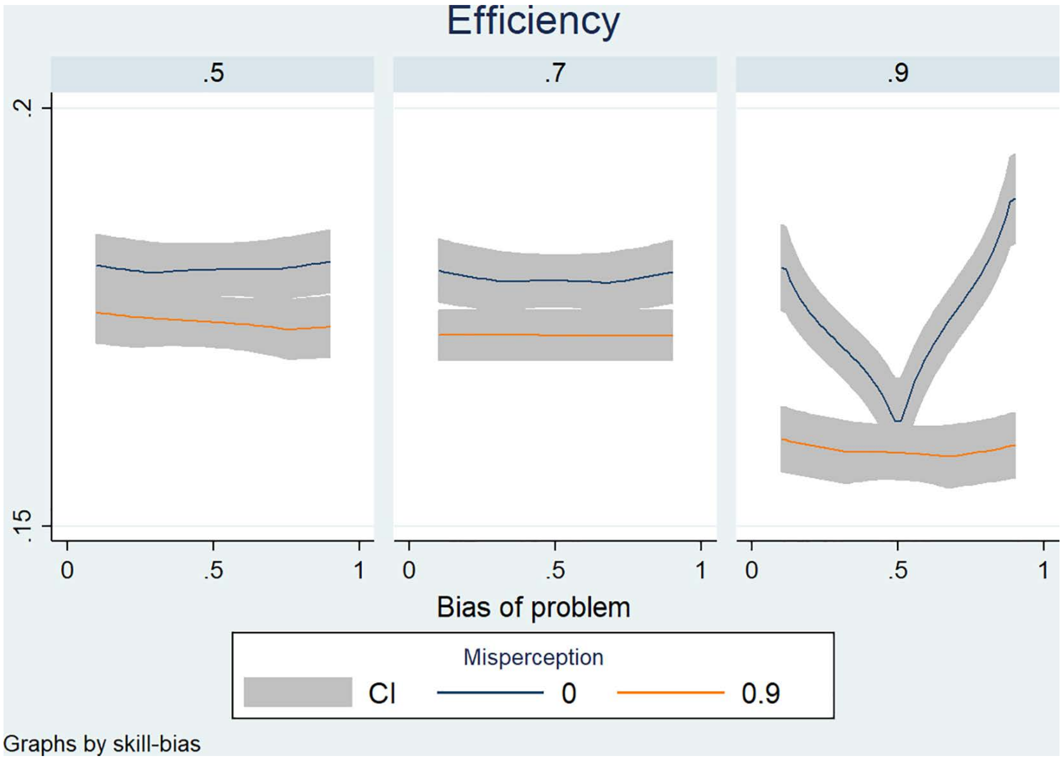

Fig 9. Efficiency as a function of the degree of misperception in the HN-model.

https://doi.org/10.1371/journal.pcsy.0000055.g009

There are several types of hierarchy. [23] distinguishes order hierarchy, inclusion hierarchy, control hierarchy, and level hierarchy. For the description of problem-solving HSS, the control hierarchy, in which orders flow down the hierarchy and information and requests flow up, seems to be the appropriate type. If agents at higher levels of a hierarchy can control those at lower levels, the hierarchy implies a vertical differentiation of roles in the system. It also implies that the behavior of lower-level agents is dependent on the goals and commands of higher-level agents. A control hierarchy thus limits the autonomy of some agents and imposes behavioral constraints and structure on decision processes. Finally, hierarchy is necessary to integrate the work of specialists who have little or no overlap in their skills. Thus, the integration of partial outputs is achieved through a centralized organization of the workflow. A structurally complex HSS in this sense is designed to allow a high degree of specialization of agents dealing with specific aspects of a problem and to manage the problem-solving process in an efficient way.

A control hierarchy can be represented as a tree graph. The number of control levels of the tree graph is a natural metric to measure the degree of hierarchical organization of the HSS. In the H-model, there is only a single control level at which the manager is located. Therefore, we can say that the H-model has a degree of hierarchy equal to 1. The N-model has no hierarchy. It cannot be represented as a tree, because the graph is cyclic. Therefore, all agents are at the same level, so the degree of hierarchy is 0. The HN-model could be seen to have a degree of hierarchy of 2, since it has a manager and two team leaders. However, the team leaders do not exert control over the other team members so that it is more appropriate to assume the same hierarchy as in the H-model.

