

1 The main concept

General note: this is a crude description of an idea, not a "final" design I'm convinced will work. IOW, this document should be enough for discussions and future plans, not something we can now go and implement.

1.1 Assumptions

This document proposes an implementation of the reputation model described in the "Microeconomic Golem Reputation"¹, and thus accepts all assumptions made there, most important being the expected utility maximization.

Additional assumptions:

1. The ground truth about all invoices on the market (amount, sender, receiver) is known.
2. Obtaining & parsing the data about invoices/payments costs neither money nor time.
3. The only information provider cares about is "how much I will be paid?" and the only information requestor cares about is the quality of the services received.
4. The market is balanced, i.e.:
 - (a) Good enough and sufficiently cheap provider will spend no more than X% of time waiting for agreements, for a reasonable X (e.g. 10%).
 - (b) Requestor willing to pay a sufficient amount of money will find enough providers fast enough

Some of them are questionable even as approximations - there is a further section discussing them.

1.2 General idea

The full knowledge about invoices and payments gives us a crude approximation of the "reputation-like" information:

1. Provider POV: the requestor who paid bigger part of their past invoices will more likely pay another invoice than another requestor who paid less past invoices.
2. Requestor POV: the provider whose invoices are usually paid more likely provides high quality services than a provider who is paid only rarely or never.

¹https://github.com/golemfactory/golem-architecture/blob/jb/microeconomy-reputation/reputation/microeconomic_reputation.pdf [TODO - beter url when this is merged to some final location]

Justification behind these statements:

- Both first and second: agent's strategy rarely changes, so it's likely that when trading with us they will behave similarly to the way they behaved in the similar past situations.
- Second: if a requestor doesn't pay the provider, they are less likely to trade with them in the future (e.g. because of a provider's local history), and thus requestors more often pay providers they want to trade with, and those are more likely the providers we want to trade with.

1.3 Detailed strategies

Provider strategy:

1. Gather a local history of all our contracts - (requestor_id, invoice_amount, paid_amount) + additional details (e.g. agreement length, initial demand etc).
2. Send offers to the market with "base" prices, that will later (in the negotiation phase) be increased for untrusted requestors.
3. When a requestor responds to initial our offer:
 - If this is a new requestor (i.e. one that didn't yet trade with anyone on the market, or didn't trade enough):
 - (a) Counteroffer with a "payment in advance" property
 - (b) If accepted, start with sending a debit note (for some fixed amount) with a short payment time
 - (c) Refuse to start an activity until the first debit note is paid
 - (d) Continue in the same manner, i.e. request upfront payments
 - If this is a requestor we've already traded with, and they didn't pay all the due amount:
 - (a) Add the missing amount (possibly with some discounting factor) to the fixed amount
 - (b) If agreed, start with sending a debit note for the missing amount, with short payment time
 - (c) Refuse to start an activity until the debit note is paid
 - In other cases, i.e. when requestor has some experience on the market and they either didn't trade with us or paid a full amount:
 - (a) Estimate the % of the final invoice we expect the requestor to pay us. In the most primitive version this might be just the

% of their invoices they paid until now, although this can (and should) be done much better²

- (b) If we estimate $\sim 100\%$ of the invoice will be paid, accept the offer.
- (c) If we estimate less will be paid, reply with a counter-offer with higher prices, so that they compensate for the % of the invoice we expect not to be paid.

Requestor strategy:

1. Gather similar local history as the provider
2. Sign agreements that require an advance payment only with providers you have a reason to trust (e.g. because you traded with them before or because they have a good history on the market)
3. Except for the previous point, just try to sign the most profitable agreements, including any information about the "provider quality" you have. This should include "high quality providers are paid more often than low quality providers", but also other things like "provider benchmarks" or the history of requestor's experiences.
4. When deciding about the amount to pay, consider following tradeoff. The less money I pay:
 - (gain) The more money is left on my account.
 - (loss) The more expensive deals I will have with other providers, especially if I decide not to pay a provider whose usually paid.
 - (loss) The more I will have to pay to trade with this provider again.

And pay any amount you consider best from your POV.³

1.4 The reputation deployment process

So we have defined some strategies for requestors/providers. Who will use them? How to encourage others to use similar strategies?

²E.g. if they didn't pay a provider that is never paid, there's a big chance this is an invalid provider who just does no work at all, but if they didn't pay a provider who's usually paid fully then probably something's wrong with the requestor. We should also include in the estimations few other types of the information, e.g.

- Our own experiences with this requestor (i.e. requestor who dutifully paid us a lot will probably pay again).
- Expected agreement length, proposed payment interval etc.

³This should somehow converge to "the worse is the provider, the less we want to pay" - and that's exactly what we want.

2 Discussion

2.1 Assumption 1 - ground truth about invoices

2.2 Assumption 2 - obtaining/parsing invoice data cost

2.3 Assumption 3 - agents' utility functions

2.4 Assumption 4 - market balance

2.5 Misc random notes

1. Maybe debit notes would be better than invoices?
2. If a provider starts with a debit note for the past debt first (as in the provider strategy described above), we need a way not to count the same debt multiple times