Distributed Rational Consensus

Amjed Shareef
Department of Computer Science and Engineering,
Indian Institute of Technology Madras, Chennai - 600036, India.
Email: amjedshareef@gmail.com

Abstract. The *consensus* is a very important problem in distributed computing, where among the n players, the honest players try to come to an agreement even in the presence of t malicious players. In game theoretic environment, *the group choice problem* is similar to the *rational consensus problem*, where every player p_i prefers come to consensus on his value v_i or to a value which is as close to it as possible. All the players need to come to an agreement on one value by amalgamating individual preferences to form a group or social choice. In rational consensus problem, there are no malicious players. We consider the rational consensus problem in the presence of few malicious players. The players are assumed to be rational rather than honest and there exist few malicious players among them. Every rational player primarily prefers to come to consensus on his value and secondarily, prefers to come to consensus on other player's value. In other words, if w_1, w_2 and w_3 are the payoffs obtained when p_i comes to consensus on his value, p_i comes to consensus on other's value and p_i does not come to consensus respectively, then $w_1 > w_2 > w_3$. We name it as *distributed rational consensus problem* DRC. The players can have two values, either 1 or 0, i.e binary consensus. The rational majority is defined as number of players, who wants to agree on one particular value, and they are more than half of the rational players. Similarly rational minority can be defined. We have considered EIG protocol, and characterized the rational behaviour, and shown that EIG protocol will not work in rational environment. We have proved that, there exists no protocol, which solves distributed consensus problem in fixed running time, where players have knowledge of other players values during the protocol. This proof is based on Maskin's monotonicity property. The good news is, if the players do not have knowledge about other players values, then it can be solved. This can be achieved by verifiable rational secret sharing, where players do not exchange their values directly, but as pieces of it.

Key words: Distributed computing, Byzantine agreement problem, mechanism design, monotonicity property

1 Introduction

First we explain the byzantine agreement problem, i.e., consensus, in distributed computing environment. Next, we briefly narrate the consensus problem in game theory, i.e., rational consensus. After that, we discuss the recent work, where the problems of distributed computing and cryptography (secret sharing, multiparty computation) were solved with game theory orientation (players were considered rational). Next we introduce the rational consensus problem in distributed computing environment. In other words, this is similar to the distributed consensus problem being considered in game theoretic environment.

1.1 The Byzantine Agreement Problem: Consensus

The Problem of Byzantine agreement was proposed by Lamport, Pease and Shostak[1] in 1980. It was formulated to solve the problem of Byzantine generals, in which, the generals, some of whom may be faulty, try to decide whether or not to carry out an attack. Some traitorous generals may lie about whether they will support a particular plan or what other generals told them. Formally, each player starts with an input value, from a finite set, V and decides on an output value from same set. The players have to attain the consensus, given the fact that some of the players may be faulty and may behave in a malicious manner. The conditions for consensus are specified as follows.

1. **Agreement:** No two non faulty players decide on different values.
2. **Validity:** If all non-faulty players start with the same initial value, $v \in V$, then v is the only possible decision value for non-faulty player.
3. **Termination:** The termination condition requires that all non-faulty players must eventually decide.

The faulty behaviour of malicious players is considered as Byzantine faults. There are a variety of byzantine faults like, the byzantine adversary may not follow the protocol. He may choose either to not send any values or to send different values to different players. The byzantine agreement problem can be consider as the game between faulty and non faulty players, where non faulty players are trying to come to consensus and faulty players are trying to stop them. Pease et. al. [1, 2] proved that, agreement is possible only when $n \geq 3t+1$, where n is the number of players and t is number of malicious players. In other words, no solution to the Byzantine generals problem exists if one-third or more of the players are faulty.

1.2 Rational Consensus

Game theory provides a clean and effective tool to study and analyse the situations where decision-makers interact in a competitive manner. Game theoretic reasoning takes into account, which strategy is the best for a player with respect to every other player's strategy. Thus, the goal is to find a solution that is the best for all the players in the game. Every player's decision is based on the decision of every other player in the game and hence, it is possible to reach the equilibrium state corresponding to the global optima.

