



HAL
open science

A Bargaining-Game Framework for Multi-Party Access Control

Gelareh Hasel Mehri, Benjamin Monmege, Clara Bertolissi, Nicola Zannone

► **To cite this version:**

Gelareh Hasel Mehri, Benjamin Monmege, Clara Bertolissi, Nicola Zannone. A Bargaining-Game Framework for Multi-Party Access Control. 29th ACM Symposium on Access Control Models and Technologies (SACMAT 2024), 2024, San Antonio, United States. 10.1145/3649158.3657033 . hal-04740615

HAL Id: hal-04740615

<https://hal.science/hal-04740615v1>

Submitted on 16 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Bargaining-Game Framework for Multi-Party Access Control

Gelareh Hasel Mehri
Eindhoven University of Technology
Eindhoven, The Netherlands
g.hasel.mehri@tue.nl

Clara Bertolissi
Aix-Marseille University, CNRS, LIS
Marseille, France
clara.bertolissi@univ-amu.fr

Benjamin Monmege
Aix-Marseille University, CNRS, LIS
Marseille, France
benjamin.monmege@univ-amu.fr

Nicola Zannone
Eindhoven University of Technology
Eindhoven, The Netherlands
n.zannone@tue.nl

ABSTRACT

Multi-party access control is emerging to protect shared resources in collaborative environments. Existing multi-party access control models often lack essential features to address the challenges characterizing collaborative decision-making. Collaborative access decision-making requires mechanisms that optimally account for the access requirements of all parties without requiring user intervention at evaluation time. This work fills these gaps by proposing a framework for multi-party access control based on game theory. To this end, we identify the decision factors influencing access decision-making in collaborative environments and propose two bargaining models – a cooperative model and a non-cooperative model – to investigate the impact of different cooperation assumptions on collaborative access decision-making. Our framework ensures fairness by considering the access requirements of all controllers equally, achieves optimality by relying on best response strategies, and guarantees termination. Our evaluation shows that different cooperation assumptions significantly impact the performance and outcome of collaborative access decision-making.

CCS CONCEPTS

• Security and privacy → Access control.

KEYWORDS

Social computing applications, Policy conflict resolution, Collaborative decision-making, Game Theory, Bargaining game

ACM Reference Format:

Gelareh Hasel Mehri, Benjamin Monmege, Clara Bertolissi, and Nicola Zannone. 2024. A Bargaining-Game Framework for Multi-Party Access Control. In *Proceedings of the 29th ACM Symposium on Access Control Models and Technologies (SACMAT 2024)*, May 15–17, 2024, San Antonio, TX, USA. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3649158.3657033>

1 INTRODUCTION

Social computing applications, such as online social networks and collaborative platforms, have experienced significant growth in the last decades. These applications enable users to collaboratively

create, manage, and share resources with other users in real time. Although they provide several benefits for users and organizations, these applications also open new privacy and security issues.

Users typically specify access control policies, determining the accessibility and visibility of their data. However, they might be unaware of who accesses the data as other controllers can grant access. Conflicts in users’ policies can result in undesired disclosure of sensitive information for some users and in reducing sharing utility for others [5, 6, 25, 26]. Traditional access control mechanisms cannot address these challenges as they are grounded on the assumption that resources are governed by single entities. We need a fundamental shift from well-established single-user-centric models to models in which multiple entities can jointly determine with whom the resource can be shared [18].

Multi-party access control is an emerging paradigm for protecting co-owned resources [16, 32]. To guarantee the acceptability of collective access decisions, multi-party access control solutions should (i) capture real-world decision-making dynamics, (ii) ensure fairness of the access decision-making process, and (iii) provide an optimal collective decision. In addition, (iv) they should not rely on user intervention as it is impractical in many situations.

Existing solutions for multi-party access control do not meet these desiderata. Some works rely on predefined rules for conflict resolution [1, 3, 13, 25], thus not accounting for the dynamics of collective decision-making or propose iterative negotiation protocols in which user intervention is required for conflict resolution [14, 27]. A recent stream of research has leveraged game theory as the foundation for the design of multi-party access control solutions [9, 18, 20, 21, 31] since it offers a framework for collective decision-making and understanding the interactions between individuals [28]. Although game theory has the potential to meet the desiderata for multi-party access control, existing solutions based on game theory only account for a limited set of decision factors [18, 21, 31], thus not fully capturing the complex dynamics driving decision-making, do not ensure termination [18, 31], or do not guarantee the fairness of the decision-making process [18, 20, 31].

This work investigates the application of game theory concepts to support decision-making in multi-party access control scenarios. In particular, we propose a bargaining framework, which, given the controllers’ access requirements for a shared resource, computes an optimal collective access decision. Our framework involves an iterative and simultaneous adjustment of all controllers’ access requirements until an agreement is reached. Our notion of optimality leverages several decision factors (*object sensitivity*, *peer*



This work is licensed under a Creative Commons Attribution International 4.0 License.

influence, relationship with other controllers, and sharing benefits) that have been shown in the literature to be highly influential in access decision-making. To account for different strategic environments influencing user behavior in collaborative decision-making, we propose two bargaining models: a cooperative model and a non-cooperative model. We implemented the proposed models and conducted experiments to assess their performance and outcome. The main contributions of the work can be summarized as follows:

- Our framework captures collaborative decision-making dynamics by leveraging a broad range of decision factors, achieves fairness by considering the access requirements of all controllers equally, and provides a collective decision that optimally benefits all controllers involved.
- Our framework ensures convergence to a collective access decision without user intervention. Thus, our framework poses the basis for developing automated access decision-making.
- Our evaluation shows differences in performance and outcome between cooperative and non-cooperative models, providing an understanding of how different strategic environments influence collaborative decision-making.

The remainder of the paper is structured as follows. Sec. 2 presents an overview of multi-party access control and discusses related work. Sec. 3 introduces our game theoretic framework for multi-party access control, and Sec. 4 presents its evaluation. Finally, Sec. 5 discusses the main findings and Sec. 6 concludes the paper.

2 BACKGROUND AND RELATED WORK

This section provides background on multi-party access control and discusses the drawbacks of existing multi-party access control systems, posing the basis for our proposal.

2.1 Multi-Party Access Control

Multi-party access control is an emerging paradigm for the protection of resources in collaborative systems, where resources are jointly owned and managed by multiple entities (hereafter, referred to as *controllers*) [16, 32]. Users often specify access requirements on their resources, determining who can access them and under which conditions. Ideally, when dealing with co-managed resources, all controllers’ access requirements should be enforced. However, the access requirements of a controller can conflict with the ones of other controllers [3, 13]. Failure to resolve these conflicts can result in the leakage of sensitive information [29].

Multi-party access control aims to resolve conflicts arising from evaluating policies authored by different controllers. A large body of research has investigated the problem of multi-party access control, especially in the context of online social networks, and proposed solutions, for example, to find collective non-conflicting policies, resolve policy conflicts at policy evaluation time, or support controllers in reaching a consensual decision. The next section reviews the state-of-the-art in multi-party access control.

2.2 Related Work

Existing solutions for multi-party access control can be categorized according to four main dimensions, namely *design properties* (i.e., design choices and assumptions underlying the proposed solution), *decision factors* (i.e., the factors used for conflict resolution), *policy*

evaluation properties (i.e., properties of the method used for conflict resolution) and *behavioral constraints* (i.e., human behavior patterns influencing decision making). An overview of existing approaches with respect to these dimensions is presented in Table 1.

Design Properties. Existing multi-party access control systems are based on different design properties, which determine the requirements and assumptions under which the system operates. The first block of Table 1 presents an overview of the main design properties and how they relate to the proposed systems.

A key design property is the *decision approach* adopted for multi-party data sharing management [16]. Some proposals rely on rules that predetermine how controllers’ policies are combined and evaluated [1, 3, 13, 25]. For instance, Mahmudlu et al. [13] employ policy combining algorithms, such as *first-applicable*, *permit-overrides* and *deny-overrides*, to determine priorities between controllers’ policies. Other solutions employ aggregation methods, such as voting schemes, to determine how conflicts are resolved [8] or to determine the most desired policy for the shared object [23, 29]. Another stream of research leverages game-theoretic approaches to find an equilibrium between controller’s preferences [9, 18, 20, 21], where no controller has an incentive to unilaterally change its privacy settings. Different classes of game theory have been applied to multi-party access control, e.g. auction-based games in which controllers bid the values they associate with their privacy preferences [21], leader-follower games in which the leader (usually a user with higher authority) first announces its strategy, imposing an upper-bound to data sharing, and then followers select their best response to the leader’s strategy [18], or behavioral games in which users try to maximize their utility [9]. Game theoretic approaches have been combined with an aggregation method in [31], where a preference-adjusting game derives controllers’ choices, and then a voting scheme selects the preferences to be enforced. Interactive policy negotiation protocols have been proposed in [14, 27] to support controllers in reaching an agreement over a shared object. In a similar spirit, Fogues et al. [5] propose a computational model based on machine learning to predict an appropriate access control policy for a given multi-party scenario. An orthogonal approach is proposed by [10], in which the decision is shifted to the individual level. However, this approach does not preserve *object integrity* (i.e., the object is modified as a result of policy enforcement [23]), thus affecting the other controllers’ expected utility.

