Trust is Risk: Generalized Max Flow for Strategies between Idle and Conservative

Orfeas Stefanos Thyfronitis Litos

University of Edinburgh
o.thyfronitis@ed.ac.uk

Abstract. Previous versions of Trust is Risk present the Conservative and the Idle strategy as distinct and unrelated. This work is an attempt to generalize this idea into a continuous spectrum of strategies, the two ends of which correspond to the two previously defined strategies. Prior to querying the system for an indirect trust towards Bob, Alice can attribute a specific expected strategy to each one of the participating players, or even fine-tune the response of each player to a steal action from each one of the players she directly trusts. The system then executes the generalized MaxFlow [1] to determine the indirect trust from Alice to Bob, given the specified strategies of the rest of the players.

1 Introduction

In our previous work, we presented three distinct strategies for the players: the Idle, the Conservative and the Evil strategy. The process of determining the indirect trust from *Alice* to *Bob* involved assigning the Idle strategy to *Alice*, the Evil strategy to *Bob* and the Conservative strategy to all other players. The indirect trust from *Alice* to *Bob* would then be the worst case scenario for *Alice* when *Bob* initiates a "chain reaction" of steal actions. This value was proven to be equal to the maximum flow from *Alice* to *Bob*. Consider however the two following cases:

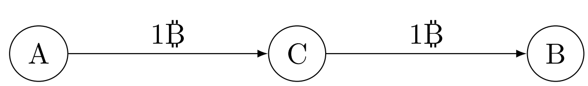

Fig. 1: $\mathcal{G}_1$, Few hops

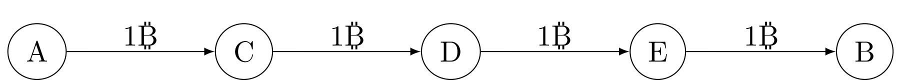

Fig. 2: $\mathcal{G}_2$, Many hops

One could argue that intuitively it should be $Tr_{\mathcal{G}_1,A\rightarrow B} > Tr_{\mathcal{G}_2,A\rightarrow B}$, since the longer chain of players connecting A and B in $\mathcal{G}_2$ introduces more uncertainty as to whether B is trustworthy. Nevertheless, according to our prior approach, it is $Tr_{\mathcal{G}_1, A \to B} = Tr_{\mathcal{G}_2, A \to B}$ since $maxFlow_{\mathcal{G}_1}(A, B) = maxFlow_{\mathcal{G}_2}(A, B)$.

To mitigate this problem, we introduce a generalization of the previous approach that can handle these cases more robustly. The mechanism that we propose is analogous to an attenuation factor that accounts for the number of hops, but is better suited for the ambience of maximum flow. Each edge $(v, w) \in \mathcal{E}$ of the graph is supplemented with an additional number called *loss factor*, which intuitively represents the "leakage" ratio on this edge, or the percentage of the damage incurred by w that is carried over to v through this edge, in case w is stolen some funds. The player who initiates the maxFlow calculation is the one that should specify the loss factors, according to her beliefs about the other players.

2 Strategy Generalization

More specifically, apart from the capacity c_e, each edge $e \in \mathcal{E}$ is assigned a gain factor $\gamma_e \in [0, 1]$.

[Definition of new conservative strategy]

The MaxFlow algorithm is substituted with the generalized MaxFlow algorithm [proof needed]. For the sake of example, consider the following two graphs:

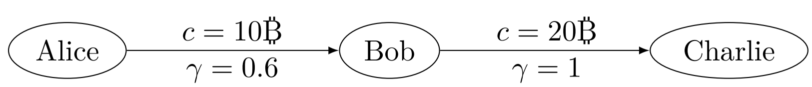

Fig. 3: $Tr_{Alice \to Charlie} = 6$฿

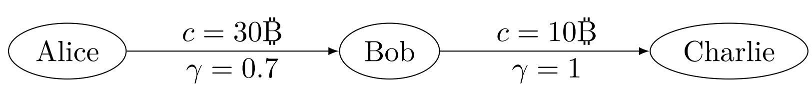

Fig. 4: $Tr_{Alice \to Charlie} = 7$฿

We will now attempt to obtain some insight as to how players behave on the two ends of the spectrum of the loss factor γ. Consider an edge $e = (Alice, Bob) \in \mathcal{E}$. If $\gamma_e = 0$, then *Bob* will not steal any funds from *Alice*, no matter how big a loss he has suffered. On the other hand, if $\gamma_e = 1$, then *Bob* can replenish any amount of stolen funds through stealing from *Alice*, given that she directly trusts him with sufficient funds.

If $\forall v \in \mathcal{V}, \gamma_{(v,Alice)} = 0$, then *Alice* is following the Idle strategy, whereas if $\forall v \in \mathcal{V}, \gamma_{(v,Alice)} = 1$, then *Alice* is following the Conservative strategy. [proof needed] The incoming direct trusts to the Evil player should all have a gain factor equal to 1, since we consider that she steals all her incoming direct trust.

References

1. Wayne K. D., Tardos E.: Generalized Maximum Flow Algorithms. Ph.D. thesis: Cornell University (1999)