The group choice problem can be considered as rational consensus, where every player p_i prefers come to consensus on his value v_i or to a value which is as close to it as possible. All the players need to come to agreement on one value by amalgamating individual preferences to form a group or social choice. This problem does not consider the existence of malicious players and it is treated as group decision-making in game theory literature, and has been solved with many paradigms of preferences in many areas. Few papers concerned about the consensus are: group decision-making by Eliaza et.al.[3], group choice problem by Paine[4] and by Herrera et.al.[5]. Readers can refer paper by Kim[6] for a brief note on this problem.

1.3 Game Theory in Distributed Computing:

In distributed computing or secret sharing or multi-party computation, the players are mostly perceived as either honest or malicious players. The honest player follows the protocol perfectly where as the malicious player behaves in an arbitrary manner. Attributing rationality to players models the real life scenarios, where players (computers in distributed environment) need not be either honest or dishonest and are rather rational. It fits into the practical situations perfectly. In 2004, Halpern and Teague[7] introduced the problem of *secret sharing* (a primitive of cryptography) assuming that the players are rational, which is known as *rational secret sharing.* The impossibility of Rational secret sharing is proved by Halpern and Teague[7]. They show that rational secret sharing is not possible with any mechanism that has a fixed running time by iterated deletion of weakly dominated strategies (the strategy of not sending the share weakly dominates the strategy of sending the share). They also proposed a randomized protocol for $n \geq 3$. All these results apply to multi-party computation. Recent solutions to this problem are [8–11].

1.4 Rational Consensus in Distributed Computing

We consider the Consensus problem with n players, among whom at most t players can be malicious and at least $(n - t)$ players are rational. The rational players' behaviour is selfish. They have their own preferences and utility function (the profit they get). They always try to maximize their profits and behave accordingly. A rational player follows the protocol only if it increases his expected utility.

For any player p_i, let w_1, w_2, w_3 be the payoffs obtained in the following scenarios. $w_1 - p_i$ comes to consensus on his value, $w_2 - p_i$ comes to consensus on other player's value and $w_3 - p_i$ does not come to consensus. The preferences of p_i is specified by $w_1 > w_2 > w_3$. In brief, every rational player primarily prefers to come to consensus and secondarily, prefers to come to consensus on his2 value. We name this problem as *Distributed Rational Consensus* - DRC. The conditions for distributed rational consensus are specified as follows.

1. **Agreement:** No two rational players decide on different values.
2. **Validity:** If all rational players start with the same initial value, $v \in V$, then v is the only possible decision value for rational player.

3. **Termination:** The termination condition requires that all rational players must eventually decide.

Motivation

The motivation for the work is to analyse the consensus problem in game theoretic environment, which arises in many practical situations. An example for this situation is a distributed database in which data is distributed at various locations (computers). Suppose a decision has to be taken regarding the distribution of data and all the terminals are involved in this decision taking. Each terminal has its own estimate (depending on some criteria) on sharing the data and all the non-faulty terminals should come to an agreement on the load sharing(few terminals are faulty). Multiparty computation among rational and malicious players is solved by Lysyanskaya and Triandopoulos[12]. They proved that the functions can be computed when the number of malicious players are $\leq \lceil \frac{n}{2} \rceil - 1$(rest are rational players), in the presence of synchronous broadcast channel. The rational computation bounds in the absence of broadcast is left open. The Rational secret sharing problem is solved by Koal and noar [9] by assuming broadcast channel. The other papers which assumed broadcast channel are [7, 8]. We study this basic primitive which is essential for rational computation.

1.5 Model and Assumptions

We model the distributed rational consensus problem(DRC) as a game, denoted by Γ. Let the number of players be n and the byzantine faults be at most t. Let $\{p_1, \ldots, p_n\}$ be the n players participating in the game. Let $\{v_1, \ldots, v_n\}$ be the corresponding values on which the players prefer the consensus. We assume that all the players are connected to each other through secure private channels independently, which ensures that a player can send his data to a selected number of players. There is no broadcast medium available. In the synchronous model, the game is finite with respect to time and the starting and ending points are precisely defined. All the players send their data to other players and receive other players' data simultaneously at predefined synchronized points of time. It means, the messages will be delivered in fixed amount of time and all the players are synchronized with respect to a global clock. All players are assumed to be computationally bounded. There is no trusted mediator.