Role differentiation could be measured by simply counting the defined distinct roles, i.e., two roles (manager and specialist) in the H-model, one role (generalist) in the N-model and three roles (manager, team leader and team member) in the HN-model. While this works in the models presented here, it is a crude way of measuring role differentiation. In many real-world cases, roles may not be so easily distinguished, for example, because there are hybrid roles like the role of the team leader, in which an agent has both managerial tasks and specialist tasks. Therefore, I assign task categories to each role. In the models presented above, there are two categories of tasks: (1) problem solving (P) and (2) decision making (D). A role of an agent i can be defined as set R_i with P_i and D_i as potential elements indicating whether the agent performs problem solving tasks and/or decision-making tasks. In more sophisticated models, there may be more task categories and thus a higher-dimensional role vector. Hence, the role of the manager is $R_M = \{D\}$ and the role of the specialist is given by $R_S = \left\{P\right\}$. We can consider the generalists in the N-model and team leaders in the HN-model as hybrid agents that perform both tasks, because they also decide how to proceed in the process, hence $R_{G,TL} = \{P, D\}$. As a metric of the degree for role differentiation, I define the **Role Differentiation Index** (RDI):

$$RDI = \frac{\textit{Number of distinct roles}}{RSI} - 1,$$

where RSI is an index for role similarity (Role Similarity Index). I use Jaccard Similarity (JS) to measure the similarity of the roles of different agents, i.e., the number of shared task categories in role sets of agent types k and l is divided by the number of unique tasks in the two sets:

$$JS_{kl} = \frac{\left|R_k \cap R_l\right|}{\left|R_k \cup R_l\right|}.$$

In the H-model, the intersection of roles of the manager and the specialists is empty, i.e., $\left|R_M \cap R_S\right| = \left|\left\{D\right\} \cap \{P\}\right| = 0$, whereas the union has two elements, $\left|R_k \cup R_l\right| = 2$, hence $JS_{MS} = 0$. Since all specialists have the same role, we get $JS_{SS} = \frac{\left|R_S \cap R_S\right|}{\left|R_S \cup R_S\right|} = \frac{1}{1} = 1$. The Role Similarity Index is calculated as the average of the Jaccard Similarity for all pairwise comparisons of all agents. Assuming 8 agents in the H-model, as I did in Section 3, we have 7 pairwise comparisons of the manager with the specialists and $\binom{7}{2} = 21$ comparisons of specialist pairs. Hence, we obtain

$$RSI_H = \frac{7 \times 0 + 21 \times 1}{7 + 21} = \frac{21}{28} = 0.75,$$

Resulting in a degree of role differentiation in the H-model of

$$RDI_H = \frac{2}{0.75} - 1 \approx 1.67.$$

In the N-model all generalists have the same role, hence $RSI_N = 1$ and there is no role differentiation, $RDI_N = 0$. In the HN-model with $n = 15$, the Role Similarity Index is

$$RSI_{HN} = \frac{(1 \times 1) + (2 \times 0.5) + (1 \times 0) + (12 \times 0.5) + (2 \times 1) + \binom{12}{2} \times 1}{\binom{15}{2}} = \frac{75}{105} \approx 0.714.$$

Therefore, the degree of role differentiation in the HN-model is

$$RDI_{HN} = \frac{3}{0.714} - 1 \approx 3.2.$$

The additional role of the team leader hence increases role differentiation compared to the H-model and makes the HN-model more structurally complex in this dimension.

Specialization differs from role differentiation in that it refers to the level of knowledge or the degree of expertise an agent has in a particular area. In general, there is a trade-off between breadth and depth of knowledge. A highly specialized expert has a high level of knowledge or skill in very few areas. In the case of the H-model, the specialists are assumed to have a maximum skill level of 1 in only one skill. For a problem-solving HSS, there is little direct use for multiple specialists with the same skills, since multiple specialists of the same type do not improve the quality of the solution. For this reason, I have assumed in the H-model that each specialist has a unique skill. In this sense, the degree of specialization depends on the uniqueness or non-overlap of the skills of the problem-solving agents. As a metric for the degree of specialization in the system, I propose the **Skill Non-Overlap Index** (SNOI):

$$SNOI = \frac{1}{\binom{n}{2}} \sum_{i<j} \left(1 - \frac{Overlap(i,j)}{min(k_i, k_j)}\right).$$

