ALBERT-LUDWIGS-UNIVERSITÄT
FREIBURG
INSTITUT FÜR INFORMATIK

Research Group on the
Foundations of Artificial Intelligence
Prof. Dr. Bernhard Nebel

Eliciting honest reputation feedback in a Markov setting

Studienarbeit

Jens Witkowski

Supervisor: Dr. Malte Helmert

August 2008

Acknowledgment

I want to thank my supervisor Dr. Malte Helmert for the excellent support and mentoring throughout the entire study project. He fostered my interest in the application of game theoretic models in computer science and was always open for discussions and new ideas.

Contents

Chapter 1

Introduction

Whereas fifteen years ago the term *electronic market* was primarily associated with online dependencies of large retailers, this has changed dramatically with the rise of Amazon and eBay. These companies probably understood best both chances and challenges entailed in online trading environments. As in other areas, the Internet helped market participants to drastically lower information costs. Whereas other retailers simply moved the content of their offline catalogues to an online environment, Amazon allowed customers to write feedback about the products they sell. Online auctions at eBay, on the other side, build an online market for products traditionally traded at a flea market and thereby enabled trade by distant market participants for these low- to mid-value products.

While a larger market is beneficial regarding both prices and supply, the distance and anonymity between traders give rise to challenges, as well. Reputation systems address two of these challenges: *moral hazard* and *adverse selection*.

Imagine an online auction setting where the buyer of a good is asked to pay for it in advance and only when the money arrived, the seller is supposed to send the good. Without any trust-enabling mechanism, it is not likely this trade takes place. Why should the seller send the good if he already received his money? And why should the buyer send the money knowing that the seller has no incentive acting according to the specified procedure? The seller is unable to credibly commit to sending the good. This constellation is termed *moral hazard*. The role of a moral hazard reputation mechanism is sanctioning bad behavior. eBay's reputation system is an example for such a mechanism and albeit problematic from a game-theoretic perspective, it only made possible the success of the market.[1]

Another related peril is *adverse selection*. In 1970, Akerlof [1] introduced his fa-

[1] According to Resnick and Zeckhauser [21] over 99% of all feedback given on eBay is positive. Widespread reports of consumer fraud in online auctions highly suggest that this fact does not reflect traders' actual experiences. Dellarocas and Wood [6] use statistical measures to estimate that only about 80% of ebay traders are satisfied with their respective trading partner.

mous *market of lemons*[2]. As an example, he sketched a market for used cars of different quality where the information about the cars' quality is asymmetric. That is, the seller knows each car's quality and its corresponding value while potential customers are only able to estimate the average quality and value of all cars together. A customer that is interested in a specific car (of her[3] choice) will not accept a price higher than the price corresponding to the average quality. Anticipating this, the seller withdraws all cars that have above-average quality as he would make a loss selling them at the average price. The potential buyer, again, anticipates the seller's action and adjusts the average price she is willing to pay for a car out of this reduced set of cars. Asymmetric information could thereby lead to a downward spiral of both quality and prices up to the point where there is only one car left, the one with the worst quality.[4]

The role of reputation mechanisms is to prevent these market failures. As mentioned earlier, there do exist reputation systems for both moral hazard and adverse selection scenarios. While the specific design of these systems differs[5], both types rely on feedback by agents that report their private experience with either another agent (in the case of a moral hazard setting) or a product[6] (in the case of an adverse selection scenario). The majority of reputation systems that are actually employed by internet companies simply neglect the incentive issues entailed in giving feedback. I present a mechanism that is able to cope with rational agents playing the system if that benefits them and although the focus of this work is on adverse selection settings, the principal requirements on the feedback payments are also applicable to environments with moral hazard.

The incentive issues when asking agents for feedback are coming down to two challenges in particular. The first is underprovision. Agents are usually required to register an account and are subsequently asked to fill out forms describing their experiences. This process is time consuming. In contrast to other settings such as those regulated by the legal system, we are unable to force agents into participating. Instead, we have to design the mechanism in such a way that it is in the agents' best interest to participate. In our setting, a rational agent will only invest the effort of giving feedback when remunerated appropriately.

Another—and trickier—issue is honesty. How should we incentivize agents to not only report *something* but to honestly report the quality they actually experience? In the context of giving feedback different scenarios with external interests (i. e. biases towards non-truthful reporting) are possible. Agents might fear retaliation or simply

[2]that—togehter with other related research to asymmetric information—earned him, Michael Spence and Joseph Stiglitz the Nobel Memorial Prize in Economic Sciences in 2001

[3]Following Miller, Resnick and Zeckhauser [15], I refer to the center as male and the rating agents as female.

[4]Please note that—in essence—this effect would also occur if either side's anticipation abilities are limited.

[5]see [5] for a comprehensive discussion of reputation systems in general

[6]I will use the terms *product* and *service* interchangeably.

feel uncomfortable giving bad ratings, resulting in too positive feedback. Another peril is sellers paying reporting agents for favorable feedback.[7] Imagine two companies competing for the same group of customers. Both companies may have incentives to badmouth the competitor or praise their own products and it is crucial to incorporate these issues into the mechanism's design by making it too costly for the seller to bribe.

The rest of the study thesis is organized as follows. In chapter 2 I introduce the basic setting and the needed probability calculations. Chapter 3 gives an intuition as to why the automated mechanism works and chapter 4 introduces the basic optimization problem. Chapter 5 introduces the Markov extension while chapter 6 discusses the experimental results. In chapter 7 I show how to efficiently compute the updates of the type beliefs and chapter 8 finally concludes with a discussion of future research.