Existing solutions also differ for the underlying *governance model* for shared resources, which determines the level of authority that controllers have over the shared resource [16]. Two main governance models are typically considered: *multi-ownership*, in which all controllers have the same authority level over the object [8, 10, 14, 31], and *asymmetric*, in which controllers’ control levels depend on their relationship with the object [3, 8, 13, 18]. Among approaches employing an asymmetric model, some assume a predefined set of types of controllers, e.g. owner, contributor, stakeholder, and disseminator in [8] or owner and followers in [18], whereas others support the definition of arbitrary hierarchies of controllers [3, 13].

Solutions for multi-party access control can rely on a *centralized* architecture, in which controllers’ policies are collected and processed in a central system for conflict resolution, or a *distributed* architecture, in which controllers interact with each other to resolve policy conflicts [16]. In Table 1, we report the architecture when a

Table 1: Summary of related work

	Decision Approach	Design Properties				Decision Factors					Policy Eval. Prop.				Rationality
		Governance Model	Architecture	User Intervention	Object Integrity	Object Sensitivity	Sharing Benefits	Peer Influence	Justification	Relationship	Convergence	Optimality	Decision Payoff	Fairness	
Such et al. [25]	R	MO	C	✓	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Akkuzu et al. [1]	R	MO	C	✗	✓	✓	✗	✗	✗	✓	✓	✗	-	✓	-
Mahmudlu et al. [13]	R	A	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Damen et al. [3]	R	A	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Thomas et al. [29]	AG	MO	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Squicciarini et al. [23]	AG	MO	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Hu et al. [7]	AG	MO	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Hu et al. [8]	AG	A	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	-	✓	-
Squicciarini et al. [21]	G	MO	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	SoU	✓	F
Hu et al. [9]	G	MO	C	✗	✓	✓	✗	✗	✗	✗	✓	✗	SeU	✓	F
Rajtmajer et al. [18]	G	A	C	✓	✓	✓	✗	✓	✗	✗	✗	✓	SE	✗	F,B
Squicciarini et al. [20]	G	MO	C	✓	✓	✓	✗	✓	✗	✗	✓	✗	-	✗	F
Xiao et al. [31]	G+AG	MO	C	✓	✓	✓	✗	✓	✗	✓	✓	✗	EP	✗	F
Such et al. [27]	NP	MO	C	✓	✓	✓	✗	✗	✗	✗	✗	✓	ED	✓	F
Mehregan et al. [14]	NP	MO	-	✓	✓	✓	✗	✗	✗	✗	✗	✗	-	✓	-
Fogues et al. [5]	ML	MO	C	✗	✓	✓	✗	✗	✓	✓	✓	✗	-	✗	F
Ilija et al. [10]	-	MO	C	✓	✗	✗	✗	✗	✗	✗	✓	✗	-	✓	-
Our work	G	MO	C	✗	✓	✓	✓	✓	✗	✓	✓	✓	De	✓	F

Legend:

Decision Approach R: Rule-based AG: Aggregation G: Game Theory NP: Negotiation Protocol ML: Machine Learning
 Governance Model MO: Multi-Ownership A: Asymmetric
 Architecture C: Centralized D: Distributed
 Decision Payoff EP: Emotional Payoff SoU: Social Utility SeU: Selfish Utility ED: Exception Distance SE: Social-Energy De: Desirability
 Rationality F: Full B: Bounded
 For other criteria, '✓' denotes that the criterion is supported/satisfied by the solution and '✗' that it is not. Symbol '-' denotes that the criterion does not apply to the solution.

prototype implementation is presented or the type of architecture is explicitly discussed; otherwise, it is marked as not-applicable ('-'). From Table 1, we observe that most considered solutions rely on a centralized architecture. For instance, they assume that conflict resolution is performed by a mediator or a centralized policy engine [3, 13, 25], or provide a prototype implementation that operates on top of existing social networks such as Facebook [7, 21, 23].

Existing solutions also differ for the level of *user intervention* required to solve policy conflicts. Some proposals, typically based on rules, aggregation methods, and machine learning, provide automated conflict resolution mechanisms to reduce the burden on the controllers [1, 3, 8, 23]. Other approaches require some form of user intervention during the decision-making process. In particular, some approaches require controllers to adjust their policies to achieve consensus [14, 18, 31]; others only provide controllers with a suggestion to resolve policy conflicts and require controllers to confirm such a suggestion [25, 27], possibly leading to manual negotiation if consensus is not reached.

Decision Factors. Several factors might influence the willingness of controllers to change their preferences to reach consensus [4, 17]. Table 2 reports the *decision factors* commonly considered in the context of multi-party access control, and the second block of Table 1 presents an overview of their adoption in the literature.

An intrapersonal factor often measured to determine controllers' willingness to change their preferences is *object sensitivity* [5, 7, 8], which models the controller's perception of the importance of the object. This perception can influence the extent to which controllers require their constraints to be considered in decision-making: higher is the perceived importance, the higher the controllers' tendency to opt for the decision closest to their initial preferences [5]. A few works consider controllers' perceived *sharing benefits* [7, 9, 14, 21], for instance, measured as the number of

Table 2: Decision Factors

Factor	Description
Object Sensitivity	Perceived importance towards the shared object [7].
Sharing Benefits	Perceived advantages obtained by making the shared object available to others [33].
Peer Influence	Influence that a group has on individual members of that group such that they change their attitude and behavior to match the one of the group [12].
Justification	Influence of reasoning behind other controllers' preferences and their perceived consequences [5].
Relationships	Influence of interpersonal relations with other controllers [5].

users to which the object is visible. This factor is related to the expectations of controllers and has been shown to have a significant influence on group decision-making [33].

Interpersonal factors, such as *peer influence* [9, 18] and *justifications* [5], also influence a controller's willingness to change their preferences. Solutions that consider peer influence typically assume that controllers know each other's privacy preferences and that they might change them to behave in accordance with the norms of their social group [9, 18, 31]. Justifications represent subjective arguments a controller provides to express the reasoning behind their specific preferences and the perceived consequences of sharing a certain piece of information. As shown in [5], justifications can persuade other controllers to adjust their preferences in the direction of the controller's preferences.

Another interpersonal factor that is considered in conflict resolution is represented by the *relationships* with the other controllers [1, 5, 9, 31]. These relationships affect how each controller perceives the multi-user scenario and, particularly, influence the extent a controller respects the opinion of other controllers [5]. Some works model and measure interpersonal relationships quantitatively, e.g. as trust level [1, 9, 31], whereas others model them qualitatively, such as friend or colleague [5].

Table 3: Policy Evaluation Properties

Property	Description
Convergence	Policy evaluation terminates and a non-conflicting result is returned in a reasonable time [14].
Optimality	Policy evaluation returns the decision that is the most beneficial for all controllers [5].
Decision Payoff	Controllers' expected gain/loss for a decision [9, 21].
Fairness	Controllers should not be able to influence policy evaluation in a way that exceeds their level of authority [25].

Policy Evaluation Properties. In the context of multi-party access control, policy evaluation should ensure that a non-conflicting decision is always reached and that such a decision is acceptable for all controllers [5, 31]. To assess to what extent and how these requirements are addressed by existing solutions, we have identified four main properties, as described in Table 3. An analysis of the literature w.r.t. these properties is reported in the third block of Table 1.

A key property for policy evaluation is that a non-conflicting decision should be always obtained in a reasonable time (*convergence*) [14]. This property is typically achieved by rule-based [1, 3, 13], aggregation-based [7, 8, 29] and machine learning-based [5] approaches, which provide a mean to automatically resolve policy conflicts. Other approaches such as game theoretic approaches [9, 18, 31] and negotiation protocols [14, 27] aim to find a consensual decision rather than merely resolving policy conflicts. However, these approaches often provide only conditions under which convergence exists [14, 31] but do not ensure convergence. In general, convergence is not guaranteed by approaches that require user intervention (e.g., adjusting their preferences) [14, 18, 31] or require controllers to accept the suggested collective policy [25].

The acceptability of collective decisions is typically addressed by game theoretic approaches which aim to find a decision that is the most beneficial for all controllers according to a given *payoff* function (*optimality*). Several payoff functions have been proposed: some aim to maximize the utility controllers would gain as a group [21], whereas others aim to maximize individuals' utility [9]. Some payoff functions also account for controllers' tendency to maintain their initial privacy preferences [18, 27, 31] or aim to minimize the number of exceptions in the resulting privacy policy so that the solution is as understandable as possible for users [27].