The overlap of skills of a pair of agents i and j is determined as

$$Overlap(i,j) = \sum_{s=1}^{S} \min\left(Skill(i,s),\ Skill(j,s)\right),$$

where $Skill(i,s) = 1$ if agent i has skill s, and 0 otherwise and S is the total number of skills. We normalize the skill overlap by the total number of skills k possessed by the agents with the fewest number of skills in the pair. The SNOI of a pair of agents is given by

$$SNOI(i,j) = 1 - \frac{Overlap(i,j)}{\min\left(k_i, k_j\right).}$$

In the H-model, each specialist possesses a unique skill such that $Overlap(i,j) = 0$ and $SNOI(i,j) = 1$ for all $\binom{7}{2} = 21$ specialist pairs. Hence, the overall degree of specialization is given by

$$SNOI_H = 1,$$

which is the maximum this index can take.

Since the skills are randomly assigned in the N-model and there is no restriction that there cannot be an overlap, we must expect some overlap, depending on the total number of skills and the number of skills possessed by each generalist. Due to the randomness, the SNOI the N-model will be different in each case, but we can calculate the expected SNOI. If a generalist has k different skills, $\frac{k}{S}$ is the probability that the generalist has one particular skill. If the skills are independently assigned to the agents, the probability that two generalists share a particular skill is $\left(\frac{k}{S}\right)^2$ assuming that both have the same number of skills. The expected overlap of a specific skill is then $\left(\frac{k}{S}\right)^2 \times S$, resulting in

$$SNOI(i,j) = SNOI_N = 1 - \frac{k}{S}.$$

If $k = 3$ and $S = 15$, as assumed above, the expected degree of specialization in the N-model is equal to 0.8 and hence lower than in the H-model. In the HN-model, the same reasoning applies resulting in expected $SNOI_{HN} = 0.8$, too.

As discussed above, Andersson and Törnberg [18] argue that the components of a system with high structural complexity are "enslaved", meaning that they are highly dependent on other components and have little autonomy. In an HSS, "enslavedness" is related to the degree to which individual agents can or cannot make autonomous decisions. I propose the **Decision Centralization Index** (DCI) to measure the degree to which decision-making is concentrated on a few agents, implying a high degree of "enslavedness" of the other agents:

$$DCI = \frac{H_{\max} - H_{Decisions}}{H_{\max}},$$

which depends on the Shannon entropy of all decisions made in the problem-solving process. $H_{Decisions} = -\sum_{i=1}^{n} d_i \log_2 d$ is the observed decision entropy with d_i being the proportion of decisions made by agent i. Decision entropy is maximized, if all agents have the same share of decisions, $d_i = \frac{1}{n} \forall i$, hence $H_{\max} = \log_2 n$. In this case, decision-making is totally decentralized such that no agent is enslaved and $DCI = 0$. In the opposite case, if all decision-making is done by a single agent, $H_{Decisions} = 0$, meaning that there is no dispersion of decision-making and $DCI = 1$.

In the H-model, the manager is the only decision-making agent. The manager decides who to assign tasks to and stops the process after the partial solutions have been assembled into the final solution. The specialists make no decisions. Therefore, in this model, $H_{Decisions} = 0$ and $DCI = 1$ indicate maximum decision centralization. In the N-model, each generalist can make two decisions: (1) to continue or to stop the process, and (2) to whom to pass on the problem (if there are more than one link neighbors). The actual number of decisions an agent makes depends on the process, because due to the cyclical nature of the network, each generalist may receive the problem several times or not receive the problem at all. However, if we consider the chances or the right to make decisions, all generalists are equal, so at least in expectation $H_{Decisions} = 1$ and $DCI = 0$. In the HN-model, the decision centralization also depends on the process, but we can derive a theoretical approximation of the DCI. If with $n = 15$ all agents could make the same number of decisions, we had $H_{\max} = \log_2 15 \approx 3.9$. In the model, however, decisions are distributed unequally. With R rounds of the process, the manager's share of decisions is $p_M = \frac{2}{R+2}$. Assuming that all 7 members of the active team (including the leader) get the problem the same number of times, their share is $p_{TM} = p_{TL} = \frac{R}{7(R+2)}$, hence

$$H_{Decisions} = -\left(\frac{2}{R+2}\log_2\frac{2}{R+2} + 7\frac{R}{7(R+2)}\log_2\frac{R}{7(R+2)}\right).$$

In Section 3, I found that the average number of rounds was $R = 17$, hence we can expect $DCI_{HN} = 0.512$. With $R = 10$ and $R = 30$, we get $DCI_{HN} = 0.354$ and $DCI_{HN} = 0.656$ respectively.