[7] From a deceiptive seller's perspective, this may well be worth a thought: In a controlled field experiment, Resnick et al. [22] sold the same product (vintage postcards) on ebay with two different reputation profiles operated by the same seller. They found that the buyers' willingness-to-pay was 8.1% higher for the high reputation profile than for the low reputation profile.

Chapter 2

The Base Case

2.1 The Setting

The basic setting is that of Miller, Resnick and Zeckhauser [15] (henceforth MRZ) including adaptions by Jurca and Faltings [10, 12, 11]. A group of agents in an online market experiences the same product. The quality of the product (henceforth its *type*) stays the same[1] and is drawn out of a finite set of possible types $\Theta = \{\theta_1, \ldots, \theta_{|\Theta|}\}$. All agents share a common belief[2] regarding the prior probability $Pr(\theta)$ that the product is of type θ with

$$\sum_{\theta \in \Theta} Pr(\theta) = 1.$$

while $Pr(\theta) > 0$ for all $\theta \in \Theta$.

The quality observations by the agents are noisy, so that after purchasing the product, a buying agent does not know with certainty the product's actual type. Instead, she receives a signal drawn out of a set of signals $S = \{s_1, \ldots, s_M\}$.
Let O^i denote the signal received by agent i and let

$$f(s_m|\theta) = Pr(O^i = s_m|\theta)$$

be the probability that agent i receives the signal $s_m \in S$ given that the product is of type $\theta \in \Theta$. We assume that different types generate different conditional distribution of signals and that all $f(s_m|\theta)$ are common knowledge[3]. As these signal emissions again constitute a probability distribution, all signal probabilities sum up to 1:

[1] this assumption will be relaxed in section 5

[2] I will use the words *belief* and *estimate* interchangeably

[3] These rather strong assumptions can potentially be relaxed to incorporate noise *up to* a certain level. Jurca and Falting have presented a solution to the related problem of incorporating deviations in prior type beliefs[12].

$$\sum_{m=1}^{M} f(s_m|\,\theta) = 1 \qquad \forall\,\theta \in \Theta$$

We will allow the mechanism to pay agents for their feedback. A simple solution to the problem of underprovision is to pay the agents more than the rating process costs them. Let C^i be the (positive) costs reflecting agent i's time spent for this process. We ease the mechanism's requirement regarding these reporting costs by assuming there exists a C with $C = \max_i C^i$ that is not too far away from the individual costs of each agent.[4]

Let $\Delta^i(s_j, s_h)$ be the external benefit agent i could gain by falsely announcing signal s_h instead of signal s_j (the one actually received). Similar to the above-mentioned reporting costs, we relax the assumption of individual lying incentives and assume we know the upper bound $\Delta(s_j, s_h) = \max_i \Delta^i(s_j, s_h)$, denoting the maximal external benefit an agent may obtain by falsely announcing signal s_h instead of s_j. Furthermore, $\Delta(s_j, s_j) = 0$ and $\Delta(s_j, s_h) \geq 0$ for all $s_j \neq s_h$.

2.2 Comparing Signal Reports

In the rating process a central authority (the reputation mechanism) asks each agent for feedback regarding the quality information she perceived. This information is private to the agent and she can choose whether to report the signal actually received, to lie (i. e. to report some other signal $s \neq O^i$) or to not report at all. Let $a^i = (a^i_1, \ldots, a^i_M)$ be the reporting strategy of agent i, such that she reports signal $a^i_j \in S$ if she received s_j. The honest strategy is $\bar{a} = (s^i_1, \ldots, s^i_M)$, i. e. always reporting the signal received.

The payment assigned to the reporting agent is determined by comparing the signal announced by her with that of another agent $r(i)$, called the reference reporter. While different designs on how to choose the reference reporter can be considered, I will score the agents by simply evaluating two neighboring reports (in a time sense).[5]

The central idea of comparing two signal reports is that knowing one of the received signals should tell you something about the other. This is generally the case since both signals are emitted by the same underlying type and we demanded that signal emissions conditional on types are different for different types. This concept was coined *stochastic relevance*.

[4]neglecting the issue of private preferences.

[5]With regards to collusion properties of the mechanism, a random choice can be beneficial [11]. The drawback of a random choice is that the mechanism has to wait longer before publicly updating the type beliefs as it needs more private reports to randomly choose from. In addition, random choices may harm the mechanism's credibility if the random variable is not publicly verifiable. As an alternative one may consider rating an agent against an entire group of reporting agents. Jurca and Faltings [10] show that this approach can lead to a more efficient budget.

Definition 1. *Random variable O^i is* stochastically relevant *for random variable $O^{r(i)}$ if and only if the distribution of $O^{r(i)}$ conditional on O^i is different for different realizations of O^i.*

That is, O^i is *stochastically relevant* for $O^{r(i)}$ if and only if for any distinct realizations of O^i, call them s_j and s_h, there exists at least one realization of $O^{r(i)}$, call it s_k, such that $Pr(s_k|s_j) \neq Pr(s_k|s_h)$. I elucidate on the actual requirements on $Pr(\theta)$ and $f(s|\,\theta)$ in section 5.5. For the moment, I will assume stochastic relevance holds.