The game proceeds round by round. We describe DRC game by forest of n node. The root node of the forest denotes the actual configuration of the players situation. This nodes in next level(later stage) indicates results of the players moves at each step. The players receives the values from previous round sent by other players, and do manipulations(computations) on the set of received values and send new set of values. In other words we are using the game trees notation from the standard game theory literature. The agreement will be done after h rounds. At each node, each player has a local state that describe the set of values received, sent and computations performed. In short the history. A player p_i utility is u_i. Associated with each player, leaf node denotes the set of

utilities of all player; $(u_1, u_2 \dots u_n)$.
Utilities:
Every rational player primarily prefers to come to consensus on his value and secondarily, prefers to come to consensus on other player's value. In other words, if w_1, w_2 and w_3 are the payoffs obtained when p_i comes to consensus on his value, p_i comes to consensus on other's value and p_i does not come to consensus respectively, then $w_1 > w_2 > w_3$.

2 Basics of Game Theory

We define some basic terminology of game theory in this section [13].
A *strategy* can be defined as a complete algorithm for playing the game, implicitly listing all moves and counter-moves for every possible situation throughout the game. And a *strategy profile* is a set of strategies for each player which fully specifies all actions in a game. A strategy profile must include one and only one strategy for every player.

Let $\Gamma(N, L, U)$ represents an n persons game, where N is a finite set of n players $(p_1, \dots p_n)$, $L = \{L_1, \dots, L_n\}$ is a set of actions for each player p_i, $i \in \{1, \dots, n\}$ and $U = \{u_1, \dots, u_n\}$ is a utility function for each player, where $u_i : L{-}{>}\mathrm{R}$

Let a_{-i} be a strategy profile of all players except for the player p_i. When each player p_i, $i \in \{1, \dots, n\}$ chooses strategy $a_i \in L$ resulting in strategy profile $a = \{a_1, \dots, a_n\}$, then player p_i obtains payoff $u_i(a)$. Note that, the payoff depends on the strategy profile chosen, i.e., on the strategy chosen by player p_i as well as the strategies chosen by all the other players.

Definition 1. *Strict Domination: In a strategic game with ordinal preferences, player $p_i^{\prime}s$ action $a'' \in L_i$ strictly dominates his action $a' \in L_i$ if*
$u_i(a'', a_{-i}) > u_i(a', a_{-i})$ for every list L_{-i} of the other players' actions. We say that the action $a^{'}$ is strictly dominated.

Definition 2. *Week Domination): In a strategic game with ordinal preferences, player $p_i^{\prime}s$ action $a'' \in L_i$ weekly dominates his action $a' \in L_i$ if*
$u_i(a'', a_{-i}) \geq u_i(a', a_{-i})$ for every list L_{-i} of the other players' actions. We say that the action $a^{'}$ is weekly dominated.

2.1 Strategies

How do the players exhibit rational behaviour ?
As every rational player prefers to come to consensus on his value the most, he tries to push the agreement towards his value. Suppose a player p_j wants to agree on 0, he does all the things which will lead the agreement value to 0. For example, in every round of the protocol, the players send the previous round values to all other players. Player p_j while sending the set of values he received in the previous round to some arbitrary player p_i, sends all values as $(0, 0, 0, \dots, p_i\text{'}svalue, \dots, 0)$ as he wants to push the agreement on 0. A malicious

player behaves with a utility function to stop the consensus, while a rational player work with utility function to stop consensus on other player's value and come to consensus on his value.
Suppose player p_i wants to agree up on $'0'$, the player p_j wants agree up on $'1'$, then the player p_i forwards that p_j wants to agree on $'0'$ to all the remaining players. But if a player p_l wants to agree up on $'0'$, then honestly forwarding his value. The distinction between malicious and honest player is clear, malicious players works with utility to stop consensus, where as rational players works with utility that consensus should happen on their value. Building a mechanism which leads to consensus irrespective of rational behaviour in the presence of malicious players is a very complex task. The rational player always tries his best effort to consensus should happen on his value. The rational player will not exhibits his rational behaviour(or decrease rational behaviour) only if it leads to no consensus. A rational player p_i, in round r, honestly forwards k_{ij}^r values if by forwarding $k_{ij}^r - 1$, values leads to no consensus. If the value of (k_{ij}^r) for a player p_i increases, than his behaviour is shifting towards honest. If the value of k_{ij}^r decreases than his behaviour is shifting towards rational.