Table 3 summarizes the proposed structural complexity metrics and their values for the three example models. According to each metric, the H-model has a higher degree of structural complexity than the N-model. Not surprisingly, the HN-model shares features of both other models. It has the same hierarchy as the H-model and same degree of specialization as the N-model. Its degree of role differentiation is higher than in the H-model due to the presence of team leaders. The degree of decision centralization is almost exactly in the middle between the completely centralized H-Model and the completely decentralized N-model. Note that although the reasoning for the derivation of the metrics was often similar and the concepts are related, the measures contain different information because they use different data. This may mean that we can approximate some of the metrics with others when the necessary data is not available in practice.

4.2. Measuring dynamic complexity

As discussed earlier, [18] describe the characteristics of HSS that have high dynamic complexity. Some of the properties, such as many system components (but few component classes) at the same organizational level or high redundancy due to high similarity of the components, are related to the structure of the system. Hence, they are captured by the metrics proposed to measure structural complexity, such as the number of levels, role differentiation, and specialization. Low values of these metrics indicate that the system is likely to exhibit dynamic complexity.

However, conceptualizing dynamic complexity in this way implies that it is precisely the opposite of structural complexity which would make it redundant. A more appropriate approach is to view dynamic complexity as the *outcome* of certain structural features of the system, or as a particular *behavior* of the system and its agents. This follows from the definition of a complex system in which non-trivial or disordered interactions lead to emergent outcomes that are difficult to predict. Thus, to measure the dynamic complexity of the system, we need metrics that capture the unpredictability of the outcomes. We could approximate unpredictability by volatility, since the latter may be easier to measure. I therefore focus on two types of dynamic complexity:

- Volatility in the problem-solving process
- Diversity of the solution

Volatility in the problem-solving process means that the same group of agents solves the same problem in different ways in different instances. This cannot happen with the H-model. The manager will always assign the same problem to the same specialists, who will produce exactly the same solution. Therefore, any metric of problem-solving volatility will be zero in the H-model. Due to the randomness involved in who starts working on the model and the sequence of generalists to whom the problem is assigned, problem solving volatility is not zero in the N-model. In the HN-model, the manager will always assign the same (perceived) problem to the same team. Volatility in the problem-solving process hence stems

Table 3. Summary of the metrics of structural complexity.

Aspect	Metric	Data	H-model	N-model	HN-model
Hierarchy	Number of control levels	Network structure	1	0	1
Role differentiation	RDI	Role categories	1.67	0	3.2
Specialization	SNOI	Skills	1	0.8	0.8
Dependence	DCI	Distribution of decision rights	1	0	0.51

https://doi.org/10.1371/journal.pcsy.0000055.t003

from the interactions in the respective team alone, which is analogous to what happens in the N-model. For the sake of brevity, I hence ignore the HN-model in the following and focus on the N-model.

To compute metrics of problem-solving volatility, the N-model must be simulated several times with a given network structure, given skills, and a given problem. Keeping the network, skills, and problem fixed eliminates all randomness except that in the order in which the generalists receive and work on the problem. As a metric of problem-solving volatility, I propose **Path-Length Variability** (PLV)

$$PLV = \frac{\sigma_{PL}}{\mu_{PL}},$$

where σ_{PL} is the standard deviation of the path length over several runs and μ_{PL} is the mean.

The more diverse the set of final solutions is, the more difficult it is to predict, with which solution a system will come up. I propose the **Normalized Solution Diversity Index** (NSDI)

$$NSDI = \frac{\text{Number of unique solutions} - 1}{\text{Total number of runs} - 1}.$$

The total number of runs indicates how often the system works on a problem. Again, to focus on the behavioral aspects of the process and to abstract from the structural properties of the system, it makes sense to keep the network, the skills and the problem fixed, when computing this metric. I subtract 1 in the numerator and the denominator of the *NSDI*, because every system will find at least one unique solution. Normalization ensures that $NSDI = 0$, if the system generates only one unique solution, and $NSDI = 1$, if it generates a unique solution in every run. The H-model has no solution diversity, because for given skills and a given problem, it will always produce the same solution. In general, $NSDI > 0$ in the network model, but the actual value will depend on the parameters of the model.