Let

$$g(s_k|\,s_j) = Pr(O^{r(i)} = s_k|\,O^i = s_j) \tag{2.1}$$

represent the posterior belief that $r(i)$ received signal s_k given that agent i received signal s_j.
2.1 can be extended to:

$$g(s_k|\,s_j) = \sum_{\theta \in \Theta} f(s_k|\,\theta)\;\cdot Pr(\theta|\,O^i = s_j). \tag{2.2}$$

Applying Bayes' Theorem to the second part of the sum in 2.2, we receive

$$Pr(\theta|\,O^i = s_j) = \frac{f(s_j|\,\theta) \cdot Pr(\theta)}{Pr(O^i = s_j)} \tag{2.3}$$

and for the denominator of 2.3:

$$Pr(O^i = s_j) = \sum_{\theta \in \Theta} f(s_j|\,\theta)\;\cdot\;Pr(\theta). \tag{2.4}$$

2.3 Example

Consider the simple example of only two possible types, a good type G and a bad type B. Furthermore, there are only two possible signals, namely a high signal h and a low signal l. The prior type probabilities are $Pr(G) = 0.7$ and $Pr(B) = 0.3$, the signal probabilites conditional on types are $f(h|G) = 0.75$ and $f(h|B) = 0.35$. With these given, we can calculate the prior probability of a rater receiving a certain signal. With a slight abuse of notation, I refer to $Pr(O^i = s_m)$ as $Pr(s_m)$.

$$
\begin{aligned}
Pr(h) &= f(h|\,G) \,\cdot\, Pr(G) \,+\, f(h|B) \,\cdot\, Pr(B) \\
&= 0.75 \cdot 0.7 + 0.35 \cdot 0.3 \\
&= 0.63
\end{aligned}
$$

$$
\begin{aligned}
Pr(l) &= 1 - Pr(h) \\
&= 0.37
\end{aligned}
$$

Bayes' Theorem gives us the probabilities for types conditional on signals:

$$
\begin{aligned}
Pr(G|h) &= \frac{f(h|G) \cdot Pr(G)}{Pr(h)} \\
&= \frac{0.75 \cdot 0.7}{0.63} \\
&\simeq 0.83
\end{aligned}
$$

$$
\begin{aligned}
Pr(B|h) &= 1 - Pr(G|h) \\
&\simeq 0.17
\end{aligned}
$$

$$
\begin{aligned}
Pr(G|l) &= \frac{f(l|G) \cdot Pr(G)}{Pr(l)} \\
&= \frac{0.25 \cdot 0.7}{0.37} \\
&\simeq 0.47
\end{aligned}
$$

$$
\begin{aligned}
Pr(B|l) &= 1 - Pr(G|l) \\
&\simeq 0.53
\end{aligned}
$$

Eventually, we can calculate the probability of a rater $r(i)$ receiving a certain signal conditional on the signal rater i received:

$$\begin{aligned} Pr(O^{r(i)} = h | O^i = l) &= g(h|l) \\ &= f(h|G) \cdot Pr(G|l) + f(h|B) \cdot Pr(B|l) \\ &\simeq 0.75 \cdot 0.47 + 0.35 \cdot 0.53 \\ &\simeq 0.54 \end{aligned}$$

$$\begin{aligned} Pr(O^{r(i)} = l | O^i = l) &= g(l|l) \\ &= 1 - g(h|l) \\ &\simeq 0.46 \end{aligned}$$

$$\begin{aligned} Pr(O^{r(i)} = h | O^i = h) &= g(h|h) \\ &= f(h|G) \cdot Pr(G|h) + f(h|B) \cdot Pr(B|h) \\ &\simeq 0.75 \cdot 0.83 + 0.35 \cdot 0.17 \\ &\simeq 0.68 \end{aligned}$$

$$\begin{aligned} Pr(O^{r(i)} = l | O^i = h) &= g(l|h) \\ &= 1 - g(h|h) \\ &\simeq 0.32 \end{aligned}$$

Please note, that rater $r(i)$ receiving a *high* signal is the most probable outcome whether rater i's signal was *high* or *low*. That is, simply paying the agents for agreement does not necessarily induce truthtelling.

Chapter 3

Peer-Prediction-Scoring

While I will later use Linear Programming (LP) in order to formulate the requirements mentioned in section 2.1, I introduce the mechanism's general concept by presenting the orignal model of MRZ [15] who use explicitly-stated scoring rules to elicit the agents' signals. This provides us with an intuitive understanding of the general concept behind the LP formulation.

3.1 Scoring Rules

Scoring rules [3, 26] are functions that can be used to incentivize rational agents to truthfully announce their private beliefs about a probability distribution[1]. For Ω, a set of mutually exclusive events, and P, a class of probability distributions over them, they take the form: $R : P \times \Omega \rightarrow \mathbb{R}$. A scoring rule is said to be *proper* if the agent is maximizing her expected score by truthfully announcing $p \in P$ and *strictly proper* if the truthful announcement is the only announcement maximizing her expected score. If not explicitly stated otherwise, I refer to *strictly proper* scoring rules.

The timely order is as follows: First, the agent is asked for her belief announcement $p \in P$. Second, an event $\omega \in \Omega$ materializes and, third, the agent gets paid $R(p, \omega)$, i. e. the score associated with the probability announcement and the event that actually took place. I reward agents with the score generated by a strictly proper scoring rule and assume that agents are aiming to maximize expected payoff (or expected utility with a linear utility function)[2].

Three commonly cited strictly proper scoring rules are the logarithmic, the spheri-

[1] Other usages of scoring rules exist. For instance, they are applied to rank probabilistic predictions, such as weather forecasts with one another. For an interesting information-theoretic interpretation of the log-scoring rule see [23].

