TrustBets: Betting over an IOU Network

(Extended Abstract)

Sharad Goel[1], Mohammad Mahdian[2], David M. Pennock[1], and Daniel M. Reeves[3]

[1]Yahoo! Research
[2]Google Inc.
[3]Beeminder

ABSTRACT

We consider the problem of operating a gambling market where players pay with IOUs instead of cash, and where in general not everyone trusts everyone else. Players declare their degree of trust in other players—for example, Alice trusts Bob for up to ten dollars, and Bob trusts Carol up to twenty dollars. The system determines what bets are acceptable according to the trust network. For example, Carol may be able to place a bet where she is at risk of losing ten dollars to Alice, even if Alice doesn't trust Carol directly, because the IOU can be routed through Bob. We show that if agents can bet on n events with binary outcomes, the problem of determining whether a collection of bets is acceptable is NP-hard. In the special case when the trust network is a tree, the problem can be solved in polynomial time using a maximum flow algorithm.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Intelligent agents, Multiagent systems*

General Terms

Economics

Keywords

Auction and mechanism design, electronic markets, economically motivated agents, peer to peer coordination, trust, reliability, and reputation

1. INTRODUCTION

A typical betting market is run by a central entity who is responsible for transferring payments from losers to winners. The market organizer collects cash deposits from the participants and carefully limits bets to ensure that all participants can cover their losses. Participants must tie up their cash in the system if they want to trade in the market.

We consider an alternate framework where no central entity collects deposits or verifies the creditworthiness of participants. Instead, participants declare their degree of trust in one another by stating the maximum amount of money they are willing to loan to specific individuals. For example, Alice may say she trusts Bob for up to ten dollars, while Bob says he trusts Carol for up to twenty dollars.

In our setting, loans have no strings attached, meaning there are no restrictions on what recipients can do with the borrowed money. In particular, it is allowed and indeed expected that loan recipients can in turn loan out the money to other people that *they* trust. Thus we envision participants declaring relatively conservative levels of trust so that, if worse comes to worst, they are fully prepared to absorb any and all losses stemming from defaulted loans. In the above example, Alice is in effect vouching for Bob, promising to cover up to ten dollars of Bob's debt if he defaults in the system. Trust in this sense is a directed binary relation with a real-valued weight. The set of all declarations of trust forms a weighted directed graph, known as the trust graph or *trust network* [9].

A *payment* in our system takes the form of an "I owe you" (IOU) from one participant to another, or more generally a sequence of IOUs among several participants. For example, a \$10 payment from Carol to Alice might be a direct IOU from Carol to Alice, or it might consist of two IOUs, one from Carol to Bob and one from Bob to Alice. A *feasible payment* is a payment that respects the trust network. More specifically, it is a payment that can be achieved via a series of binary IOUs following links backwards in the trust graph. In the running example, assuming no other IOUs have been issued, Carol's \$10 payment to Alice is feasible, since it can be achieved by issuing a \$10 IOU from Carol to Bob and a \$10 IOU from Bob to Alice, both within the limits that Alice and Bob declared.

Our previous work introduced and formalized the concept of a trust network as a distributed payment system and examined how to conduct a multi-unit auction when the buyers and seller are nodes in the network, showing the problem is NP-hard [9]. Subsequent work analyzed the liquidity of such trust networks [5], and their formation by strategic agents [6]. We note, moreover, that the trust network framework generalizes the case of budget-constrained agents, which has many practical applications and has been studied in both the economics and computer science literature [2, 3, 1, 7]. In addition to this theoretical work, there are at least two prototype implementations of trust networks [8, 10].

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

2. PROBLEM STATEMENT & RESULTS

We now formally define the problem of betting on a trust network, denominating bets in a hypothetical currency, *utils*, that represents an abstract measure of utility [10].

A trust network [9] is given by a weighted directed graph defined on a set of vertices $V = \{0, \ldots, m\}$ representing m agents. Edges in this graph represent trust relationships among agents. Each edge $(i, j) \in E$ with weight c_{ij} specifies that i has extended j a credit line of c_{ij} utils. The power of a trust network so defined is that arbitrary payments can be made by passing obligations between agents that explicitly trust each other if the network is sufficiently well-connected. A payment of x utils from agent u to agent v is feasible as long as there is a way to route x utils from u to v in the network, respecting the capacities given by the c_{ij}'s.