In the H-model, all metrics are known to be zero, indicating that this model has no dynamic complexity. To analyze the dynamic complexity of the N-model, which results from the random transmission of the problem from agent to agent, I simulate the model 1000 times. The network, the skill distribution, and the problem remain fixed to isolate the effect of the random problem-passing process that generates dynamic complexity.

Since there is a general presumption that dynamic complexity increases with the degree of interconnectedness, I use three different networks to determine the metrics of dynamic complexity. The example network shown in Fig 2 is the baseline, in which agents have three links on average (the network has 10 links in total). Fig 10 shows the two alternative networks: one in which 3 links have been removed so that the average number of links per agent is 2 and another in which 4 links have been added resulting in an average of 4 links per agent. The three networks thus have network densities of 0.333, 0.476 and 0.667.

Table 4 shows the results of 1000 simulation runs for each version of the network. In terms of dynamic complexity, PLV suggests a positive relationship between density and complexity, and NSDI suggests a negative relationship.

In all networks, the minimum path length was 5 and the maximum was 14 (models with 2 and 3 links) or 15 (model with 4 links). Note that a path length of 15 is the maximum with 7 agents and the max-work-constraint set to 3. The means of the path length are 7.158 (2 links), 7.818 (3 links), and 8.842 (4 links). Hence, more connections make longer solution paths more likely, which is plausible because there are more possible paths available. With more paths the stopping criterion *max-work* is reached later on average. The standard deviations of the path lengths increase with more links (1.983, 1.986, 2.159), but since the means increase even more, the Path-Length Variability (PLV) decreases.

The Normalized Solution Diversity Index (NSDI) is designed to capture the dynamic complexity in a system's output also depends positively on network density. In the model with 2 links per agent, about 2.6% of the solutions are unique. This figure rises to 5.7% with three links per agent and to 13.2% in the model with 4 links per agent.

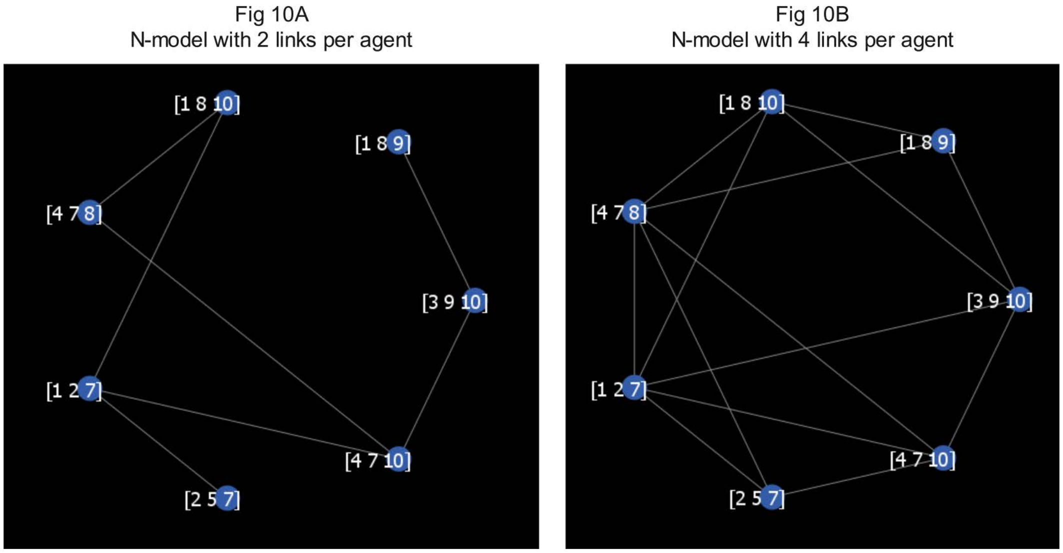
Fig 10A
N-model with 2 links per agent

Fig 10B
N-model with 4 links per agent

Fig 10. Different network structures in the N-model.

https://doi.org/10.1371/journal.pcsy.0000055.g010

Table 4 also shows the actual average values of DCI. This metric of structural complexity is 0 only regarding the right of each generalist to make decisions. In practice, not every generalist is involved in the process, such that the empirical DCI is different from zero. We see that decision concentration is higher in less dense networks, which is plausible. In terms of performance, higher network density leads to better average quality of the final solution, but also to higher total cost. Interestingly, efficiency seems to be inverse-U-shaped, being lower for both low and high network densities than for intermediate densities.