To foster the acceptability of collective decisions, policy evaluation should also guarantee the *fairness* of the decision-making process by accounting for the preferences of all controllers [8, 23, 31]. In solutions employing multi-ownership models, fairness implies that all controllers should influence the decision-making process equally [23]. On the other hand, in solutions employing an asymmetric data governance model, where controllers are assumed to have a different level of authority on the shared object, fairness should also account for such differences. From Table 1, we can observe that most approaches guarantee the fairness of the conflict resolution process. An exception is the proposal in [7] in which the data owner can adjust the weights used to determine the trade-off between data sharing and privacy loss; while this can help effectively resolve policy conflicts, the data owner can influence the conflict resolution process to the detriment of other controllers. Approaches based on machine learning such as [5] might also not satisfy fairness. Bias in the data used to train the classifier can lead to the controllers' preferences to be not equally representative.

Behavioral Constraints. A few works have investigated human behavioral patterns and biases influencing access decision-making in multi-party scenarios and how these can be integrated into policy evaluation to approximate human behaviors [18, 21, 31]. In particular, they considered how *rational* users act for their decisions. The last block of features in Table 1 reports for each proposal the adopted rationality model. For solutions in which policy evaluation is performed by the system without any regard for humans' behavioral patterns, we consider the feature *not applicable* ('-').

Most game-based approaches assume full rationality, where users choose their strategy to find an outcome that maximizes utility [9, 21, 27]. Rajtmajer et al. [18] observe that the rationality of individuals in decision-making might be limited, e.g., by their knowledge and beliefs. To account for this, they propose a bounded rationality model for multi-party access control to account for controllers' cognitive limitations. A different approach to emulate human decision-making is proposed in [5], which employs a classifier to predict the optimal collective policy.

2.3 Discussion

Several works have investigated the problem of multi-party access control. A stream of research has proposed solutions that require user intervention for conflict resolution [14, 18, 31], which however is impractical and unacceptable in several situations [22, 31] and does not guarantee to reach an agreement between controllers. A few studies have proposed game theoretical approaches to simulate group dynamics and human interactions, thereby enabling the prediction of group access decisions that optimize collective utility. Group decision-making is a complex human interaction and human cognition process, where several factors can influence the dynamics between users and group decision-making. However, existing approaches for multi-party access control tend to focus only on a limited number of decision factors, only partially capturing the dynamics and complexity inherent in group decision-making.

To address these gaps, we investigate game theoretical approaches for multi-party access control, which account for a comprehensive set of decision factors at play in group decision-making. By doing so, our solution offers a robust solution to group access decision-making, paving the way for effective and acceptable collective decisions in practical scenarios.

3 A GAME THEORETIC MODEL FOR MULTI-PARTY ACCESS CONTROL

We propose a framework for multi-party access control based on game theory. This section provides preliminaries on game theory and introduces our framework. Based on this framework, we propose two models for collaborative access decision-making, which rely on different cooperation assumptions.

3.1 Preliminaries on Game Theory

Game theory is a theoretical framework proposed to analyze strategic interactions between agents, in which the outcome of an agent's choices depends on the ones of others. Each agent (or *player*) has a set of possible *strategies* they can select from to act at any point in the game. The strategies selected by the players determine the *outcome*. The game outcome is computed for each player using

a *payoff function*, which accounts for the strategies chosen by both the player and all other players. Indeed, payoffs’ interdependence is what distinguishes a game setting from optimization problems [28].

Equilibria are typically employed in game theory as analysis tools to study the behavior of players. An *equilibrium* is a state of the game in which all internal forces of the game balance each other. The equilibrium typically adopted in game theory is the Nash equilibrium, which denotes a state of the game in which no players can benefit by unilaterally changing their strategy. At the Nash equilibrium, all players select their *best response* strategy, which leads to the highest payoff for a player given the strategies of all other parties.

A game has underlying assumptions describing the characteristics of the environment and players. The players are assumed to have a set of behavioral capacities, referred to as *rationality*. A fully rational player is expected to select the strategies that yield the most preferred outcomes. Bounded rationality, conversely, leads to satisfactory decisions (in contrast to optimal decisions) [18]. Games also differ in *playing sequence*, i.e., the order in which players make decisions. All players make decisions simultaneously in simultaneous-move games, while sequential-move games are played in turns.

In this work, we focus on a particular type of games, namely *bargaining* games, which applies to situations where players have a common interest, i.e., reaching a consensus is preferable to *no agreement*. However, they might have conflicting interests [2]. As two alternative interpretations, bargaining interactions can be analyzed either through a *cooperative* perspective, in which prescriptive rules are employed to obtain a desirable agreement for all players, or a *non-cooperative* perspective, in which a descriptive analysis of players’ moves are used to determine their best response [11]. It is worth noting that cooperative and non-cooperative do not refer to the degree of cooperation between the parties; rather, they are alternative ways of facing the strategic environment.

3.2 Multi-party Access Control Framework

We model the decision-making process in multi-party access control as a game framework that simulates the interactions between controllers. This section presents our game framework along with the underlying assumptions and its representation.

Assumptions. We assume that users participating in the multi-party access control game are *fully rational*. This is because we aim to design a multi-party access control mechanism that computes the “best” access decision based on the controller’s access control policy rather than an environment to simulate their interactions. We also assume that controllers play *simultaneously*, i.e., there is no specific order in which the strategies of the controllers are applied in the decision-making process. Any notion of sequential behavior, although possible, would introduce inherent asymmetry in the authority that controllers have on the shared object.

Game Graph. We represent the game settings for the multi-party access control mechanism as a graph, called *game graph*. A game graph $G = (S, E)$ is a directed graph, where S is the set of states and $E \subseteq S \times S$ is the set of edges. For an n -player game, a state is modeled as a tuple $\mathbf{P} = (P_1, \dots, P_n)$, where P_i denotes the access control policy of controller i at state \mathbf{P} . For the purpose of this work, we do not introduce a specific policy specification language; instead, we assume that a controller specifies the set of users to whom the

controller allows access to a given object (hereafter we refer to this set as *access preferences*). Edges represent transitions between states and are labeled with an n -tuple $e = (\sigma_1, \dots, \sigma_n)$, where σ_i denotes the action performed by controller i at a given point of the game. Controllers’ actions are determined by the strategies they play, from a given strategy space Σ . Intuitively, a strategy represents how a controller could update her access preferences. The next state is determined by combining the strategies of all controllers. Note that our framework does not impose specific constraints on the strategies the controllers can select; we only require they operate on sets (see Example 1 for sample strategies).

EXAMPLE 1. *Let us consider a two-player multi-party access control game played by Alice and Bob where P_a and P_b denote their initial access preferences, respectively. We assume a strategy space comprising three strategies: $\Sigma = \{\text{con}, \text{per}, \text{res}\}$. The conservative strategy (con) models situations in which a controller prefers to keep her preferences as they are. The permissive strategy (per) models the cases where a controller is willing to relax her preferences by granting access also to the users allowed by other controllers; this strategy can be modeled as the union (\cup) of the controllers’ preferences. The restrictive strategy (res) models the cases where a controller is willing to restrict the accessibility of the object to the users that both controllers grant access; this strategy can be modeled as the intersection (\cap) of controllers’ preferences.*

An excerpt of the game graph for the two-player game with strategy space Σ and initial state $\mathbf{P}^0 = (P_a, P_b)$ is shown in Fig. 1. The edges correspond to the application of the strategies of the two controllers. For simplicity, we do not introduce dedicated labels to represent action; instead, we annotate edges with the strategy that induced the action. It is worth noting that our game graph admits loops. They can occur, for instance, when both players apply the conservative strategy (represented by edges with label [con, con]), or through traversing several states (e.g., after moving from \mathbf{P}^3 to \mathbf{P}^{10} , applying strategies [res, per] would bring the system back to \mathbf{P}^3).

A state is *terminal* if all controllers of a shared object have an agreement about granting access to a given user. To evaluate an access request against the controllers’ preferences, we introduce the evaluation function *eval*. Formally, given an access request issued by a user r for an object and the access preferences P specified by a controller for that object, the function *eval* is defined as:

$$\text{eval}(P, r) = \begin{cases} \text{Permit} & \text{if } r \in P \\ \text{Deny} & \text{otherwise} \end{cases} \quad (1)$$

Based on Eq. 1, we say that a state $\mathbf{P} = (P_1, \dots, P_n)$ is terminal for an access request issued by a requester r iff:

$$\text{eval}(P_1, r) = \text{eval}(P_2, r) = \dots = \text{eval}(P_n, r) \quad (2)$$

It is worth noting that the exact set of terminal states can only be determined based on the given access request. Nonetheless, it is trivial to observe that states in which the preferences of all controllers are the same are always terminal states regardless of the given access request. Hereafter, we refer to these states as *sink* states. For instance, states \mathbf{P}^4 , \mathbf{P}^8 , and \mathbf{P}^9 in the game graph of Fig. 1 are sink states. We can observe that, with the strategy space defined in Example 1, the application of any (combination of) strategies to a sink state results in a self-loop.