[2] See [18] for an adaption of proper scoring rules that incorporate deviations from expected value maximization.

cal and the quadratic scoring rule. For its notational simplicity [3], I use the logarithmic rule to exemplify their usage:

$$R(p, \omega) = \log_b(p_\omega) \qquad | \quad b > 1,\, p_\omega > 0 \;\forall \omega \in \Omega.$$

Say, you want to incentivize a selfish agent to truthfully declare her probability estimate that presidential candidate A will win the general election of the United States in 2008. As there are only two possible events (i. e. *win* and *not win*), the distribution is well-defined by announcing the probability estimate of only one of them. Let p be the agent's probability estimate (or belief) that candidate A wins and let q be her announcement.

According to the scoring rule, the agent receives $\log_b(q)$ if the candidate wins and $\log_b(1-q)$ if he does not win. Her expected score is thus:

$$E_q = p \cdot \log_b(q) + (1-p) \cdot \log_b(1-q).$$

Setting the derivative to 0 gives $q = p$ as the only solution.[4] The second derivative is strictly negative at all positions of E_q, so the function is solely maximized by truthfully announcing $q = p$. In order to make it individually rational[5] for agents to participate, a constant may be added to the score without losing truthfulness.[6] Furthermore, the rule can be scaled with a scalar which allows creating the necessary incentives for the agent to invest costly effort into acquiring her actual belief.

3.2 The Peer-Prediction Method

As seen, *proper scoring rules* can be tailored to truthfully elicit private estimates about publicly observable events. In our setting, though, there is no such publicly observable event. Instead, the mechanism conditions the payments to agent i on the announcements made by her reporting agent $r(i)$. The key idea is to construct payments that make honest reporting by agent i the single best response to an honest report by $r(i)$ and vice versa.

From section 2.2 we know that O^i is stochastically relevant for $O^{r(i)}$. That is, different realizations of O^i generate different beliefs about $O^{r(i)}$. As in equation 2.1,

[3]The score of the logarithmic rule depends only on the probability assigned to the event that actually materialized (i. e. it is local) and it is the only proper scoring rule with this property.

[4]The derivative with respect to q is $\frac{dE}{dq} = p \cdot \frac{1}{q \cdot \ln(b)} + (1-p) \cdot \frac{1}{(1-q) \cdot \ln(b)} \cdot (-1)$

[5]See [20, p. 34f.] for a treatment of the different IR constraints.

[6]Note that this holds true for *interim* IR and the logarithmic scoring rule although the logarithm diverges to $-\infty$ around 0. The expected score is computed by multiplying the score of the events with their (believed) probabilities, so that $\log_b(\epsilon)$ is multiplied by ϵ. Adding the absolute value of E_p's minimum to the log-rule results in an expected payment that is positive for all $p \in P$. Equally, one may add a constant to push the expected score above the participation costs. In the case of the logarithmic rule, E_p's minimum is the uniform distribution of all p as this is the least-informative (as captured by the notion of entropy).

let $g(s_k|\, s_j)$ denote the posterior belief that $r(i)$ received s_k given that agent i received s_j. With a slight abuse of notation, let $R(a^{r(i)}|\, a^i)$ denote the result of the strictly proper scoring rule calculated by

$$R(a^{r(i)}|\, a^i) \equiv R(p, \omega) = R(\, g(s_k|\, s_j), a_k^{r(i)}). \tag{3.1}$$

As mentioned in section 3.1, scoring rules can be extended to incorporate both participation and effort constraints. The same method allows for the incorporation of external benefits. I will explicitly state them in the LP formulation in the next chapter. For the moment, I will assume that $R(\cdot|\cdot)$ is a strictly proper scoring rule scaled in a way that it complies with these constraints. Let $\tau(a^i, a^{r(i)})$ be the payment agent i receives if she announced a^i and the reference reporter announced $a^{r(i)}$.

Proposition 2. *If we assign the payments according to*

$$\tau(a^i, a^{r(i)}) = R(a^{r(i)}|\, a^i),$$

honest reporting by both agent i and $r(i)$ is a strict Nash equilibrium of the simultaneous reporting game.

Proof. Honest reporting is a Nash equilibrium if and only if honest reporting by agent i is the single best response to an honest report by $r(i)$ and vice versa. As our setting is symmetric, it is sufficient to show that given an honest report by $r(i)$, agent i is strictly maximizing her expected score by reporting honestly herself.[7] Given agent $r(i)$ is reporting honestly, i. e. $a^{r(i)} = s_k$, the best response by agent i is

$$\arg\max_{a^i} \sum_{s_k \in S} \underbrace{g(s_k|s_j)}_{Probability} \cdot \underbrace{R(s_k|\, a^i)}_{Score}.$$

Since $R(\cdot|\cdot)$ is calculated by a strictly proper scoring rule, the term is solely maximized by $a^i = s_j$. Thus, given an honest report by $r(i)$ agent i's best response is to announce her signal truthfully, as well. □

The expected payment (a priori) is

$$\sum_{j=1}^{M} Pr(s_j) \left(\sum_{k=1}^{M} g(s_k|\, s_j) \, \cdot \, \tau(s_j, s_k) \right) \tag{3.2}$$

Unfortunately, the honest equilibrium is not unique. MRZ [15] argue that the honest equilibrium will be chosen albeit a possibly pareto-optimal alternative as honest reporting is a focal point that makes it attractive for agents to coordinate on.[8] Jurca

[7]One may say this approach is decision-theoretic rather than game-theoretic as the mechanism is built around a single agent and extended towards a multi-agent setting only in the last step.