5. Discussion and conclusions

The aim of this paper was to clarify the different conceptions of complexity in human social systems (HSS) that engage in problem-solving. I posed three research questions: How can we characterize the structural and dynamic complexity of problem-solving HSS? What is the relation between the two complexity types with the performance dimensions solution completeness, efficiency and robustness? How can we measure structural complexity and dynamic complexity of HSS? To answer these questions, I developed three simple agent-based models and six metrics of complexity properties.

Structural complexity refers to organizational properties and is characterized by the degree of hierarchy, differentiation of roles, specialization in skills and centralization of decision-making (or dependence). The H-model has high degrees in all these features: there is a manager who makes all decisions and specialist problem-solvers that have single and

Table 4. Simulation results for the 3 versions of the N-model.

Network properties	Average links per agent	2	3	4
	Density	0.3333	0.4762	0.6667
Dynamic complexity	PLV (mean)	.27698612	.25404928	.2441933
	NSDI (mean)	.02602603	.05705706	.13213213
Structural complexity	DCI (mean)	.3779163	.3052814	.2265085
Performance	Solution quality (mean)	2.4164	2.6852	2.9382
	Total cost (mean)	7.158	7.818	8.843
	Efficiency (mean)	0.329	0.339	0.327

https://doi.org/10.1371/journal.pcsy.0000055.t004

unique skills at a maximum skill level. In contrast, the N-model was designed to have low degrees of structural complexity in these dimensions: there is no hierarchy, but a network consisting of agents with equal roles and overlapping skills, and decision-making is (ex ante) fully decentralized. Both systems are highly stylized to work out the dimensions of structural complexity as clearly as possible. The more realistic HN-model has features of both other systems: a hierarchy as well as a network organization, centralized decision-making by a manager and decentralized decision-making in teams of generalists with overlapping skills, and potentially some partial specialization across teams. While it is straightforward to design systems with high or low structural complexity, dynamic complexity cannot be designed directly, because it refers to the behavior of the system, not its organization. The key characteristic of dynamic complexity is that disordered interaction of the system's elements leads to outcomes that are difficult to predict. That said, there are nevertheless organizational features that are more likely to allow dynamic complexity than others. The H-model imposes a strong structure on the interaction of the agents. The specialists have no discretion and do exactly what the manager demands. They are fully "enslaved". The manager will always make the same decisions (absent misperception), such that the system's behavior is perfectly predictable and hence not dynamically complex. Therefore, high structural complexity prevents dynamic complexity from occurring. In the N-model and the HN-model, dynamic complexity arises from the random interaction of the agents that work on the problem. The interaction is not fully random, because the problem is only passed from generalists to generalist along network links imposing structure on the interaction. It is the randomness of the choice of the next agents who will work on the problem that characterizes the disordered interaction and that produces large volatility in the duration of the problem-solving process, its efficiency, and the completeness of the solution across different runs.

The model simulations showed that there is no simple answer regarding the relation between the complexity and the problem-solving performance of a HSS. On average, the H-model with high structural complexity generated more complete solutions than the less structurally complex N-model. However, the HN-model, which is less structurally complex than the H-model solved the problems more completely on average. The H-system's strength is that it solves problems efficiently, i.e., at lowest costs. It is designed to solve problems fast (in one round) and without the involvement of agents that cannot contribute to the solution. The efficiency of the N-model and especially of the HN-model is much lower on average because of the largely disordered interaction between the agents. The HN-model does particularly badly in this respect, because ordinary team members do not decide on when to stop the process. The N-model and the HN-model show a much larger variation in completeness and efficiency than the H-model, which is a sign of dynamic complexity. Due to its high structural complexity, the H-model is prone to errors that can significantly reduce completeness and efficiency. If the manager, as the only decision-making agent, misallocates the problem to the wrong specialists due to misperception, the system's strength turns into a weakness. It is the function of the manager to allocate resources efficiently. If the manager cannot fulfill this function correctly due to internal or external problems, there is no correction mechanism in the system as nobody else is allowed to make decisions and to repair the manager's bad decisions. The N-model is not affected by this kind of failure because it has no centralized decision making. The HN-model is less robust against errors than the N-model, but more so than the H-model, because the manager only makes a general decision on which team should work on the problem but does not control the process fully. My analysis hence confirms the intuitive and widespread notion that there is a tradeoff between efficiency and robustness. Note that the results obtained from simulating these very simple and stylized models are suggestive, but far from comprehensive. To obtain more robust and general results, more sophisticated models that ideally could also be analyzed mathematically are desirable. This is left for future research. The models proposed here could be informative how such models could be designed.