In this work, we restrict our focus to multi-party access control games whose graph is admissible. A game graph is *admissible* if

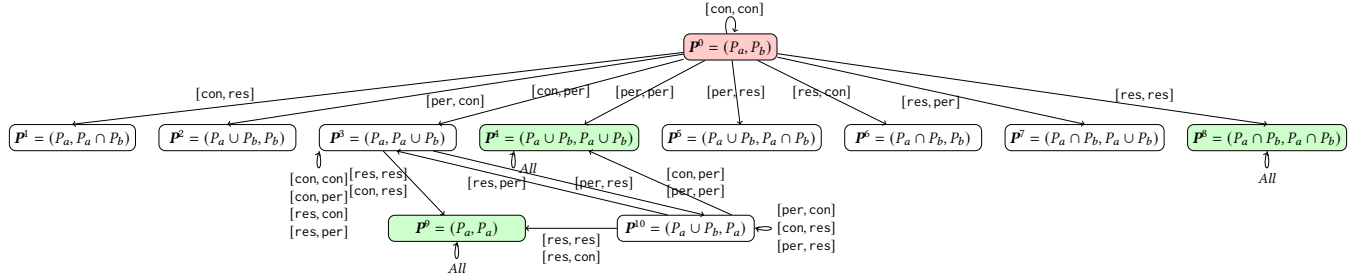


Figure 1: Game graph for a two-player case with three strategies. The label ‘All’ indicates all combinations of possible strategies.

the following conditions hold: (1) There is at least one sink state in the game graph. (2) There is at least one directed path to a sink state from every state in the game graph. Note that this restriction does not affect the generality of our approach. For example, games in which the strategy space includes strategies that account for the preferences of all controllers equally (such as strategies *res* and *per* in Example 1) ensure the existence of both sink states and a one-step path from each state to a sink state. Strategies that involve the preferences of a subset of controllers also guarantee the existence of sink states, but multiple steps might be needed to reach a sink state from a given state.

Payoff Function. For our game framework, we employ a payoff function representing the *desirability* of a state for a controller, i.e., how agreeable an outcome of the game is for the controller. Several *decision factors* can influence the willingness of a controller to change their preferences to reach a consensus (cf. Table 2). The payoff function should account for these factors to reflect the actual decision-making process in multi-party access control. Note that some factors require comparing the preferences of a controller with those of other controllers. Thus, the payoff function should not consider only the preferences of the controller for which the outcome is assessed but also the preferences of all other controllers.

Based on these observations, we determine the payoff of our multi-party access control game for a controller in terms of the decision factors identified in Section 2.2.¹ Specifically, we define a payoff function that computes the desirability of a state $\mathbf{P} = (P_1, \dots, P_n)$ according to a controller i as follows:

$$u_i(\mathbf{P}) = s_i \cdot Jac(P_i, P_i^0) + \frac{1}{N-1} \sum_{j \in \{1 \dots n\}, j \neq i} rel_{ij} \cdot Jac(P_i, P_j) + sb_i \cdot |P_i| + \frac{pe_i}{N-1} \sum_{j \in \{1 \dots n\}, j \neq i} Jac(P_i, P_j) + \epsilon \quad (3)$$

where $s_i, rel_{ij}, sb_i, pe_i \geq 0$, and $\epsilon > 0$ is an arbitrary small positive constant that serves to ensure termination of the bargaining models.

The first term represents a controller’s willingness to maintain her initial preferences P_i^0 based on the perceived object sensitivity s_i (the more sensitive an object is, the higher s_i is). Intuitively, this term denotes that a controller more likely tends to prefer her initial preferences for objects perceived as highly sensitive, decreasing the payoff when its similarity with the updated preferences decreases. To compute the similarity between the updated preferences P_i and

her initial preferences P_i^0 , we use the *Jaccard index*, a standard measure for the similarity between two sets, defined as:

$$Jac(P_i, P_i^0) = \frac{|P_i \cap P_i^0|}{|P_i \cup P_i^0|}$$

The second term represents the influence of interpersonal relationships on the payoff, such that a controller is expected to gain a higher payoff when her preferences are more similar to the ones of strongly related controllers. Coefficient rel_{ij} denotes the strength of the relationship between i and j . In particular, the stronger the relationship (e.g., friends vs. acquaintances), the higher rel_{ij} is.

The third term represents the contribution of sharing benefits to the payoff. Coefficient sb_i denotes the importance of sharing an object with a large audience for controller i . For example, an influencer could gain higher profit from sharing than a regular user.

Finally, the last term is used to account for peer influence on the payoff. In particular, this term measures the similarity between a controller’s preferences and the group norm (i.e., the preferences of all other controllers).² Coefficient pe_i is an indicator of the extent controller i is subject to the influence of her peers. Similarly to the first term, we employ the Jaccard index to calculate the similarity between the access preferences of two controllers.

Note that the payoff function in Eq. 3 should be defined for each controller who can weigh the factors differently (by setting parameters s_i, rel_{ij}, sb_i , and pe_i). Further discussion is provided in Sec. 5. **Bargaining Frameworks.** Controllers’ access preferences might conflict, where some controllers allow a user to access the shared object while others do not. To reach a consensus on whether access should be granted, controllers might have to adjust their preferences. Indeed, if no agreement is reached, the enforced decision might carry low utility for controllers, for example, leading to not sharing the object at all, or unbalanced empowerment between controllers. This provides a natural context for applying bargaining frameworks in multi-party access control, where the controllers adjust their preferences to reach an agreement.

We propose two bargaining models, a *cooperative* model (Section 3.3) and a *non-cooperative* model (Section 3.4), for our multi-party access control game framework. The cooperative model emulates a group decision-making process by aggregating the payoffs of the individual controllers into a group-representative payoff using a prescriptive rule (hereafter referred to as *bargaining rule*). At each game step, the model selects the state that optimizes these

¹We do not consider the *justification* factors in this work. This factor expresses the reasoning behind preferences, which cannot be expressed as a numerical value and, thus, easily integrated into the payoff function. A possibility is to represent this factor as a constraint to the controllers’ optimization objectives and transform the problem into a constrained game. We leave the study of this solution for future work.

²Note that peer influence and interpersonal relationships measure different aspects. While interpersonal relationships account for cases where a controller changes her preferences towards the ones of a closely related individual, peer influence captures the attitude toward group dynamics.

values until a group agreement (represented by a terminal state) is reached. Conversely, the non-cooperative model aims to determine each controller’s best response by focusing on individual choices. To this end, it relies on a descriptive analysis of the decision-making processes obtained by simulating the multi-party access control game to identify, for each controller, the preferences that optimize her payoff until an equilibrium is reached.

It is worth noting that the bargaining procedure may not terminate since the game graph can have loops. To prevent the procedure from entering an infinite loop and thus guarantee termination, we introduce a discount factor ρ (where $0 < \rho < 1$). In the cooperative model, the discount factor is used to reduce the collective payoff of a state based on the number of times the state is visited. On the other hand, the non-cooperative model utilizes the discount factor to reduce the payoff of a state for a controller based on the number of times the controller chooses a specific preference. We present the two bargaining models in the following sections.

3.3 Cooperative Model

The cooperative bargaining model relies on a collective approach to decision-making. For every possible outcome of the game, a collective value is calculated based on the payoffs of all controllers, and the outcome with the highest value is chosen as the solution of the game. This could be interpreted as if the controllers *jointly* select the outcome that is more desirable for all of them [2].

To support cooperative decision-making, we employ a *bargaining rule* that specifies the aggregative selection procedure. Intuitively, a bargaining rule takes the payoffs of all controllers as input and returns a collective value. Several bargaining rules have been proposed in the literature, motivated by different applications. Well-known bargaining rules are the Nash rule, Kalai-Smorodinsky rule, Egalitarian, and utilitarian rule [15]. Among them, we employ the utilitarian rule, which selects the outcome that maximizes the sum of individual payoffs, since it satisfies crucial properties for multi-party access control (cf. Section 2), namely optimality (it provides Pareto optimal solutions) and fairness (payoffs of all controllers are included in the bargaining rule).

Algorithm 1 presents the cooperative bargaining process. The algorithm takes as input a game graph (S, E) and the access request (issued by a user r) to be evaluated and returns the access decision to be enforced. For each state \mathbf{P} , the algorithm keeps a counter $k_{\mathbf{P}}$, initialized to 0 (lines 1 to 3), recording the number of times \mathbf{P} has been visited. The game starts from the initial state \mathbf{P}^0 , which comprises the controllers’ initial access preferences, and traverses the game graph until a terminal state is reached. While the current state (denoted by \mathbf{C}) is not terminal (cf. Eq. 2), the algorithm increases the counter $k_{\mathbf{C}}$, and then uses the bargaining rule to compute the collective payoffs for each state connected to \mathbf{C} via a direct edge (represented by $R(\mathbf{C}) = \{\mathbf{P} \mid (\mathbf{C}, \mathbf{P}) \in E\}$). The state providing the maximum collective payoff is then selected as the next state to be visited (line 8). If two or more states have equal sum values, one is selected randomly.