[8]For an illustration of focal or Schelling points, see for example [14, p. 248f.].

and Faltings [9] examine the application of reports that are true with high probability (so-called *trusted reports*). They find that rating other raters against these trusted reports makes honest reporting the only equilibrium strategy or renders unattractive lying equilibria.

3.3 Example

Applying the logarithmic scoring rule with base e to our example from section 2.3, results in the following payments:

$$\tau(h,h) = \ln(g(h|h)) \simeq \ln(0.68) \simeq -0.39$$

$$\tau(h,l) = \ln(g(l|h)) \simeq \ln(0.32) \simeq -1.14$$

$$\tau(l,h) = \ln(g(h|l)) \simeq \ln(0.54) \simeq -0.62$$

$$\tau(l,l) = \ln(g(l|l)) \simeq \ln(0.46) \simeq -0.78$$

We want all payments to be positive, so we add 1.14 to $\tau(\cdot,\cdot)$ and the resulting payment matrix is:

		$r(i)$: h	$r(i)$: l
i	h	$0.75, 0.75$	$0, 0.52$
	l	$0.52, 0$	$0.36, 0.36$

Note that in this example always reporting h by both agents and always reporting l by both agents are strict Nash equilibria. Furthermore, the former is pareto-optimal to the honest equilibrium whose expected payment (equation 3.2) is:

$$\begin{aligned} u(\bar{a}^i, \bar{a}^{r(i)}) &\simeq 0.63\,(0.68 \cdot 0.75 + 0.32 \cdot 0) + \\ & 0.37\,(0.54 \cdot 0.52 + 0.46 \cdot 0.36) \\ &\simeq 0.49 \end{aligned}$$

It is $0.75 \simeq u(h,h) > u(\bar{a}^i, \bar{a}^{r(i)}) \simeq 0.49$. For binary settings (i. e. only two possible signals), there are always lying equilibria with one of them being pareto-optimal to the honest equilibrium [9].

Chapter 4

An Automated Mechanism

An alternative to the use of explicit proper scoring rules is the formulation of the above-mentioned equilibrium requirements as a Linear Program. This technique was coined *Automated Mechanism Design* [2, 24] and is advantageous in our setting for several reasons. First, we will be able to find the budget-optimal, i. e. the cheapest, mechanism while preserving the requirements for truthtelling and voluntary participation. Second, we can focus on thinking about the requirements while delegating a large fraction of algorithmic considerations to an external solver. That not only limits the possibility of programming mistakes but also allows for an easier adaptation for more complex extensions to the model than what would have been feasible with a manual formulation using explicit scoring rules. Nonetheless, it is insightful analyzing the mechanism's behavior with the intuition of strictly proper scoring rules (compare chapter 3).

Definition 3. *Let $A \in \mathbb{R}^{m \times n}$ be a matrix, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$. A Linear Program in* standard form *has the following form:*

$$\begin{aligned} min \quad & c^T x \\ \\ subject\,to \quad & Ax \geq b \\ & x \geq 0 \end{aligned}$$

We are looking for the budget-optimal mechanism, i. e. the minimization of expected payments. We already formulated the expected payment in equation 3.2 and we may use it as objective function in our LP. All probability calculations are the same as in chapter 3. The only difference is that our $\tau(\cdot, \cdot)$ is no longer defined by an explicitly stated scoring rule. Instead, the LP solver will search for the optimal assignment. Please note that the objective function is the expected payment of the honest equilibrium since this is what we expect the agents to coordinate on (after all, this is the mechanism's entire purpose). For instructions how to calculate $g(s_k|\, s_j)$, see section 2.2.

The core of the LP consists of the constraints that make sure that the honest signal announcement is the single best response given an honest reporting agent. For every possible signal $O^i = s_j \in S$, there exist $M - 1$ dishonest announcements $a^i_j \neq \bar{a}_j$. Given that the reference report is honest, we want the expected payment of an honest announcement by agent i to be strictly larger than the expected payment of any other announcement. More accurately, we want them to be strictly larger by a margin $\Delta(s_j, s_h)$[1], that is,

$$\sum_{k=1}^{M} g(s_k|\, s_j) \cdot \tau(s_j, s_k) - \sum_{k=1}^{M} g(s_k|\, s_j) \cdot \tau(s_h, s_k) > \Delta(s_j, s_h)$$

$$\forall s_j,\, s_h \in S,\, s_j \neq s_h$$

and (in a more compact notation)

$$\sum_{k=1}^{M} g(s_k|\, s_j)\, (\tau(s_j, s_k) - \tau(s_h, s_k)) > \Delta(s_j, s_h).$$

$$\forall s_j,\, s_h \in S,\, s_j \neq s_h$$

In our setting, an agent decides whether to participate in the rating process after experiencing the good (i. e. she knows her own signal) but without knowing the signals received by the other agents. Therefore, we use *interim* IR:

$$\sum_{k=1}^{M} g(s_k|\, s_j) \cdot \tau(s_j, s_k) > C \quad \forall s_j \in S$$

In addition, all payments need to be positive as we have no possibility withdrawing credit from the agents.

