

Breaking the Bank: Tycoon and Capital Integrity

Kristaps Džonsons, KTH/PDC

kristaps@kth.se

Abstract

The Tycoon bank is the central component of Tycoon¹, a resource management system for distributed computing. Tycoon abstracts compute resources into an economic model of supply and demand; the bank manages the capital with which this model operates. The bank is not appropriated within Tycoon’s machinery of market-driven economics: it may not be valued or devalued depending on its perceived integrity. The bank, in other words, must be trusted by all participating members or the entire mechanism is undermined. This in and by itself poses no conflict; however, the current design model of the bank lacks a facility for asserting account and capital integrity, allowing a bank under adversarial control to manipulate capital without detection. This is a serious issue: the bank, not being governed by the market, must have its own governance or the market’s equilibrium may be compromised by an adversary. In this document, we propose that the current bank has an intractable protocol for arbitrating an accusation of adversarial agency: an adversary, in the general case, can *always* disprove accusations. We follow this analysis with a system design that provably demonstrates adversarial conditions, and moreover, is able to identify the involved parties.

1 Introduction

The Tycoon system is a dynamic resource allocation and scheduling system for distributed environments. It uses a market strategy in order to incentivise users to request only necessary resources for their jobs, ensuring optimal usage of the total available resources.

¹See <http://tycoon.hpl.hp.com>

The BalticGrid Project² began experimenting with the Tycoon system in 2006, in order to provide grid-connected clusters with optimal resource usage. In 2008, due to the findings of this and other papers, the Tycoon system has been deprecated for use on the BalticGrid. The factors of this deprecation are enumerated in §4.

Although a detailed analysis of the Tycoon system is beyond the scope of this paper, we give a brief introduction for the sake of clarity. In a Tycoon system, users are allocated credits (capital), which may be used to purchase computation resources. This methodology, that of a market-based system, has endured a significant investment of study since the inception of multicomputer processing. The Tycoon system distinguishes itself by using virtual hosts for processing nodes and allowing preemption of jobs, among other features.

A Tycoon deployment is divided into the following components.

1. Hosts: provide resources in the form of virtualised hosts on a single physical host.
2. SLS: manages the global set of resources within the system.
3. Bidder: uses the SLS to find hosts with relevant resources then bid on these resources.
4. Auctioneer: auction off the resources of a host to bidders.
5. Bank: manages the flow of capital between auctioneers and bidders.

The bank, the focus of this document, has a straightforward function: it transfers accounts from a buyer (the “client”) to the seller (the “resource

²See <http://www.balticgrid.org>

broker”). During the servicing of a buyer’s request, capital is held in escrow by the bank. These steps have well-defined transitions and do not concern this proposal. In deference to simplicity, our analysis will involve generic “transfers”. A transfer into escrow is considered a transfer to the destination; a cancellation is a transfer back into the originating account.

Our proposal considers an adversarial agency that may at times transfer credits from a particular client to a broker without modifying the client’s account holdings, allowing a client to purchase resource beyond her means. This requires the adversarial collaboration of a bank and a client. Without oversight, these operations may go unnoticed by other account holders. In this document, we propose than an accusation of adversarial agency is mathematically unprovable.

Further, we consider a system that specifically addresses capital integrity. In it, we assume a clever adversary who may compromise any bank component at her caprices. We modify the semantics of capital transfer to involve a rigorous set of record-keeping, both from the client and the bank, to a set of auditing hosts. We base the perspective of this analysis from a single, trusted auditor. No other components are assumed as trusted. By making clever inferences from the coherence of records (an un-balanced “till”) as distributed across auditors, clients, and the bank, we create a system to prove adversarial agency.

Our modifications do not affect the market-economics of the Tycoon system, nor do they place outrageous demands on finite network and processing resources. Furthermore, our model of accountability acknowledges that clients, banks, and auditors themselves may be adversarial. The technical implementation of this system is beyond the scope of this document; this describes a *design theory*. In deference to brevity, we make casual reference to bank components that do not currently exist, but could without unreasonable work. We consider this reasonable as these extensions are not difficult to implement.

This design is *not* backward-compatible with the existing protocol. We make this decision in the interests of simplicity; this is addressed in §3.2 and

§3.1. We note, however, that it’s not unreasonable to construct a client-proxy that translates the old client protocol into the new without loss of integrity. This is not discussed in this document.