So far, the discussion of the three systems' structural and dynamic complexity was heuristic, because I only described, but did not measure the complexity of the systems. I propose that the degree of structural complexity can be measured by the number of control levels, a role differentiation index, an index of non-overlap of skills and an index of the centralization of decisions. According to all these metrics, the H-model is in fact more structurally complex than the N-model. The HN-model is generally more structurally complex than the N-model (except for the same degree of specialization) and even more complex than the

H-model in terms of role differentiation. To measure dynamic complexity, I propose path-length variability, which captures the unpredictability of the problem-solving process, and the diversity of the final solutions. Since the H-model always generates identical solutions to the same problem within one round, these metrics assign a dynamic complexity of zero to the H-model. In the N-model, the measures are non-zero and depend on the network density. Path-length variability decreases in denser networks, whereas the diversity of the solutions increases. This result suggests that the degree of interconnectedness has an ambiguous role for dynamic complexity. Intuition might say that systems with higher network density generate more dynamic complexity, because there are more possibilities for disordered interaction. The higher diversity of the solution found in denser networks supports this intuition. However, in less dense networks, the process can differ a lot depending on which agent started the problem-solving process. There is no point in discussing which conception of dynamic complexity is the correct one, since both capture a particular aspect of it – variation of the process vs. variation of the outcome. The result highlights the importance of defining clearly and measurably, what is meant when we speak of dynamic complexity.

As argued before, most real-world HSS are likely to be “trans-complicated” or “trans-complex”, i.e., like my hybrid HN-model, they exhibit both some degree of structural complexity and of dynamic complexity. According to [18], trans-complicatedness occurs in structurally complex (or complicated) organizations, in which components have separate agendas. I argue that every formal human organization that does not strictly enforce command structures has this feature. Within every formal organization informal groups self-organize and develop their own goals that are only more or less aligned with the goals of the organization. For example, the owners of a firm want the firm to make profits and ask the management to design an organizational structure that generates output at minimal costs. However, within the firm groups of employees might form different goals, e.g., minimizing stress and workload, pursuing pet projects, outcompeting and even sabotaging other groups within the firm, cultivating friendships and good relationships with others, promoting their own status and reputation and many more. In the HN-model, the control hierarchy that gives some agents the power to begin and end the process and to decide who can be involved corresponds to the designed formal organization. The networks in the teams can be interpreted to represent informal interaction patterns, e.g., based on personal sympathies or status considerations. The random passing of the problem to other team members might represent minimization of private effort or any other motivation that is unrelated to the goal of solving the problem in an efficient way. As it turns out, although these private agendas reduce the efficiency of the process, they nevertheless can produce fairly good (in the sense of complete) solutions, which do not suffer much from wrong decisions by the manager, unless the specialization of the whole team is rather strong. Aligning the goals (or at least the actions) of the agents with the goals of the organization can cause various costs like monitoring and enforcement costs, demotivation of the agents or loss of initiative and robustness. Hence, decreasing dynamic complexity and increasing structural complexity to achieve more efficient processes and more predictable outcomes requires efforts that should be weighed against the potential benefits.

The insights from this paper could be used to perform empirical studies on how the performance of real organizations is related to their complexity. Of course, measuring the structural complexity and dynamic complexity of real organizations is more difficult than in my simple models. Hierarchy and the centralization of decisions are, at least in principle, observable, e.g., from organizational charts or by observing who makes which decisions in an organization. Role differentiation and specialization require human resource data, some of which are proprietary and some of which is publicly available, such as job descriptions or occupational classification data. To measure dynamic complexity, one would have to look at problem-solving processes and their results, possibly in a qualitative way by surveys or self-reporting of involved agents. Based on my finding that the efficiency of the structurally complex H-system decreases significantly in the presence of misperception, I hypothesize that structurally complex organizations should be less successful and hence less present in the long run in uncertain environments. If there is a lot of change or ambiguous data in an environment, networks of generalists might be better in finding appropriate ways to deal with problems than hierarchical organizations that rely on good top-down decisions. In other words, we might expect to find a negative correlation between the presence (and/or performance) of highly structurally complex organizations and the degree of uncertainty or volatility of their environment.