The mechanism's final LP in standard form LP is:

LP 1.

$$min \quad W = \sum_{j=1}^{M} Pr(s_j) \left(\sum_{k=1}^{M} g(s_k|\, s_j)\, \cdot\, \tau(s_j, s_k) \right);$$

$$\begin{aligned} s.t. \quad & \sum_{k=1}^{M} g(s_k|\, s_j)\, (\tau(s_j, s_k) - \tau(s_h, s_k)) > \Delta(s_j, s_h); \\ & \qquad \forall s_j,\, s_h \in S,\, s_j \neq s_h \\ & \sum_{k=1}^{M} g(s_k|\, s_j) \cdot \tau(s_j, s_k) > C \quad \forall s_j \in S; \\ & \tau(s_j, s_k) \geq 0 \quad \forall s_j, s_k \in S; \end{aligned}$$

[1] Remember that $\Delta(s_j, s_h)$ are possible external benefits from lying (compare chapter 2.1).

Jurca and Faltings present numerous extensions to this base model. They show how to further lower the budget by using multiple rating reports and a filtering technique for reports that are false with high probability (while still paying these reports) [10]. In order to incorporate prior beliefs that are slightly different from the center's, they built a mechanism robust to small changes in these beliefs [12]. An insightful presentation of the expressive abilities entailed in the LP formulation is the work on colluding agents and sybil attacks (i. .e one agent controlling several accounts) [11].

Chapter 5

Time-dependent quality changes

So far we have only considered situations where the quality of the product is fixed. This situation is not given very often. Imagine a freemail provider that constantly improves its services as it is competing for market share (e. g. expanding the allowed storage space) or technical products such as television devices that get outdated because of technological change. Fortunately, these quality changes happen rather slowly. An agent rating a television device today can trust with high probability that the signal received by her reference agent arriving a week later stems from the same underlying type. These slow and constant changes can be taken care of by adding a small constant to the $\Delta(s_j, s_h)$ bounds in the LP.

This is different for more complex settings often occuring in online environments. Imagine a webservice that has a certain probability to be offline for some time but that returns to its old quality type after it got fixed. Or—in a similar vein—another webservice might have different loads depending on the number of clients it has to serve at the time. With the fixed-type mechanism we are unable to model these situations appropriately as we do not have a possibility for near-time reports being correlated stronger than those further away.

5.1 A Markov Extension to the Base Setting

In order to allow for these time-dependent quality changes, I extend the base setting by introducing a transition matrix modelling quality changes as a Markov Process (MP). Together with the noisy perception of the signals, the resulting structure is a discrete-time hidden Markov model (HMM) with a finite set of states.

Besides the fact that selfish agents make the observations, I also extend the standard definition of an HMM to allow for both *null transitions* and *multiple observations* (i.e. for each t, there might be no agent receiving a signal or more than one agent receiving a signal, respectively).

The transition matrix $\boldsymbol{P}$ is given with the problem definition and stays the same for all t (i. e. the MP is time-homogeneous). Given type (i. e. state) θ_i, the probability to go to type θ_j in the next time step is $Pr(\theta_j^{t+1} \,|\, \theta_i^t)$. Please note that $\boldsymbol{P}$ is a so-called *left* stochastic matrix (i. e. the *columns* sum up to 1). This is coherent with the format of the other matrices I use and it allows for a simple way to determine the probability of the type vector in a certain time step. Since we will need to rate at least one agent against a *succeeding* agent, $\boldsymbol{P}$ is required to be reversible. In addition, we demand it to be non-singular.

$$\boldsymbol{P} = \begin{pmatrix} Pr(\theta_1^{t+1} \,|\, \theta_1^t) & \dots & Pr(\theta_1^{t+1} \,|\, \theta_{|\Theta|}^t) \\ \vdots & \ddots & \vdots \\ Pr(\theta_{|\Theta|}^{t+1} \,|\, \theta_1^t) & \dots & Pr(\theta_{|\Theta|}^{t+1} \,|\, \theta_{|\Theta|}^t) \end{pmatrix}$$

We assume the agents announce their signals at the time they receive it. That is, their strategies do not depend on t. This is reasonable in cases where the reputation system is located at an intermediary, such as expedia.com or amazon.[1] Here, agents cannot lie about the time step in which they consumed the product as the reputation mechanism already knows this from the booking data. A possible weakness is that the good's consumption may be either postponed or brought forward if that is beneficial in the rating process. Yet, I believe that when booking a holiday at expedia it is reasonable to believe that the time of travel is independent from possible minor advantages in the subsequent rating process. However, these issues can be important in certain applications and constructing a dynamic mechanism remains future work.

5.2 The Optimal Time-Dependend Payment Scheme

Without loss of generality, let $r(i)$ receive her signal at time t_1 while agent i receives her signal at time t_2. Let $s_k^{t_1}$ and $s_j^{t_2}$ denote the signals received by $r(i)$ and agent i, respectively. Slightly abusing the notation, I alter $g(\cdot \,|\, \cdot)$'s defintion (see equation 2.1) to incorporate the timing information.

The expected payment to agent i is thus:

$$\sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \,\cdot\, \tau(a_j^i, a_k^{r(i)}).$$

After observing her signal $s_j^{t_2}$, agent i wants to maximize her expected payment. Thus, given that her reference rater $r(i)$ reports truthfully, her optimal choice is

$$\arg\max_{a_j^i} \sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \,\cdot\, \tau(a_j^i, a_k^{r(i)}).$$

[1]The same holds true for ebay in moral hazard environments

$g(s_k^{t_1} \,|\, s_j^{t_2})$ can be computed as:

$$g(s_k^{t_1} \,|\, s_j^{t_2}) = \sum_{l=1}^{|\Theta|} Pr(s_k^{t_1} \,|\, \theta_l^{t_1}) \cdot Pr(\theta_l^{t_1} \,|\, s_j^{t_2}). \tag{5.1}$$

$Pr(s_k^{t_1} \,|\, \theta_l^{t_1})$ can be simplified to $f(s_k \,|\, \theta_l)$ as the probability of receiving a signal given a certain type is independent of when it is received as long as it is in the same time step (for reasons of simplicity, I will sometimes omit the timing information for these cases in subsequent calculations).