2 Breaking the Bank

We begin our analysis with a simple adversarial scenario. We assume a bank and several clients; records, as mentioned below, describe bank transfers, and we assume that the records themselves are ultimately authoritative, although their contents may be falsified. It goes without saying that an adversarial client implies an adversarial bank.

Alice operates a bank; Alice’s account holders are Bob, Fred, and Sally. Alice and Sally, unknown to Bob and Fred, are in an adversarial relationship allowing Sally to claim more resources than she has means. Bob bids ten credits for resources brokered by Fred. Although Sally has only 5 credits, Alice, in controlling the bank, allows her to bid and transfer 15 credits. Since neither Bob nor Fred know the bank’s total managed capital, this arrangement goes unnoticed by both parties. The bank’s gross capital, in this scenario, increases by 10.

Let’s assume that Bob notices Sally winning an inordinate number of bids; his notice may arise from the relatively few number of Sally’s serviced resources compared to the number of won bids. In order to support his claim, Bob collaborates with Fred. We assume that Bob and Fred are ultimately trustworthy. Bob and Fred bring their complaint to Alice, and ask her to produce Sally’s records. Since Alice is in collusion with Sally, she’s able to produce a fake record of transactions; thus, we must depend on other means of detection.

Formally speaking, if we define the capital of Bob at i to be c_i^b and Fred’s to be c_i^f , it’s clear that making any assumptions about Alice’s account, c_i^a , is intractable, as the initial capital investment of Alice’s account c_0^a is unknown; furthermore, if Alice is adversarial, she may produce a false value if asked.

Let’s consider a scenario where the bank’s initial gross capital, C_0 , is made public. Subsequent modifications of this capital (say, if an account is added

with a small initial sum), C_i , are also made public. We may assume that $C_i < C_{i+1}$. The value C refers to the current gross capital of the bank, and correspondingly, $C_0 < \dots < C_{n-1} < C$. For the time being, let's assume that values of C_i are trustworthy.

With this data, Bob and Fred calculate Sally's capital at i as follows:

$$c_i^s = C_i - (c_i^f + c_i^b)$$

In this, we see that Sally's capital equals the difference of the total capital and Bob and Fred's summed capital. Bob and Fred then reason that if the sum of Sally's alleged transfers exceeds the gross sum at any i less the sum of Bob and Fred's accounts, Sally must be lying.

Let us generalise this calculation. We define participating clients as $p \in P$, and the gross capital as the sum of all previous transactions $t \in T^p$, where T^p is an ordered set of transactions for $p \in P$. The generalised formula for C' , the calculated equivalent of C , follows:

$$C' = \sum_{i=0}^{\|P\|} \sum_{j=0}^{\|T^p\|} t_j^p \quad (1)$$

In this formula, we assume that T^p is produced for all clients and trustworthy. This is not likely the case, as the bank is not required to produce records. If the bank were to produce T^p , we must assume it false and requiring authentication. Since authentication requires us to corroborate the bank's T^p with every $p \in P$, we can reduce complexity by building a transaction record directly from clients. Since collaboration among clients is non-compulsory, whether to verify a T^p or to construct one, we're left with partial-sums from participating clients with which to reconstruct C' .

In subsequent analysis, we consider building C' by collaborating with other trusted clients. A "trusted" client is one whose data is assumed uninfluenced by an adversary. We take the liberty of assuming a pre-existing non-mathematic trust relationship between these clients, as building a trust relationship with the given data is demonstrable

intractable.

If we define a quorum $Q \subset P$, with $q \in Q$ of trusted clients, then we can calculate a probability of adversarial agency. Let's assume, to begin, that $Q = P - \{p^a\}$, or that Q is all participating clients but for the suspected adversary $p^a \in P$. We can construct an inferred transaction record T^{p^a} by calculating the losses and gains for p^a from Q . Since Q , in this regard, is all clients but p^a , this record is authoritative. We derive this from formula (1).

$$C' = \sum_{i=0}^{\|Q\|} \sum_{j=0}^{\|T^p\|} t_j^p + \sum_{j=0}^{\|T^{q^a}\|} t_j^{q^a} \quad (2)$$

Thus, if $C' \neq C$, then p^a is adversarial. This method assumes that Q is the difference of P and the suspected adversary p^a , which, unfortunately, is somewhat contrived for large values of $\|P\|$. If Q is significantly reduced quorum, then our analysis becomes probabilistic. First, we calculate C'' as being the gross capital for Q , where $C - C''$ is the gross capital for the remaining quorum, which possibly contains an adversarial agency. We can follow our logic above by first calculating the gain and loss capital between Q and $P - Q$. If this exceeds $C + C''$, then the quorum $Q' = P - Q$ contains an adversarial agency. This further derives formula (2).