3 EIG protocol

First we formally present the EIG protocol mentioned in [14]. Next we show that this protocol will not work when the players are rational.

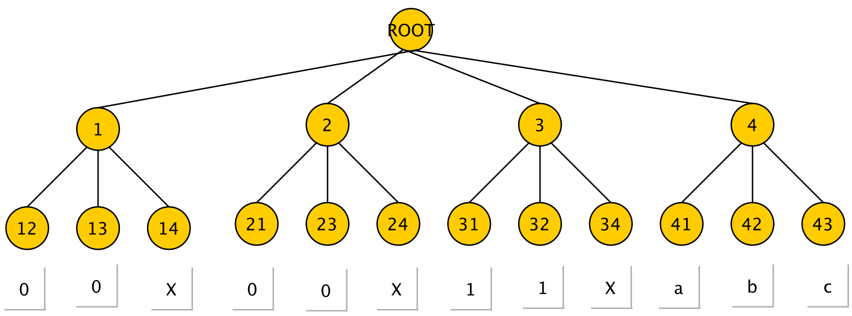

Fig. 1. Example illustrating EIG tree with $n = 4$

Let x denotes the value sent by malicious player. Let 4th player is malicious. In the above figure for the label 14, the value is x. Means, the honest player 1's values is reported by malicious player 4 as x. Malicious player 4 can report this value x as different to different players. Means 4^{th} players can tell to 2^{nd} player x value as 1 and to the 3^{rd} player as 0. So the x value could be different in each plays EIG tree. Where as the values a, b, c are the values reported by honest player. But the values are of malicious players'. The lable 41 value is a means, the malicious player 4's value reported by honest player 1. As player 1 is honest, he reports the same value to all the players. So the value a is same in all the honest players table. Similarly the values b, c. It leads to unique evolution of sub tree 4. For better understanding of EIG protocol please read [14], pages 108 to 111, and 120 to 121.

3.1 EIG protocol with rational players

Every rational player has a choice up on which he wants to agree. As we are considering binary value consensus, there are only two values v_1 and v_2. Among the rational players, let *maj* be the number of majority of players and *min* are the number of minority of players. For example, among 10 rational players, seven players want to agree on v_1 and three players wants to agree on v_2. In this case *maj* is 7 and *min* is 3. We define the majority player as, the player belongs to majority group, i.e who wants to agree on v_1 and minority player as, the player belongs to the minority group and wants to agree on v_2. Consider EIG tree with n = 5, majority is 3, minority is 1, t =1 for better understanding.

Lemma 1. *When the number of players are more than 4, and* $maj > (min + t)$*, in the EIG protocol, for Distributed Rational Consensus problem(DRC), the agreement always is on majority players values.*

Proof. For a minority player, the best rational behaviour could be masking all majority players values with his value. Suppose all the majority players wants to agree on the value v_1 and minority players prefer to agree on v_2. Let us assume a particular minority player p_j behaved rationally. Then while forwarding values, the minority player p_j, mask majority player, p_i's value with his value v_2. After $(t+1)^{th}$ round, all the players calculate the final value, at the $(t+1)^{th}$level sub tree, and then they roll back to t^{th} level and finally to the level 1. As all the players calculate majority in the sub tree and $maj > (min + t)$, the sub tree evolutes to the majority players value v_1. Similarly all the majority players values, at $(t+1)^{th}$level sub trees evolutes to v_1. And final agreement is on majority players value v_1. So in this case, for the minority player, being rational does not fetch any additional advantage. In fact the agreement is on majority players value, irrespective of the minority players behave honest or rational. So being honest results in output which is equivalent to rational behaviour. With the same reasoning, if a majority players behave honest, then agreement is on his value. Suppose they behave rational while forwarding minority players values, as $maj > (min + t)$ they can make minority plays sub tree evolutes to their value. While taking decision at first level, majority of values at second values should be considered, and the final agreement evolutes to their value v_1. Intuitively, as $maj > (min + t)$, for majority players, even by being honest they might not influence the minority players values(evolution of sub trees), but at level two due to the equality $maj > (min + t)$ agreement is on their values.