Applying Bayes' Theorem to equation 5.1 we receive:

$$Pr(\theta_l^{t_1} \,|\, s_j^{t_2}) = \frac{Pr(s_j^{t_2} \,|\, \theta_l^{t_1}) \cdot Pr(\theta_l^{t_1})}{Pr(s_j^{t_2})}. \tag{5.2}$$

Let

$$Pr(\boldsymbol{\theta}) = \begin{pmatrix} Pr(\theta_1) \\ Pr(\theta_2) \\ \vdots \\ Pr(\theta_{|\Theta|}) \end{pmatrix} \tag{5.3}$$

be the vector of prior type probabilities (i. e. $t = 0$).

As we know both the topology and the parameters of the HMM, calculating the entire probability vector $\boldsymbol{\theta}^t$ is straightforward:

$$\begin{aligned} Pr(\boldsymbol{\theta}^t) &= \begin{pmatrix} Pr(\theta_1^t) \\ Pr(\theta_2^t) \\ \vdots \\ Pr(\theta_{|\Theta|}^t) \end{pmatrix} \\ &= \boldsymbol{P}^t \times Pr(\boldsymbol{\theta}) \end{aligned} \tag{5.4}$$

Please note that in the context of a matrix, the superscript denotes an exponentiation[2]:

$$\boldsymbol{P}^t = \underbrace{\boldsymbol{P} \times \ldots \times \boldsymbol{P}}_{t}. \tag{5.5}$$

[2]For the matrix exponentiation I use a simple divide and conquer approach having a runtime of magnitude $O(|\Theta|^3 \cdot \log t)$. This mark can possibly be further improved by more sophisticated numerical algorithms but since the runtime required for the exponentiation is not the limiting factor of application, I leave this to future work. Compare section 6.1.2 for details regarding the running time depending on t.

Using 5.4, we get:

$$Pr(s_j^t) = \sum_{l=1}^{|\Theta|} f(s_j \,|\, \theta_l) \cdot Pr(\theta_l^t). \tag{5.6}$$

The probability that agent i receives signal $s_j^{t_2}$ given that the type was $\theta_l^{t_1}$ can be computed as follows:

$$Pr(s_j^{t_2} \,|\, \theta_l^{t_1}) = \sum_{o=1}^{|\Theta|} f(s_j \,|\, \theta_o) \cdot Pr(\theta_o^{t_2} \,|\, \theta_l^{t_1}).$$

For the probability of a certain type at time t_2 knowing the type at time t_1 we need to distinguish two cases.

- $t_2 \geq t_1$

  Here, we only need a minor change of equation 5.4:

$$Pr(\boldsymbol{\theta}^{t_2} \,|\, \theta_l^{t_1}) = \boldsymbol{P}^{t_2 - t_1} \times \begin{pmatrix} \theta_1^{t_1} = 0 \\ \vdots \\ \theta_l^{t_1} = 1 \\ \vdots \\ \theta_{|\Theta|}^{t_1} = 0 \end{pmatrix},$$

  i. e. the lth column of $\boldsymbol{P}^{t_2 - t_1}$.

- $t_2 < t_1$

  From Bayes we know

$$Pr(\theta_o^{t_2} \,|\, \theta_l^{t_1}) = \frac{Pr(\theta_l^{t_1} \,|\, \theta_o^{t_2}) \cdot Pr(\theta_o^{t_2})}{Pr(\theta_l^{t_1})}$$

  and we can calculate the $Pr(\theta_l^{t_1} \,|\, \theta_o^{t_2})$ analogue to the $t_2 \geq t_1$ case:

$$Pr(\boldsymbol{\theta}^{t_1} \,|\, \theta_o^{t_2}) = \boldsymbol{P}^{t_1 - t_2} \times \begin{pmatrix} \theta_1^{t_2} = 0 \\ \vdots \\ \theta_o^{t_2} = 1 \\ \vdots \\ \theta_{|\Theta|}^{t_2} = 0 \end{pmatrix}.$$

In the Linear Program, we need $g(s_k^{t_1} \,|\, s_j^{t_2})$ for all $s_k^{t_1}, s_j^{t_2} \in S$. I store this data in a $M \times M$ matrix and extend the calculations of this section to matrix multiplications where possible. Except for the two Bayes transformations, all calculations can be rewritten in such a way (see Appendix A). The resulting code is easier to read, compact and optimized for use with numerical software, such as Matlab or Python's NumPy package. In addition, the feasibility analysis (section 5.5) is easier and more intuitive when interpreting the calculations as linear transformations.

