Oracle-Agent Problem

10/22/2018

1 Introduction

The UMA (Universal Markets Access) protocol is meant to provide universal access to financial markets so that everyone might have the opportunity to benefit from sharing risks with others. The way that we will achieve this is by building a platform for trading this risk and an *oracle* that will dictate what market prices are for various forms of risk... In order to support having a non-trivial amount of risk on the UMA protocol, we need to ensure that the oracle has the ability to produce accurate price information, and, in particular, it must be resistant to bribery attacks where an individual attempts to move prices in their favor through corrupt means.

This document describes a model which we use to influence decisions about how to ensure that the cost of corruption (CoC) is higher than the profit from corruption (PfC). The system is economically stable as long as the CoC > PfC. All things equal, it is best if the system generates a large spread between CoC and PfC to ensure that attacking the system requires as high a cost as reasonable, however, raising this spread will likely come at the cost of the efficiency of the system so it is important to think about the externalties imposed by the oracle on the trading protocol.

We begin by describing a static environment that allows us to think about the tradeoffs faced when attempting to raise the CoC-PfC spread and ensuring that the system is difficult to corrupt.

2 Static Environment

In the static world, the trading protocol generates $T = 2\tau N$ revenue and has mN margin that is exposed to being seized if the Oracle reports incorrect information[1].

There is a random state of the world $S \in \{0, 1\}$. There are three types of agents:

[1] For the assumptions that generate these outcomes, see Appendix A

1. The oracle: Responsible for reporting $\hat{S}$. The oracle wants to choose $\hat{S} = S$ but cannot observe S.
2. The malevolent: Can observe S and receives mN if the oracle reports $\hat{S} = 1 - S$.
3. The individuals: Can observe S and are incentivized by the oracle and the malevolent into voting $x = \{0, 1\}$.

There are K agents who decide whether to truthfully report the state to the Oracle... A truthful report entails $x = S$ and an untruthful report is $x = 1 - S$. Agents vary in their aversion to lying which is measured by $l \sim F$.

The Oracle's goal is to report the true state of world $\hat{S} = S$ in spite of the fact that it is, to the Oracle, unobservable. The Oracle does this by offering to pay the agents who vote $\gamma(x, X)$ where x is an individual agent's vote and X is the number of agents who vote $x = 0$[2]. After receiving the votes made by the agents, the Oracle reports whichever option received the majority vote. The payments to the voters are denominated in *rights to vote for tomorrow* and are also claims to a percentage of the tax revenue. The *rights to vote for tomorrow* are valued at $p' = \sum_{t=1}^{\infty} \left(\frac{1}{1+r}\right)^t \frac{T}{K}$ if the Oracle reports the truth and 0 otherwise.

There is also a malevolent agent who is allowed to make payments to the agents, $\tilde{\gamma}(x, X, S)$, but the malevolent can also condition their payments on S.

Agents report to maximize:

$$V(x, X, S, l) = \max_{x \in \{0,1\}} (p' + \frac{T}{K})\gamma(x, X) + \tilde{\gamma}(x, X, S) - \mathcal{I}_{x \neq S} l$$

Thus, given γ and $\tilde{\gamma}$, we can compute an implied distribution over X values. Let this distribution be called G.

The malevolent would like to corrupt the system[3] as cheaply as possible. The cost at which the malevolent can corrupt the system is given by

$$\begin{aligned} C^M(\gamma(x, X), S) = \min_{\tilde{\gamma}(x,X,S)} & \sum_X g(X) \sum_l f(l) \tilde{\gamma}(x_l, X, S) \\ & \text{subject to} \\ & 1 - G(K/2) > \xi_m \end{aligned}$$

The oracle would like to report the truth at the minimum cost. It's problem is given by

[2] We could have also chosen $x = 1$; as soon as we know how many people vote 0 (1) then we know how many people voted 1 (0).

[3] Here, corrupt means to raise the probability of corruption above a threshold ξ_m

$$
\begin{aligned}
V^O = \min_{\gamma(x,X)} & \sum_X g(X) \sum_l f(l) \gamma(x_l, X) \\
& \text{subject to} \\
& \sum_l \gamma(x_l, X) \leq K \quad \text{(Budget Constraint)} \\
& C^M(S, \gamma(x, X)) \geq mN \quad \text{Malevolent Incompatible}
\end{aligned}
$$

The Malevolent Incompatible constraint is what ensures that the oracle chooses a payment scheme that cannot be corrupted because it ensures that CoC > PfC.

3 Game Plan

The game plan is to solve for a discretized version of $\gamma(x, X)$ and determine the the corresponding $C^M(S, \gamma(x, X))$. With these in hand, we can likely pick a function that approximates $\gamma(x, X)$ "well enough" which mostly means that it doesn't require too much taxation on the trading system and that it satisfies the constraints of the oracle's problem.

Once we have a solution to the static problem, we will move onto the dynamic problem. This will allow for p' to be an endogenous object. The dynamic problem will also give the oracle more tools, such as positive reputation, to encourage truthful reporting. The malevolent agent does not have access to dynamic incentives because upon success the system shuts down. This should mean that the static results are still a solution to the truthful oracle problem (albeit an expensive one)...

4 Progress

4.1 Given an Oracle Policy

Imagine that we take as given an oracle policy. We will consider two potential policies:

1. No redistribution: $\gamma(x, X) = 1 \forall x, X$
2. Complete redistribtion: $\gamma(x, X) = \begin{cases} \frac{K}{X} & \text{if } X < \frac{K}{2} \text{ and } x = 0 \\ \frac{K}{K-X} & \text{if } X \geq \frac{K}{2} \text{ and } x = 1 \\ 0 & \text{else} \end{cases}$

With these two policies in hand, we can determine what the optimal corruption policy of the malevolent agent is. Using this corruption policy, we can do "back of the envelope" type calculations that allow us to back out the required tax rate to prevent corruption.

Using a discretized version of the model with the following parameters:

- $K = 5$
- $l \sim N(1, 0.5)$
- $T = 0.05$
- $r = 0.025$
- $\xi_m = 0.65$

The cost of attacking is

- No redistribution: 3.40
- Full redistribution: 3.55

Imagine that we associate the "profit from corruption" to $1,000

	Model	Example w $CoC = PfC + \varepsilon$	Example with $\tau = 5\%$
CoC	3.55	$1,000	$2,000
PfC	(?)	$1,000	$1,000
T	0.05	$14.05	$100
τ	(?)	0.7%	5%

4.2 Theorems

Theorem 1. $\gamma(x, Y) = \gamma(1 - x, K - Y)$
Theorem 2. $\tilde{\gamma}(1 - S, X, S) = 0$
Theorem 3. $\tilde{\gamma}(0, X, 1) = \tilde{\gamma}(1, K - X, 0)$
Theorem 4. *$\exists \bar{l}$ such that $\forall l > \bar{l}$ $x^*(l) = S$ and $\forall l < \bar{l}$ $x^*(l) = 1 - S$*

5 Appendix A

To generate the assumption of $T = 2\tau N$ and an amount of seizable margin of mN we assume:

1. There are N contracts held in the trading protocol.
2. These contracts are symmetric in the sense that each counterparty holds m margin.
3. The malevolent is a counterparty on each of the outstanding contracts and if the wrong state is reported, the malevolent can extract each of their counterparty's margin.
4. The counterparty of each contract is responsible for paying a tax, τ, in order to be provided access to the oracle's information.

Note that in some ways, this is a worst case analysis. The malevolent agent being a part of each contract is what drives the PfC to mN. In practice, the PfC would typically be lower than this.