$$C' = \sum_{i=0}^{\|Q\|} \sum_{j=0}^{\|T^q\|} t_j^q + F(Q')$$

Here, function F obscures transaction records that are unknown for the un-trusted quorum Q' . We can reduce the unknown quantity by iterating over all known $t \in T^q$ and producing a loss and gain record for the quorum Q' , which we label as $C^{Q'}$. This introduces uncertainty to formula (2).

$$C' = \sum_{i=0}^{\|Q\|} \sum_{j=0}^{\|T^q\|} t_j^q + (F(Q') - C^{Q'})$$

Since $F(Q')$ is unknown, we may detect an adversarial relationship only if C is less than the summed transaction record C^q for all $q \in Q$ and the inferred

capital $C^{q'}$ for all $q' \in Q'$. As $\|Q\|$ grows, the probability of detecting an inconsistency also grows.

The previous method only allows us to deduce an adversarial relationship within Q' . However, we may narrow the scope of this by carefully pruning entities from Q' whose loss and gain records in Q balance to 0. In other words, those entities $Q'' \subseteq Q'$ whose deduced $T^{q''} = 0$ when $q'' \in Q''$. Thus, the possible set of adversaries becomes $Q' - Q''$.

These methods fail if the adversary employs a *balanced* agency. In the previous example, we considered the adversary's influence to be detectable by examining the full historical context. Balanced agency amounts to a loan from the bank to an adversarial client. A loan destabilises the system for the duration of the loan-out. Although these aren't strictly-speaking violations of the system's long-term integrity, operation proportionate to the loan's magnitude and duration is effected. Loan violations may be detected by examining each interval over the course of the bank's operation. This detection follows the same formulae described earlier in this section.

Unfortunately, if an adversary influences the bank, then the published values for C must also be suspect. If the bank modifies values of C to agree with the adversary's falsified account record, then modifications are unprovable regardless the other account holders' suspicions. However, we may significantly improve this situation by making public the bank's initial capital at C_0 , which is not unreasonable. Our previous formulae must thus introduce a value of $C_0^{q'}$, or the possible gross capital offset generated by added accounts $q' \in Q'$ which involved a bank investment. This lessens the probability of adversarial detection in proportion to the $\|Q'\|$, which effects $C_0^{q'}$.

We thus determine that proving integrity is mathematically infeasible given the available data, although we may generate a probability. More importantly, we determine that data describing the bank's gross capital, if authoritative, is only helpful with the participation of a significant subset of clients.

In the following sections, we continue with the as-

sumption of an adversary who may control clients and the bank at will. We propose a system of accounting that operates at several points in the transfer system to ensure integrity, or, if integrity is mathematically infeasible, at least a significant empirical probability of integrity.

3 Solutions

The previous section has shown that an accurate C considerably increases the probability of detecting an adversary. If, however, an accurate T^p for $p \in P$ were available, then this probability advances. In this section, we consider a scenario of publicising all T^p . To the components of client and bank we add an *auditor*, and a corresponding *audit pool*. We define an auditor as a and an audit pool as A , where $a \in A$. Like any component, an auditor may be under adversarial control.

3.1 Detecting the Adversary

Consider a scenario where a client contacts the bank. At this time, a transaction record is generated with a sequence number s that has strict total ordering. Specifically, $s_0 < s_1 < \dots < s_n$, where there are no two $s_i < s_{i+1}$ where $s_i < s_j < s_{i+1}$. Once this sequence number has been established, both client and bank individually communicate this value to each $a \in A$. We assume, as we assumed in the previous section, that the *fact* of a communication is indisputable, while the *contents* of a communication may be disputed. One could implement this by cryptographic signatures. An auditor accepts a transfer record t_s from the bank. When a record t_s is generated, it is signed by both the client and the bank. We formalise this function as follows, where s is the totally-ordered sequence number, p is the signing client, b is the signing bank, and r is the record (including the value and identity of the receiving entity).

$$t_s \leftarrow f(s, p, b, r) \quad (3)$$

The signing function f provides an authoritative value. This is of considerable importance, as f

guarantees that the client and the bank have collaborated for the record and that the record was generated at s in the totally-ordered set. If an auditor a is off-line during the time of transmission, it may later query a bank for any t_s , which the bank must provide. We re-state that the message’s *fact* is undisputable, but the *content* may be falsified by adversarial collusion.