Lemma 2. *For a majority player, in state θ, where $maj\ = min\ + t$ rational behaviour strictly dominates honest behaviour. But in state θ', where $maj\ < min\ + t$ honest behaviour strictly dominating rational behaviour.*

Proof. Consider EIG tree with n = 4, majority is 2, minority is 1, t =1,($maj = min\ + t$) for better understanding. By behaving rational, majority players can influence the malicious players values, (otherwise it is decided by malicious players value or default value). Hence, by influencing they can force agreement on their value. Hence, for majority players behaving rational strictly dominates.

Now we consider a example, when $n = 7$ and, t =2, majority is 3 and minority is 2. If a minority player behaves rational it leads to no agreement. Let first 3 players, want to agree on 0 and 4th and 5th wants to agree on 1. 6th and 7th plays is malicious . If the 4th player behaves rationally and sends first players value as 1 to others, then while calculating the value at node 1 of the sub tree of EIG tree of first player, the majority value in the sub tree 1 is x, as if x is 1, the sub tree evolutes to 1 and if x is 0 the sub tree evolutes to 0. But one player can get x as 1 and other player can get x as 0 as the value x is sent by malicious player. Similarly the sub tree with node label 2 also evolutes with malicious players value. Hence there is no agreement. Above situation occurs when $maj\ < (min\ + t)$, as any manipulation of majority player value evolutes to inconsistency majority at the root of sub tree.

Lemma 3. *For a minority player, in state θ'', where $maj\ > min\ + t$ rational behaviour yields same outcome as honest behaviour. But in state θ', where $maj\ < min\ + t$ honest behaviour strictly dominating rational behaviour.*

Proof. In state θ', similar to majority player, same reasoning applies. In state θ'' reasoning applies from lemma 1.

Definition 3. *In a more general setting in which f can be set valued, monotonicity requires that, for all states θ, θ' and all outcomes a, if $a \in f(\theta)$ and $u_i(a, \theta) \geq u_i(b, \theta)$ implies $u_i(a, \theta') \geq u_i(b, \theta')$ for all i and bs, then $a \in f(\theta')$ [15, 16] .*

In other words, in [15] quoted as " Suppose that outcome a is optimal in state according to the social choice rule f in question, that is, $f(\theta) = a$. Then, if a does not fall in any ones ranking relative to any other alternative in going from state θ to state θ' , monotonicity requires that a also be optimal in state $\theta' : f(\theta') = a$. However, if a does fall relative to some outcome b in some ones ranking, monotonicity imposes no restriction."

Theorem 1. *(Maskin 1977): If a social choice rule is implementable, then it must be monotonic.[15, 16]*

Theorem 2. *EIG protocol(mechanism) do not solve DRC problem.*

Proof. For a majority player in state θ, rational behaviour dominates honest behaviour. But, in θ' state, rational behaviour fall in the ranking relative to honest behaviour. Hence, by monotonicity property we can say that EIG mechanism cannot be implementable. (Similarly, for minority player in state *theta*$''$, rational behaviour yields same outcome as honest. But in state θ', honest behaviour dominates rational behaviour.)

Theorem 3. *There exists no protocol, which solves distributed consensus problem in fixed running time, where players have knowledge of other players values during the protocol.*

Proof. Suppose there is a mechanism which solves DRC problem. Then based on validity property, if all rational players starts with same value, then agreement should be on the same value. So, the mechanism should incorporate $majority(V)$, and not $minority(V)$, where V be the set of values received during the protocol execution. Suppose there is a protocol which solves the rational consensus problem, following the agreement on $minority(V)$. Then, it breaks the validity property, hence a contradiction. Hence, any working protocol follows all the properties of EIG, so it follows all previous EIG lemmas. From Maskins monotonicity property we can say that mechanism for rational consensus problem cannot be implementable.

4 Good news: Possibility

The players exhibits rational behaviour when they receive other value (which is of not their interest). If they do not have knowledge of other players values, then they may have incentive behave honestly. The notion of secret sharing was introduced by Shamir [17]. The scheme was based upon the fact that it requires m unique points in order to define a polynomial of degree $(m-1)$. According to the scheme, a dealer generates a random polynomial, f, of degree $(m-1)$ such that $f(0) = s$, where s is the secret to be shared. Then he generates n points on the polynomial and distributes the couplet $(x_i, f(x_i))$ to every player i. If any m players come together, they will be able to regenerate the polynomial via Lagrange's interpolation and hence will be able to obtain the secret. So players first form shares of their value, and send each share to each player. Next, players mutually exchange their these shares of values to construct the message and finally agree. There can be some pitfalls as players may be first willing to learn other players share, before sharing their. But these things can be avoided by early stopping threat by other party. Significant work has been done to propose consensus protocol by using verifiable secret sharing as a black box.