To prevent entering an infinite loop, the bargaining rule uses the discount factor ρ to reduce the cumulative payoff provided by a state \mathbf{P} based on the number of times it was visited (recorded by $k_{\mathbf{P}}$). Formally, given the set of states reachable by the current state

Algorithm 1: Cooperative Bargaining

Data: Game graph (S, E) , access request issued by user r
Result: access decision

- 1 **forall** $\mathbf{P} \in S$ **do**
- 2 | $k_{\mathbf{P}} \leftarrow 0$
- 3 **end**
- 4 $\mathbf{C} \leftarrow \mathbf{P}^0$;
- 5 **while** \mathbf{C} is not a terminal state **do**
- 6 | $k_{\mathbf{C}} \leftarrow k_{\mathbf{C}} + 1$;
- 7 | $\mathbf{P}^* \leftarrow \arg \max_{\mathbf{P} \in R(\mathbf{C})} \rho^{k_{\mathbf{P}}} \cdot \sum_{i=1}^n u_i(\mathbf{P})$;
- 8 | $\mathbf{C} \leftarrow \mathbf{P}^*$;
- 9 **end**
- 10 **return** $eval(C_i, r)$

\mathbf{C} , the bargaining rule is defined as follows:

$$\mathbf{P}^* = \arg \max_{\mathbf{P} \in R(\mathbf{C})} \rho^{k_{\mathbf{P}}} \cdot \sum_{i=1}^n u_i(\mathbf{P}) \quad (4)$$

Note that in Eq. 4, controllers are assumed to have equal bargaining power. Accounting for asymmetric power can be easily achieved, for instance, by adding a user-specific weighting to the payoff contribution of each controller.

The algorithm terminates when a terminal state is reached by returning the access decision corresponding to the reached terminal state (line 10). Recall that, in a terminal state, the evaluation of the access preferences of all controllers yields the same access decision; therefore, it is irrelevant which one is selected to determine the access decision to be enforced.

EXAMPLE 2. Consider the game graph in Fig. 1. Assuming that the initial state \mathbf{P}^0 is not terminal, the bargaining rule calculates the collective payoff for all states directly connected to \mathbf{P}^0 and selects the one that maximizes the aggregated value, e.g., \mathbf{P}^3 . Algorithm 1 checks the termination condition for \mathbf{P}^3 and, assuming it is not satisfied, the bargaining rule is applied again to that state to find a state with the highest collective payoffs in $R(\mathbf{P}^3)$, e.g., \mathbf{P}^9 . Note that both $[\text{res}, \text{res}]$ and $[\text{con}, \text{res}]$ edges would perform the same transition. State \mathbf{P}^9 is a terminal state and, thus, Algorithm 1 terminates by returning the corresponding access decision.

Algorithm 1 always terminates when the game graph is *admissible*. Here, we present the main result and the intuition of the proof.

THEOREM 1. Consider a multi-party access control game such that the game graph is *admissible*. The cooperative bargaining approach, as defined in Algorithm 1, always terminates.

A sufficient condition for termination of Algorithm 1 is to reach a sink state. When the state selection process enters a loop, using the discount factor in the bargaining rule ultimately breaks out of the loop. Indeed, after a certain number of loop iterations, the payoff of the states forming the loop decreases to a point where the bargaining rule selects a state outside the loop. Since no loop is iterated infinitely and each state has a path to a sink state, the algorithm eventually reaches a sink state and terminates. It is worth noting that Algorithm 1 may terminate before reaching a sink state; indeed, it is sufficient that a terminal state is reached for the algorithm to terminate (Recall that sink states are a special case of terminal states).

3.4 Non-cooperative Model

Our non-cooperative model relies on the standard game-theoretic notion of Nash equilibrium as the solution of the multi-party access control game. At each step of the game, the controllers simultaneously play their best response strategies based on the *latest* preferences of the other controllers until a Nash equilibrium is reached. In this respect, the Nash equilibrium can be interpreted as a self-enforcing solution by rational players where it is in each player's interest to maintain their selected access preferences [28].

Algorithm 2 presents the non-cooperative bargaining procedure. As for the cooperative model presented in Section 3.3, the algorithm takes as input a game graph (S, E) and the access request (issued by a user r) to be evaluated and returns the access decision to be enforced. For each controller i , the algorithm uses a counter k_{P_i} to keep track of how many times the controller has selected P_i . These counters are initialized to 0 (lines 1 to 5). The current state \mathbf{C} is initialized to state \mathbf{P}^0 , which comprises the initial access preferences of the controllers for the requested object (line 6).

The algorithm checks if the current state is an equilibrium (line 7). Intuitively, the game converges to the Nash equilibrium when all controllers keep their current access preferences, indicating that such preferences result in the highest payoff for them with respect to the preferences of the other controllers. Formally, the current state \mathbf{C} is a Nash equilibrium if the following condition holds:

$$\forall i \in \{1, \dots, n\} \forall P_i \in \Pi_i(\mathbf{C}) \quad \rho^{k_{C_i}} \cdot u_i(\mathbf{C}) \geq \rho^{k_{P_i}} \cdot u_i(\mathbf{C}_{-i}, P_i) \quad (5)$$

where $\Pi_i(\mathbf{C}) = \{P_i \mid \mathbf{P} \in R(\mathbf{C})\}$ represents the set of possible access preferences that controller i could obtain by applying a strategy to \mathbf{C} and $u_i(\mathbf{C}_{-i}, P_i)$ is the payoff of the current state in which the access preferences of controller i are replaced with P_i . Similar to the bargaining rule in Eq. 4, we use the discount factor ρ to prevent entering an infinite loop but this time we consider the number of times preferences have been chosen by the controller.

If the current state is not an equilibrium, the counter k_{C_i} is updated (line 9), and the algorithm computes the best response for every controller based on the preferences of the other controllers in the current state (lines 8 to 11). Given the current state \mathbf{C} , the resulting preferences for controller i through the application of the best response strategy are calculated as follows:

$$P_i^* = \arg \max_{P_i \in \Pi_i(\mathbf{C})} \rho^{k_{P_i}} \cdot u_i(\mathbf{C}_{-i}, P_i) \quad (6)$$

Intuitively, each controller's best response is selected among the set of possible preferences that can be obtained by applying a strategy to the current state, $\Pi_i(\mathbf{C})$, in such a way that it maximizes the controller's payoff assuming that other players maintain their preferences, i.e., $u_i(\mathbf{C}_{-i}, P_i)$. After the best response of every controller is computed, the best responses of all controllers are combined to update the current state (line 12)³. If two or more preferences lead to equal payoffs for a controller, the preferences used to determine the new current state are selected randomly. The algorithm continues until an equilibrium is reached.

At equilibrium, all controllers are set with the preferences that result in the highest payoff for them according to the preferences of other controllers. If the equilibrium corresponds to a terminal

Algorithm 2: Non-cooperative Bargaining

Data: Game graph (S, E) , access request issued by user r
Result: access decision

```

1 foreach player  $i$  do
2   foreach access preferences  $P_i$  do
3      $k_{P_i} \leftarrow 0$ 
4   end
5 end
6  $\mathbf{C} \leftarrow \mathbf{P}^0$ ;
7 while  $\mathbf{C}$  is not a game equilibrium do
8   foreach player  $i$  do
9      $k_{C_i} \leftarrow k_{C_i} + 1$ ;
10     $P_i^* \leftarrow \arg \max_{P_i \in \Pi_i(\mathbf{C})} \rho^{k_{P_i}} \cdot u_i(\mathbf{C}_{-i}, P_i)$ 
11  end
12   $\mathbf{C} \leftarrow (P_1^*, \dots, P_n^*)$ ;
13 end
14 if  $\mathbf{C}$  is terminal then
15   return  $eval(\mathbf{C}, r)$ 
16 else
17   return  $MAJ(\mathbf{C})$ 
18 end

```

state, the algorithm returns the associated access decision (line 15). Otherwise, if the equilibrium does not correspond to a terminal state, this would mean that a consensus cannot be reached. Intuitively, such a state indicates that *the controllers agree that they disagree*. In this case, we use majority voting to determine the final decision (line 17). In particular, given a state \mathbf{P} , function $MAJ(\mathbf{P})$ returns *permit* if the majority of controllers would grant access to the requester based on their preferences at \mathbf{P} ; otherwise, the access decision is *deny* (ties are resolved in a *deny* decision).