Each auditor maintains a set of record sets, or in other words, a set R , where each $r \in R$ contains the set of records T . Each $t \in T$ is a totally-ordered set. In practise, a record at time z generated by client x would correspond to client $r^x \in R$ and thus record t_z^x . We may calculate the bank’s gross capital on auditor a^x by iterating over these values:

$$C^{a^x} = \sum_{j=0}^{\|R^{a^x}\|} \sum_{k=0}^{\|T^r\|} t_k^r$$

This formula is the auditor-equivalent to (1). At this time, we address the notion of a conflict. A conflict occurs when an auditor determines an inconsistency in the records. An inconsistency is *always* the result of adversarial agency. An auditor a^x may detect inconsistencies in gross capital by comparing gross values at time C_{i-1} and C_i from $1 \rightarrow i$; if the gross capital changes without $\|R\|$ increasing with the new $r \in R$ of invested capital $C_i - C_{i-1}$, then a conflict exists.

When a conflict has been detected, it is escalated to resolution in the audit pool. When an audit pool convenes, we must validate the integrity of the pool. In this, we assume that auditor a^t , the convening auditor or our “perspective” of the system, is trustworthy.

$$\bigcup_{i=0}^{\|A\|} R^a \subseteq R^{a^t} \quad (4)$$

If any disparity arises between records, then a conflict exists. If equivalence is not met, one or more auditors is adversarial. We’re assured, from (3), that bank records are authoritative. We specifically define $A' \subset A$, where $a' \in A'$, as those auditors whose transaction set $R^{a'} \neq R^{a^t}$. Those a' are demonstratively adversarial.

3.2 Identifying the Adversary

With the situation of auditor equivalence from (4), or a reduced-equivalence quorum in $A - A'$, deducing adversarial parties is straightforward. In this situation, we unify all distributed values, as they are equivalent for all participating auditors. Earlier in this document, we determined that a disparity between stated and actual capital signifies an adversarial relationship; however, we were unable to demonstrate the fact, in practice, due to a lack of the full T^p . In this section, since a full record is available, we attempt to deduce the involved adversarial parties.

First, we determine the time $i > 0$ where the disparity occurs. It’s clear that at this point a transaction has occurred that imbalances the capital record. We define the set I as the initial value of each $p \in P$, then we see whether the used capital exceeds held capital. In this formula, i^p is the current capital for $p \in P$.

$$i^p = \sum_{j=0}^{\|T^p\|} t_j^p$$

If any $i_j < 0$, an adversarial situation exists for the corresponding $p_j \in P$. This calculation must be executed for each client. Obviously, $\sum_{i=0}^{\|I\|} = C_I - C$. With this formula, the subset of negative values $I' \subset I$ corresponds to adversarial clients and, by extension, the bank.

4 Conclusions

In this document, we’ve demonstrated that an adversary, in the current bank design, can *always* disprove her agency in the general case. We carefully defined the general case as having a trustworthy quorum $Q \subset P$, where $\|Q\| + 1 < \|P\|$. Since the Tycoon system is designed for very large grids, with thousands of inter-political clients, if not more, we don’t consider this an unreasonable criterion. Further, we demonstrate several simple measures that provide a probability of detecting adversarial agency, but ultimately may not be used in conclusively proof. We consider this a serious

issue that, in adversarial scenarios, may undermine the stability of the Tycoon market-economy equilibrium.

After identifying the issue, we proposed a scenario for auditing the Tycoon bank. We considered the complications of this system a small price to pay for integrity and accountability. Our proposed system allows for a pool of auditors to continuously verify the integrity of a Tycoon system during runtime. Auditors need not be always available; more importantly, our design acknowledges that an adversary may comprise auditors as well as clients and the bank. We do not consider the complexity of this system unreasonable; in fact, implementing this design is a fairly straightforward process. Modifying clients and the bank to support this design is also straightforward.

Since the publication of this document and subsequent documents regarding the Tycoon system on the BalticGrid³, further use of Tycoon has been discouraged. We note that the findings of this document have immediate solutions, those detailed herewith, which, if implemented, would have satisfied the issues raised. However, subsequent findings have revealed a super-set of issues, which are not so easily amended.

³See Džonsons, K., *Resource Allocation on the Baltic-Grid*, BalticGrid Project, 2008.