5 Conclusion and Open problems

We have defined Distributed Rational Consensus problem. It has shown that the exponential Information gathering(EIG) algorithm do not work. By using Maskin's monotonicity property, we have proved that there is no protocol, which can solve, Distributed rational consensus problem, where players have the knowledge of other players values. The good news is, if players do not have knowledge about other players knowledge it may be possible, using secret sharing primitives.

We think we have left more open problems than what we have solved. Next thing to do is to propose a protocol when message is sent as parts, i.e using verifiable secret sharing. We expect all the bounds of verifiable secret sharing applies to Distributed rational consensus also. Lysyanskaya and Triandopoulos[12] proposed generalized theorem, bounds, about when it is possible to compute any multi party function in the rational environment in the presence of few malicious players. All the results are assuming the broadcast channel. In the absence of broadcast channel results are completely open. We believe, once we have bounds and protocols for Distributed rational consensus, we can propose generalized theorems, for multiparty computation, in rational and malicious players, without broadcast channel. Indeed, it is very interesting to see, what happen when there are few honest players, among rational and malicious players. Using cryptographic authentication and encryption primitives general bounds for DRC problem can be improved.

Multivalued distributed consensus problem seems to be very complex even to characterize the rational behaviour, and rigorous tools of mechanism design (like arrow's theorem) and game theory, are essential to solve it. Interestingly, we see that to get possibility and impossibility results, characterization of rational behaviour, we need core mechanism design knowledge, but to propose protocols we require good understanding of Cryptographic protocols. we can see that this problem lies at intersection of Game theory(mechanism design) and Cryptography areas.

References

1. Pease, M., Shostak, R., Lamport, L.: Reaching agreement in the presence of faults. J. ACM **27**(2) (1980) 228–234
2. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Trans. Program. Lang. Syst. **4**(3) (1982) 382–401
3. Eliaz, K., Ray, D., Razin, R.: Group decision-making in the shadow of disagreement. Volume 132, Issue 1. (2007) 236–273
4. Paine, N.R.: A useful approach to the group choice problem, Decision Sciences 4 (1), 2130 (1973)
5. Herrera, F., Herrera-Viedma, E., Verdegay, J.: A rational consensus model in group decision making using linguistic assessments. Volume 88. (1997) 31–49
6. Kim, H.K.: Rational consensus in science and society: A philosophical and mathematical study : Keith lehrer and carl wagner london: Reidel, 1981, 165 pages, 26. Mathematical Social Sciences **10**(1) (1985) 99–99
7. Halpern, J., Teague, V.: Rational secret sharing and multiparty computation: extended abstract. In: STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, New York, NY, USA, ACM (2004) 623–632
8. Abraham, I., Dolev, D., Gonen, R., Halpern, J.: Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In: PODC '06: Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing, New York, NY, USA, ACM (2006) 53–62

9. Kol, G., Naor, M.: Games for exchanging information. In: STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing, New York, NY, USA, ACM (2008) 423–432
10. Maleka, S., Shareef, A., Rangan, C.P.: The deterministic protocol for rational secret sharing. In: SSN : The 4th International Workshop on Security in Systems and Networks. (2008) 1–7
11. Kol, G., Naor, M.: Cryptography and game theory: Designing protocols for exchanging information. In: TCC. (2008) 320–339
12. Lysyanskaya, A., Triandopoulos, N.: Rationality and adversarial behavior in multi-party computation. In: CRYPTO. (2006) 180–197
13. Osborne, M.: *An Introduction to Game Theory.* Oxford University Press. (2004)
14. Lynch, N.A.: Distributed Algorithms. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1996)
15. Maskin, E.S.: Mechanism design: How to implement social goals (2007)
16. Serrano, R.: The theory of implementation of social choice rules (2003)
17. Shamir, A.: How to share a secret. Commun. ACM **22** (1979) 612–613