EXAMPLE 3. *Let us consider the game graph in Fig. 1. At \mathbf{P}^0 , Alice selects the preferences (amongst her possible preferences in the states directly connected to \mathbf{P}^0) that maximize her payoff while Bob is assumed to maintain his preferences from \mathbf{P}^0 unchanged. Similarly, the best response for Bob is computed while Alice is assumed to maintain her preferences from \mathbf{P}^0 unchanged. Assume that Alice's best response is \mathbf{P}_a (obtained through the conservative strategy *con*) and Bob's best response is $\mathbf{P}_a \cup \mathbf{P}_b$ (obtained through the permissive strategy *per*). Combining these preferences results in the selection of \mathbf{P}^3 as the next state through edge $[\text{con}, \text{per}]$. Suppose now that \mathbf{P}^3 is an equilibrium. If \mathbf{P}^3 is also terminal (recall that, besides trivial cases where the preferences of all controllers are the same, determining whether a state is terminal depends on the actual access request), Algorithm 2 returns the corresponding access decision. Otherwise, if \mathbf{P}^3 is not terminal, the access decision to be enforced is determined using majority voting based on the controllers' access preferences at \mathbf{P}^3 .*

It is worth noting that reaching an equilibrium is a sufficient condition for the termination of the non-cooperative model. Depending on the strategy space, the while condition at line 7 could be relaxed by requiring the current state to be either an equilibrium or a terminal state for exiting the loop, thus speeding up convergence. For instance, this would be the case for the strategy space introduced in Example 1. Since those strategies are based on the union and intersection operators over sets, if, at a given state, the requester

³Note that while creating the game graph, all possible combinations of strategies are considered to create the connected states to each state. Therefore, the combination of best response preferences is necessarily among the states connected to a current state.

belongs to the preferences of all controllers, the application of any combination of strategies results in a state in which the user still belongs to the preferences of all controllers, thus yielding decision *permit*. Similar reasoning applies to the case where a user does not belong to the preferences of any controller.

Algorithm 2 always reaches equilibrium and terminates when the game graph is admissible. Next, we present the main results along with the intuition of the proof.

THEOREM 2. *Consider a multi-party access control game such that the game graph is admissible. The non-cooperative bargaining approach, as described in Algorithm 2, always terminates.*

We first observe that sink states satisfy the Nash equilibrium condition. Moreover, the non-cooperative algorithm does not enter an infinite loop. This is because using the discount factor to compute the best response (Eq. 6) results in selecting a state outside the loop with a higher payoff. Similar to the cooperative model, since no loop is iterated infinitely and each state has a path to a sink state, the non-cooperative model always reaches a state that satisfies the Nash equilibrium condition (possibly a sink state) and, thus, terminates.

4 EVALUATION

We evaluated the cooperative model (Section 3.3) and the non-cooperative model (Section 3.4) in terms of their performance and outcome. In particular, we are interested in answering the following research questions:

RQ1: How does the number of controllers affect the performance and outcome of the proposed models?

RQ2: How does the discount factor affect the performance and outcome of the proposed models?

RQ3: How does the granularity of the strategy space affect the performance and outcomes of the proposed models?

The number of controllers managing the shared object can significantly affect the size of the game graph. **RQ1** aims to investigate the effect of this complexity on the performance of our bargaining models. We introduced a discount factor to ensure the termination of the decision-making process (cf. Section 3.2). **RQ2** aims to study how different choices of the discount factor impact the performance and the collective decisions returned by the models. We also expect the performance and outcome of the multi-party access control game to be influenced by the granularity of the strategy space. In particular, providing controllers with fine-grained strategies to resolve conflicts can allow controllers to identify access decisions with a higher collective payoff but at the cost of performance. **RQ3** aims to explore this trade-off.

4.1 Implementation

We implemented a prototype of our multi-party access control framework, encompassing both the cooperative (Algorithm 1) and non-cooperative (Algorithm 2) models in Python. We also implemented a variant of the non-cooperative model in which the termination condition (line 7 of Algorithm 2) is relaxed by requiring the current state to be *either* a Nash equilibrium *or* a terminal state (see Sec. 3.4 for the rationale behind this variant). We refer to this variant as the *relaxed non-cooperative* model. We implemented a strategy space that includes the *permissive*, *restrictive*, and *conservative* strategies as described in Ex. 1; the permissive and restrictive strategies

can be tuned to account for the preferences of an arbitrary number of controllers. For instance, in a three-party game, our implementation allows for an instance of these strategies in which the preferences of all three controllers are considered as well as instances in which the controllers’ preferences are considered pairwise.

The proposed models require an exhaustive search to find the neighbor of the current state with maximum payoff (Eq. 4 and Eq. 6), which is time and memory-consuming. To this end, we applied several optimizations in our implementation. For the cooperative model, we employed a depth-first search to determine the neighbor of the current state with the highest payoff. The computation starts considering one controller’s preference, and the group payoff is iteratively updated by considering one additional controller at a time. Once the preferences of all controllers are considered, the state with maximum payoff (\mathbf{P}^* in Eq. 4) is updated, and our prototype backtracks to visit the next neighbor state, thus reducing the recomputation of the terms in the payoff function that were already computed. In the non-cooperative model, we shifted our search space to controllers’ possible preference set instead of searching through all neighbor states. This is because the same preferences of a controller might appear in several neighbor states. Our prototype only generates the neighbor states comprising the preferences with the highest payoff for each controller, thus reducing the search space.

4.2 Experiment Design

We conducted three sets of experiments to answer our research questions. In this section, we present the approach employed for the generation of the data used in the experiment, the settings of the experiments, and the evaluation metrics.

Data Generation. We simulated a collaborative system comprising 30 objects and 500 users. We randomly generated the set of initial access preferences for each user using a Gaussian distribution. The size of each controller’s set of initial preferences ranges between 10 and 100 users with a mean of 50 and a standard deviation of 50. We also generated a sharing benefit (*sb*) and a peer influence (*pe*) coefficient using a Gaussian distribution. Each object is assigned a given number of controllers (see below for details); for each controller assigned to an object, we generate a sensitivity level (*sl*) for that object based on a Gaussian distribution. Note that we adopted the common statistical modeling practice of employing normal distributions to avoid strong assumptions. This approach is particularly suitable for the cooperative model, where the summation of payoffs (cf. Eq. 4) introduces variables in the form of “sums of coefficients”. These variables, based on the Central Limit Theorem, follow a normal distribution regardless of the underlying distribution. To account for the strength of the relationship (*rel*) between controllers, we generated a social graph involving the users in the system using a power-law degree distribution, as suggested in [30]. We then assigned a value to every relationship in the social graph, denoting its strength, based on an exponential distribution [24]. Access requests were generated by randomly selecting an object and a user within the system.

Settings. We performed three experiments to investigate the impact of the number of controllers, discount factor, and granularity of the strategy space on the performance and outcome of the proposed models. In each experiment, we varied one parameter while keeping the others fixed. A summary of the settings is given in Table 4.

Table 4: Experiment Settings. The parameter controlled in each experiment is reported in bold.

	Controllers	Discount Factor	Strategy Granularity
Experiment 1	[2, 20]	0.8	All
Experiment 2	10	[0.1, 0.99]	All
Experiment 3	6	0.8	{2...6}

Experiment 1: Impact of the number of controllers (RQ1). In this experiment, we fixed the discount factor to 0.8 and increased the number of controllers from 2 to 20 similar to previous studies (e.g., [8]). The strategy space comprises three strategies, conservative, permissive, and restrictive, where the last two account for the preferences of all controllers.

Experiment 2: Impact of the discount factor (RQ2). In this experiment, we fixed the number of controllers for an object to 10 and varied the discount factor from 0.1 to 0.99. We used the same settings of Experiment 1 for the strategy space.

Experiment 3: Impact of the granularity of the strategy space (RQ3). To study the impact of the granularity of the strategy space, we created different strategy spaces in which we varied the number of controllers whose preferences are considered by the permissive and restrictive strategies, thus allowing a controller to be influenced by only a subset of other controllers. In particular, we varied the number of controllers whose preferences are accounted for by the permissive and restrictive strategies from 2 (the minimum meaningful number) to the total number of controllers. We used the same settings of Experiment 1 for the discount factor. On the other hand, we reduced the number of controllers to 6 to keep the experiment manageable (see Section 5)

Evaluation Metrics. We assessed our bargaining models (*cooperative*, *non-cooperative*, and *relaxed non-cooperative*) in terms of their performance and outcome. To assess the performance, we measured the computation time of decision-making processes and the number of iterations required to reach the collective decision.

To evaluate the outcome of group decision-making, we measure the ability of the models to determine the controllers’ preferences that maximize the group payoff. To this end, for each game, we compute the ratio between the group payoff of the state used to make the decision (\mathbf{P}^f) and the one of the initial state (\mathbf{P}^0) as follows:

$$\frac{\sum_{i=1}^n u_i(\mathbf{P}^f)}{\sum_{i=1}^n u_i(\mathbf{P}^0)}$$

Note that this metric is based on the equation for computing the payoff in the collaborative model (Eq. 4) without the discount factor. This is because we are interested in measuring the outcome of the models rather than the process to reach that outcome, providing a fairer comparison between the models. We also evaluated to what extent the bargaining process impacts the collective decision. To this end, we compared the access decision obtained using our models with the decision that could be enforced considering only the initial preferences of the controllers. To compute the access decision on the initial state, we considered three baseline approaches largely used in the literature: *permit-overrides*, which returns *permit* if at least one controller grants access to the requester, *deny-overrides*, which returns *deny* if at least one controller denies the access, and *majority voting*, which returns the decision of the majority of controllers (ties are resolved in a *deny* decision).