We consider the following simple betting scenario. Agents can place bets on n events with binary outcomes, each with fixed, even odds. That is, each event has two possible outcomes, denoted by -1 and 1, and if an agent places a bet of x utils on the outcome $b \in \{-1, 1\}$, it should be paid x utils if the outcome is b; and it should pay x if the outcome is $-b$. The final outcome of the n events is denoted by a vector in $\{-1, 1\}^n$, and the bets of an agent $i \in V$ is denoted by an n-tuple in $\mathbb{R}^n$, where a negative value $-x$ in the ℓ-th entry indicates a bet of value x for the outcome -1 in the ℓ-th event, and a positive value indicates a bet for the outcome 1. Therefore, if $\vec{x_i} \in \mathbb{R}^n$ denotes the bets of agent $i \in V$ and $\vec{v} \in \{-1, 1\}^n$ denotes the outcome, then the overall payment that agent i should receive (or pay) is given by the dot product $\vec{v} \cdot \vec{x_i}$. We assume that the bets are balanced, i.e., the sum of all bets equals the zero vector ($\sum_{i \in V} \vec{x_i} = \vec{0}$). This guarantees that under any outcome, the sum of payment vectors is zero, and therefore it is not necessary to inject any additional money into the network.

We study the problem faced by a mediator who is given a set of bets and needs to decide if these bets are feasible given the constraints that the underlying trust network imposes on the routing of payments among agents. We can define multiple versions of this problem, for example, deciding whether a given set of bets can be supported by the trust network, or selecting a maximal set of bets that can be supported. Here we focus on the decision version of the problem. (The decision problem in fact captures the complexity of the problem in the sense that variants of the problem for which the decision problem can be solved efficiently correspond to variants where the optimization problem can be solved efficiently.) Formally, the problem can be stated as follows:

> GAMBLING FEASIBILITY PROBLEM
> INPUT: Trust network $G = (V, E)$ with capacities c_{ij} on the edges, an integer n, and a bet $\vec{x_i} \in \mathbb{R}^n$ for each $i \in V$.
>
> QUESTION: Decide whether for *every* $\vec{v} \in \{-1, 1\}^n$, the payments $\vec{v} \cdot \vec{x_i}$ for every $i \in S$ can be routed through the trust network.

With this statement of the gambling problem, we find two results: (1) the problem is NP-hard in general, even for seemingly simple trust network structures; and (2) despite this hardness result, the problem is tractable if the network is a tree.

THEOREM 1. *The problem of determining feasibility of a gamble over a trust network is:*

1. *NP-hard in general, even if the trust network is a bidirected complete graph with uniform weights.*
2. *solvable in polynomial time if the corresponding undirected graph is a tree.*

Though we omit proofs of these results due to space constraints, we note the following useful reformulation of the problem. By the max-flow min-cut theorem [4], under any fixed outcome $\vec{v} \in \{-1, 1\}^n$, the problem of whether the payments under this outcome can be routed is equivalent to determining if for every set T of nodes in G, the total capacity of the edges from T to $\overline{T}$ is at least the total amount that the bettors in T win under the outcome $\vec{v}$. This amount can be written as $\max(0, \sum_{i \in T} \vec{v} \cdot \vec{x_i})$. Therefore, the gambling feasibility problem is equivalent to deciding whether for every set T of nodes, the capacity of the cut $(T, \overline{T})$ is at least the *maximum* amount that bettors in T can win under *any* outcome.

Acknowledgments

We are grateful for input from Arpita Ghosh and Yiling Chen.

3. REFERENCES

[1] C. Borgs, J. Chayes, N. Immorlica, M. Mahdian, and A. Saberi. Multi-unit auctions with budget-constrained bidders. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC)*, pages 44–51, 2005.
[2] Y.-K. Che and I. Gale. Standard auctions with financially-constrained buyers. *Review of Economic Studies*, 65:1–21, 1998. Reprinted in The Economic Theory of Auctions, edited by Paul Klemperer.
[3] Y.-K. Che and I. Gale. The optimal mechanism for selling to a budget-constrained buyer. *Journal of Economic Theory*, 92:198–233, 2000.
[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 1990.
[5] P. Dandekar, A. Goel, R. Govindan, and I. Post. Liquidity in credit networks: A little trust goes a long way. *ACM Conference on Electronic Commerce*, 2011.
[6] P. Dandekar, A. Goel, M. P. Wellman, and B. Wiedenbeck. Strategic formation of credit networks. *World Wide Web Conference*, 2012.
[7] S. Dobzinski, R. Lavi, and N. Nisan. Multi-unit auctions with budget limits. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2009.
[8] R. Fugger. The ripple project. `http://ripple.sourceforge.net`, 2004.
[9] A. Ghosh, M. Mahdian, D. M. Reeves, D. M. Pennock, and R. Fugger. Mechanism design on trust networks. In *Workshop on Internet and Network Economics (WINE)*, 2007.
[10] D. M. Reeves, B. M. Soule, and T. Kasturi. Yootopia! *SIGecom Exchanges*, 6:1–26, 2006.