Such evidence could be helpful in finding the optimal degree of complexity for the environment in which an organization operates.

Supporting information

S1 File. Contains sensitivity analyses for the main parameters of the model.
(PDF)

Author contributions

Conceptualization: Michael Roos.

Formal analysis: Michael Roos.

Methodology: Michael Roos.

Software: Michael Roos.

Writing – original draft: Michael Roos.

Writing – review & editing: Michael Roos.

References

1. Tainter JA. The collapse of complex societies. Cambridge: Cambridge University Press; 1988.
2. Barton CM. Complexity, Social Complexity, and Modeling. J Archaeol Method Theory. 2013;21(2):306–24. https://doi.org/10.1007/s10816-013-9187-2
3. Daems D. Social complexity and complex systems in archaeology. London, New York: Routledge; 2021.
4. Tainter JA. Social complexity and sustainability. Ecol Econ. 2006;3:91–103.
5. Tainter JA, Taylor TG. Complexity, problem-solving, sustainability and resilience. Build Res Inf. 2014;42(2):168–81.
6. Taylor TG, Tainter JA. The nexus of population, energy, innovation, and complexity. Am J Econ Sociol. 2016;75(4): 1005–43.
7. Flaherty E. Complexity and resilience in the social and ecological sciences. London: Palgrave Macmillan; 2019.
8. Castellani B, Hafferty FW. Sociology and complexity science. Berlin, Heidelberg: Springer; 2009.
9. Byrne DS, Callaghan G. Complexity theory and the social sciences – The state of the art. London, New York: Routledge; 2022.
10. Roos M. Principles of complexity economics – concepts, methods, and applications. Cham: Springer; 2024.
11. Anguinis H, Gabriel KP. International business studies: are we really so uniquely complex? J Int Bus Stud. 2022;53:2023–36.
12. Hausmann R, Hidalgo C, Bustos S, Coscia M, Chung S, Jimenez J. The atlas of economic complexity. Mapping paths to prosperity. New Hampshire: Puritan Press; 2011.
13. Tainter JA. Problem solving: complexity, history, sustainability. Popul Environ. 2000;22(1):3–41.
14. Hidalgo CA, Hausmann R. The building blocks of economic complexity. Proc Natl Acad Sci U S A. 2009;106(26):10570–5. https://doi.org/10.1073/pnas.0900943106 PMID: 19549871
15. Ladyman J, Lambert J, Wiesner K. What is a complex system? Euro Jnl Phil Sci. 2012;3(1):33–67. https://doi.org/10.1007/s13194-012-0056-8
16. Carlson JM, Doyle J. Highly optimized tolerance: a mechanism for power laws in designed systems. Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics. 1999;60(2 Pt A):1412–27. https://doi.org/10.1103/physreve.60.1412 PMID: 11969901
17. Andersson C, Törnberg A, Törnberg P. Societal systems – complex or worse? Futures. 2014;63:145–57.
18. Andersson C, Törnberg P. Wickedness and the anatomy of complexity. Futures. 2018;95:118–38. https://doi.org/10.1016/j.futures.2017.11.001
19. Edmonds B, Le Page C, Bithell M, Chattoe-Brown E, Grimm V, Meyer R, et al. Different Modelling Purposes. J Artif Soc Soc Simul. 2019;22(3). https://doi.org/10.18564/jasss.3993
20. Wilensky U. NetLogo Software. 1999. Available from: http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.
21. Grimm V, Berger U, Bastiansen F, Eliassen S, Ginot V, Giske J, et al. A standard protocol for describing individual-based and agent-based models. Ecol Model. 2006;198.
22. Roos M. Simple models with different types of complexity. Version 1.1.0. CoMSES Computational Model Library 2025 Mar 1. Available from: https://www.comses.net/codebases/68b87f5e-582b-481e-bff9-c7f3be36ad6e/releases/1.1.0/.
23. Lane D. Hierarchy, complexity, society. In: Pumain D, editor. Hierarchy in natural and social sciences. Dordrecht: Springer; 2006. p. 81–120.