4.3 Results

Fig. 2 shows the results of the experiments. For each parameter configuration, we performed 1000 experiment runs, each run with a newly randomly generated initial set of access preferences and request (cf. Sec. 4.2). In the plots, we reported the average of the measurements along with the 95% confidence interval through shaded areas.

Experiment 1: Impact of the number of controllers. Fig 2a shows that increasing the number of controllers increases the execution time for all models. However, the increase is more prominent for the cooperative model than for the other two models. The cooperative model is faster for fewer controllers (up to 5). For larger numbers, its execution time experiences exponential growth (which is why this experiment was stopped after 14 controllers for the cooperative model). The execution times of the two other game models are similar, with the relaxed model being faster due to its relaxed termination condition. However, increasing the number of controllers causes the two approaches to converge. The increase in execution time for the cooperative model is due to an increase in the number of iterations needed to reach a terminal state (Fig. 2b) as well as to an increase in the execution time per iteration. On the other hand, the number of controllers has only a limited effect on the number of iterations to reach termination for the non-cooperative model and its variant. Fig. 2c shows the ratio of the final payoff to the initial one. We can observe that the ratio decreases with the increase in the number of controllers for all models. For a low number of controllers, the non-cooperative model provides a ratio above 1 (even higher than the cooperative model) but drops rapidly before stabilizing around five controllers. Conversely, the cooperative model experiences a slighter drop and maintains its payoff ratio above 1.

A comparison of the outcome of the three models with the three baseline approaches for conflict resolution (Fig. 2d) reveals that the cooperative model tends to behave as the *permit-overrides* baseline, while the non-cooperative models show similarity with *deny-overrides* and align more closely with the majority voting baseline. In particular, the cooperative model is more likely to resolve policy conflicts into a *permit* decision due to the maximization function used for the collective payoff (cf. Sec. 5). This becomes more evident when the number of controllers increases, as the initial state is more likely to be in conflict. The non-cooperative and relaxed non-cooperative models show the opposite behavior. Note that these two models always return the same decisions.

Experiment 2: Impact of the discount factor. Our results show that the cooperative model is less sensitive to changes in the discount factor, particularly in terms of execution time (Fig. 2e) and payoff ratio (Fig. 2g). Conversely, the non-cooperative and relaxed non-cooperative models undergo an increase in execution time and number of iterations when increasing the discount factor (see Fig. 2e and Fig. 2f). The payoff ratio for these models experiences a moderate increase, consistently staying below the payoff ratio observed for the cooperative model. It is worth noting that, particularly for higher discount factor values (i.e., 0.99), all three models show a more noticeable increase in execution time and number of iterations.

In line with prior findings, we observe that the outcome of the cooperative model does not depend on the discount factor (Fig. 2h). The other two models behave similarly to the cooperative model for

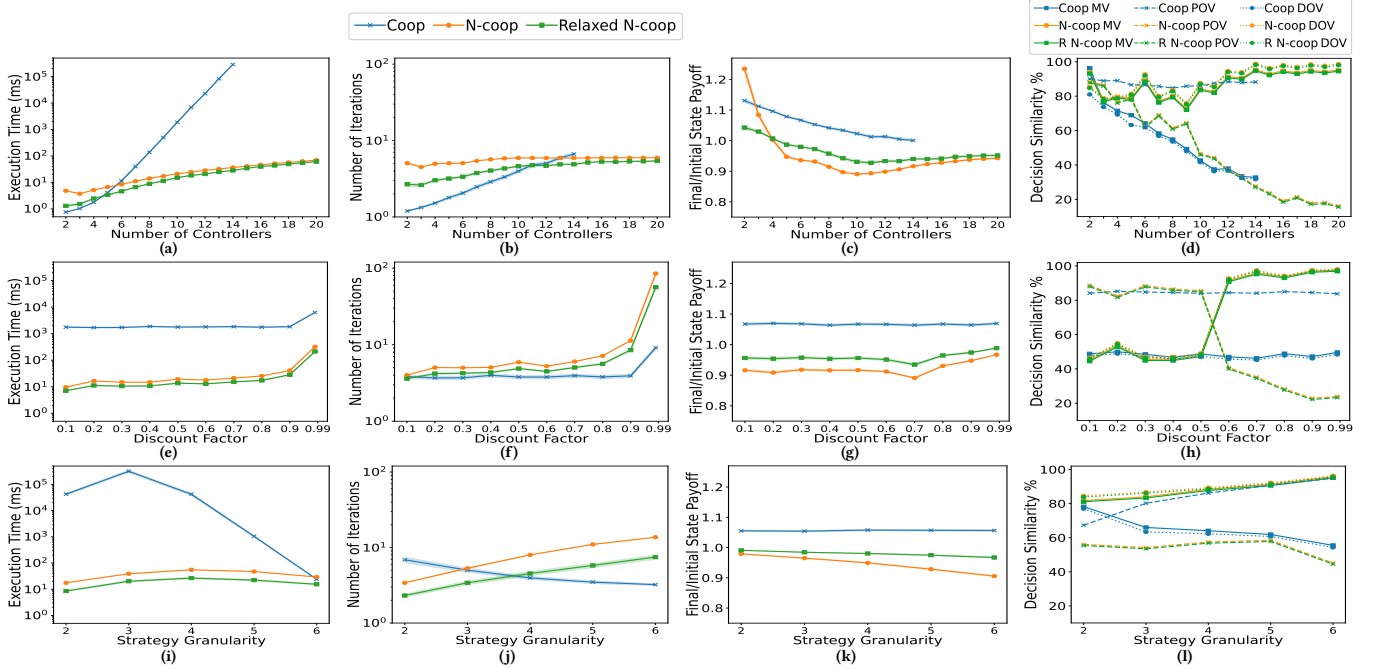


Figure 2: Results for Experiment 1 (2a, 2b, 2c, 2d), Experiment 2 (2e, 2f, 2g, 2h), and Experiment 3 (2i, 2j, 2k, 2l)

low discount factors, but they tend to return a *deny* decision when the discount factor increases. For low values of the discount factor, the non-cooperative model mostly finds *terminal* equilibrium states; for high discount factors, equilibrium states are mostly non-terminal and, thus majority voting is employed to make a decision, often resulting in a *deny* decision.

Experiment 3: Impact of the granularity of the strategy space. Our results show that the choice of the strategy space has a remarkable impact on the performance of the cooperative model. (Fig. 2i). In particular, the execution time is higher when the strategy space is larger. This can be explained by noting that the size of the strategy space significantly impacts the state space. This intuition was confirmed by analyzing the execution time per iteration. On the other hand, the non-cooperative and relaxed non-cooperative models are less subject to variations. By increasing the scope of the strategy, the cooperative model reaches a terminal state with fewer iterations (Fig. 2j) while the other two models show the opposite behavior. Finally, we can observe in Fig. 2k that the non-cooperative model achieves a higher payoff ratio when employing more fine-grained strategies.

When comparing the outcomes of the three models with the baseline approaches (Fig. 2l), we observe that the cooperative model tends to align with majority voting when the strategies involve the preferences of fewer controllers, and to return a *permit* decision when the strategies account for the preferences of more controllers. The other two models show an opposite behavior. For small strategy scope, the non-cooperative model is more likely to reach a *non-terminal* equilibrium state. In these states, controllers generally have preferences closer to their initial preferences (cf. Sec. 5), often resulting in a *deny* decision.

5 DISCUSSION

Effects of cooperation assumptions on performance and decision. As discussed in Section 3, bargaining interactions for multi-party access control can be analyzed through either a cooperative or a non-cooperative perspective. To this end, we proposed and evaluated two models, each capturing a different perspective. The results show that the cooperative model makes a decision based on a state with a higher collective payoff compared to the non-cooperative model (Figs. 2c, 2g, 2k). On the other hand, it is more computationally demanding, even if the number of iterations needed to reach consensus is generally lower. In particular, the cooperative model becomes impractical in settings with a large number of controllers (Fig. 2a) or a large strategy space (Fig. 2i). One interesting observation is the positive effect of employing more fine-grained strategies on the non-cooperative model (Fig. 2j, 2k). Although using these strategies leads to a larger state space, it better captures the possible dynamics between controllers, leading to equilibrium in fewer iterations (Fig. 2j) and with a higher payoff ratio (Fig. 2k). Furthermore, although the three models can reach similar collective decisions at certain settings, the non-cooperative model and its variant are usually more inclined toward majority vote. The high similarity with the majority vote makes the decisions of these models explainable, but it is not necessarily preferable in terms of performance (e.g., collective gain).

Characterization and parametrization of the payoff function. Our framework is equipped with a payoff function (Eq. 3) consisting of several factors influencing collaborative access decision-making. These factors have a potentially different impact on the decision outcome. For instance, the first term (*object sensitivity*) pushes towards preserving the initial preferences. The second (*relationship between controllers*) and fourth (*peer influence*) terms push toward reaching a consensus by increasing the similarity between their preferences.

The third term (*shared benefits*) increases with higher cardinality preferences, pushing toward a *permit* decision. After an in-depth analysis, we observed that the cooperative and non-cooperative models account for these terms differently. The cooperative model reinforces the terms favoring a *permit* decision (by including more users in the controllers’ preferences) as it aims to maximize the cumulative payoff. In contrast, the non-cooperative model mainly relies on the first term to reach equilibrium. The strength of the payoff terms is also influenced by the factor parametrization, which may also influence the behavior of the models. In this study, we used the same factor parametrization for all models to allow a fair comparison. The parametrization should be defined on a per-user basis. Some factors may be inferred from user behavior, while others may be provided through user self-reported data. To limit the burden on the controllers in setting their parameters, our framework can leverage pre-defined *personas* [19] to represent the different user types engaging in collaborative access decision-making.

Towards the development of multi-party access control mechanisms. Mechanisms for multi-party access control should capture the complex dynamics underlying decision-making in collaborative systems and satisfy the principles of fairness and optimality. Moreover, they should not rely on user intervention. To address these challenges, our framework leverages four factors influencing users’ decision-making to compute acceptable collaborative decisions that maximize controllers’ payoffs. In this work, we considered a multi-ownership governance model. Our framework achieves fairness by considering the payoff of all controllers equally. Although our payoff function (Eq. 4) can be adapted to handle asymmetric governance models, for instance, by using a weighted summation of controller payoffs, more research is needed to capture the dynamics underlying those governance models. We proposed two bargaining models and showed that they terminate without requiring user intervention as needed in other works [14, 27, 31]. Thus, our framework has the potential to operate in real-world collaborative environments where controllers might not be available all the time to resolve conflicts. We realized and evaluated a prototype of our framework to gain practical insights. Despite some optimizations (cf. Section 4.1), our framework, and especially the collaborative model, might not scale when objects are associated with a large number of controllers or the strategy space is ample, due to an exhaustive state space exploration. Further optimizations are needed to unlock the full potential of our framework. For instance, accounting only for “relevant” subsets of controllers while creating fine-grained strategies can reduce the search space and, thus, decrease the execution time.

6 CONCLUSION

In this work, we proposed a game theoretical framework for multi-party access control, comprising two bargaining models, to investigate different strategic environments influencing collaborative decision-making. Our models guarantee termination without user intervention, posing the basis for developing automated multi-party access control mechanisms. Our evaluation shows that different cooperation assumptions significantly impact the performance and outcome of collaborative access decision-making. In future work, we plan to perform user studies to support factor parametrization

and define personas representing typical user behaviors in collaborative access decision-making. We also plan to investigate optimizations tailored to the efficient computation of the collective payoff.

REFERENCES

- [1] G. Akkuzu, B. Aziz, and M. Adda. 2019. *A Fuzzy Modeling Approach for Group Decision Making in Social Networks*. Springer, 74–85.
- [2] K. Chatterjee. 2021. *Non-cooperative Bargaining Theory*. Springer, 531–543.
- [3] S. Damen, J. D. Hartog, and N. Zannone. 2014. CollAC: Collaborative access control. In *CTS*. IEEE, 142–149.
- [4] I. E. Dror. 2020. Cognitive and Human Factors in Expert Decision Making: Six Fallacies and the Eight Sources of Bias. *Analytical Chemistry* (2020), 7998–8004.
- [5] R. L. Fogues, P. K. Murukannaiah, J. Such, and M. P. Singh. 2017. Sharing Policies in Multiuser Privacy Scenarios: Incorporating Context, Preferences, and Arguments in Decision Making. *TOCHI* (2017).
- [6] J. D. Hartog and N. Zannone. 2016. Collaborative Access Decisions: Why Has My Decision Not Been Enforced?. In *ICISS*. Springer, 109–130.
- [7] H. Hu, G. Ahn, and J. Jorgensen. 2011. Detecting and resolving privacy conflicts for collaborative data sharing in online social networks. In *ACSAC*. ACM.
- [8] H. Hu, G. Ahn, and J. Jorgensen. 2013. Multiparty Access Control for Online Social Networks: Model and Mechanisms. *TKDE* (2013), 1614–1627.
- [9] H. Lu, G. Ahn, Z. Zhao, and D. Yang. 2014. Game Theoretic Analysis of Multiparty Access Control in Online Social Networks. In *SACMAT*. ACM, 93–102.
- [10] P. Ilija, I. Polakis, E. Athanasopoulos, F. Maggi, and S. Ioannidis. 2015. Face/Off: Preventing Privacy Leakage From Photos in Social Networks. In *CCS*. ACM.
- [11] Ö. Kibris. 2021. *Cooperative Game Theory Approaches to Negotiation*. Springer, 151–166.
- [12] B. Laursen. 2018. Peer influence. In *Handbook of peer interactions, relationships, and groups*. The Guilford Press, 447–469.
- [13] R. Mahmudlu, J. D. Hartog, and N. Zannone. 2016. Data Governance and Transparency for Collaborative Systems. In *DBSec*. Springer, 199–216.
- [14] P. Mehregan and P. W.L. Fong. 2016. Policy Negotiation for Co-Owned Resources in Relationship-Based Access Control. In *SACMAT*. ACM, 125–136.
- [15] R. B. Myerson. 1981. Utilitarianism, Egalitarianism, and the Timing Effect in Social Choice Problems. *Econometrica* (1981).
- [16] F. Paci, A. Squicciarini, and N. Zannone. 2018. Survey on Access Control for Community-Centered Collaborative Systems. *CSUR* (2018).
- [17] I.J. Pérez, F.J. Cabrerizo, S. Alonso, Y.C. Dong, F. Chiclana, and E. Herrera-Viedma. 2018. On dynamic consensus processes in group decision making problems. *Information Sciences* (2018), 20–35.
- [18] S. Rajtmajer, A. Squicciarini, C. Griffin, S. Karumanchi, and A. Tyagi. 2016. Constrained Social-Energy Minimization for Multi-Party Sharing in Online Social Networks. In *AAMAS*. 680–688.
- [19] D. Shin, R. Lopes, W. Claycomb, and G. Ahn. 2009. A Framework for Enabling User-Controlled Persona in Online Social Networks. In *COMPSAC*. IEEE, 292–297.
- [20] A. Squicciarini, S. Rajtmajer, Y. Gao, J. Semonsen, A. Belmonte, and P. Agarwal. 2022. An Extended Ultimatum Game for Multi-Party Access Control in Social Networks. *ACM Trans. Web* (2022).
- [21] A. Squicciarini, M. Shehab, and F. Paci. 2009. Collective Privacy Management in Social Networks. In *WWW*. ACM, 521–530.
- [22] A. Squicciarini, M. Shehab, and J. Wede. 2010. Privacy policies for shared content in social network sites. *Vldb J.* (2010), 777–796.
- [23] A. Squicciarini, H. Xu, and X. L. Zhang. 2011. CoPE: Enabling collaborative privacy management in online social networks. *JASIST* (2011).
- [24] I. Srba and M. Bielikova. 2010. Tracing Strength of Relationships in Social Networks. In *Web Intelligence/IAT Workshop*. IEEE, 13–16.
- [25] J. Such and N. Criado. 2016. Resolving Multi-Party Privacy Conflicts in Social Media. *TKDE* (2016), 1851–1863.
- [26] J. Such and N. Criado. 2018. Multiparty Privacy in Social Media. *CACM* (2018).
- [27] J. Such and M. Rovatsos. 2016. Privacy Policy Negotiation in Social Media. *ACM Trans. Auton. Adapt. Syst.* (2016).
- [28] E. Tardos and V. Vazirani. 2007. Basic Solution Concepts and Computational Issues. In *Algorithmic Game Theory*. Cambridge University Press.
- [29] K. Thomas, C. Grier, and D. Nicol. 2010. unFriendly: Multi-party Privacy Risks in Social Networks. In *Privacy Enhancing Technologies*. Springer, 236–252.
- [30] C. Wilson, A. Sala, K. Puttaswamy, and B. Zhao. 2012. Beyond Social Graphs: User Interactions in Online Social Networks and Their Implications. *ACM Trans. Web* (2012).
- [31] Q. Xiao and K. Tan. 2012. Peer-aware collaborative access control in social networks. In *CollaborateCom*. IEEE.
- [32] G. Xu, A. Zhou, and N. Agarwal. 2014. Special Issue on Social Computing and its Applications. *Comput. J.* (2014), 1279–1280.
- [33] J. Zhang. 2017. Benefit Model of Knowledge Sharing Behavior. *Open Journal of Social Sciences* (2017), 161–169.