The resulting Linear Program is:

LP 2.

$$min \quad W = \sum_{j=1}^{M} Pr(s_j^{t_2}) \left(\sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \,\cdot\, \tau(s_j^{t_2}, s_k^{t_1}) \right)$$

$$s.t. \quad \sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \left(\tau(s_j^{t_2}, s_k^{t_1}) - \tau(s_h^{t_2}, s_k^{t_1}) \right) > \Delta(s_j^{t_2}, s_h^{t_2}); \quad \forall s_j^{t_2},\, s_h^{t_2} \in S,\, s_j^{t_2} \neq s_h^{t_2}$$

$$\sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \,\cdot\, \tau(s_j^{t_2}, s_k^{t_1}) > C; \quad \forall s_j^{t_2} \in S$$

$$\tau(s_j^{t_2}, s_k^{t_1}) \geq 0; \quad \forall s_j^{t_2},\, s_k^{t_1} \in S$$

5.3 Example

Let us extend the example from section 2.3. As before, we have two types $\theta_1 = B$ and $\theta_2 = G$ emitting two signals $s_1 = l$ and $s_2 = h$. The probabilities given by the base setting are $Pr(G) = 0.7$, $Pr(B) = 1 - Pr(G) = 0.3$ and $f(h|G) = 0.75$, $f(h|B) = 0.35$. The transition matrix is

$$\boldsymbol{P} = \begin{pmatrix} 0.95 & 0.1 \\ 0.05 & 0.9 \end{pmatrix}.$$

and agent 1 received a signal at $t_1 = 0$ while agent 2 received a signal at $t_2 = 2$.

We want to compute the budget-optimal payments for agent 2. The first step is to compute $\boldsymbol{G}$. Therefore, we need the unconditional type probabilities at t_1 and t_2. For $t_1 = 0$ this simply is the prior type probability and for $t_2 = 2$, we compute the vector by equation 5.4:

$$\begin{aligned} Pr(\boldsymbol{\theta}^{t_2}) &= \boldsymbol{P}^{t_2} \times Pr(\boldsymbol{\theta}) \\ &= \begin{pmatrix} 0.9075 & 0.185 \\ 0.0925 & 0.815 \end{pmatrix} \times \begin{pmatrix} 0.3 \\ 0.7 \end{pmatrix} \\ &= \begin{pmatrix} 0.40175 \\ 0.59825 \end{pmatrix} \end{aligned}$$

Then, we need $\boldsymbol{A}^{\theta^{t_2}\times\theta^{t_1}}$. As $t_2 > t_1$, this simply is $\boldsymbol{P}^{t_2-t_1} = \boldsymbol{P}^{t_2}$ which we have already calculated in the preceding equation.
The next step is the calculation of the signals at t_2 conditional on a type at t_1:

$$\begin{aligned}\boldsymbol{A}^{s^{t_2}\times\theta^{t_1}} &= \boldsymbol{F} \times \boldsymbol{A}^{\theta^{t_2}\times\theta^{t_1}} \\ &= \begin{pmatrix} 0.65 & 0.25 \\ 0.35 & 0.75 \end{pmatrix} \times \begin{pmatrix} 0.9075 & 0.185 \\ 0.0925 & 0.815 \end{pmatrix} \\ &= \begin{pmatrix} 0.613 & 0.324 \\ 0.387 & 0.676 \end{pmatrix}\end{aligned}$$

Before we can calculate $\boldsymbol{A}^{\theta^{t_1}\times s^{t_2}}$ using Bayes' Theorem, we have to compute the unconditional signal probability at t_2 which is easily done multiplying $\boldsymbol{F}$ with $Pr(\boldsymbol{\theta}^{t_2})$:

$$\begin{aligned}\boldsymbol{s}^{t_2} &= \boldsymbol{F} \times Pr(\boldsymbol{\theta}^{t_2}) \\ &= \begin{pmatrix} 0.65 & 0.25 \\ 0.35 & 0.75 \end{pmatrix} \times \begin{pmatrix} 0.40175 \\ 0.59825 \end{pmatrix} \\ &= \begin{pmatrix} 0.4107 \\ 0.5893 \end{pmatrix}\end{aligned}$$

Now we have all data needed to compute the probabilities of a type at t_1 given a signal at t_2 using Bayes' Theorem (compare equation 5.2) resulting in[3]:

$$\boldsymbol{A}^{\theta^{t_1}\times s^{t_2}} = \begin{pmatrix} 0.4478 & 0.197 \\ 0.5522 & 0.803 \end{pmatrix}$$

With this result we may finally compute the $\boldsymbol{G}$ we need for the coefficients of LP 2:

$$\begin{aligned}\boldsymbol{G} &= \boldsymbol{F} \times \boldsymbol{A}^{\theta^{t_1}\times s^{t_2}} \\ &= \begin{pmatrix} 0.65 & 0.25 \\ 0.35 & 0.75 \end{pmatrix} \times \begin{pmatrix} 0.4478 & 0.197 \\ 0.5522 & 0.803 \end{pmatrix} \\ &= \begin{pmatrix} 0.4291 & 0.3288 \\ 0.5709 & 0.6712 \end{pmatrix}\end{aligned}$$

The objective function requires the products of the signal probability at t_2 and the respective entry in $\boldsymbol{G}$. The respective equations are:

[3] In the rest of the example, I do some minor rounding but continue the calculation with higher accuracy.

$$
\begin{aligned}
Pr(h^{t_2}) \cdot g(h^{t_1}|h^{t_2}) &= 0.3956 \\
Pr(h^{t_2}) \cdot g(l^{t_1}|h^{t_2}) &= 0.1938 \\
Pr(l^{t_2}) \cdot g(h^{t_1}|l^{t_2}) &= 0.2345 \\
Pr(l^{t_2}) \cdot g(l^{t_1}|l^{t_2}) &= 0.1762
\end{aligned}
$$

Setting all $\Delta(s_j|s_h)$ ($h \neq j$) to 0.15 and C to 0.1, we can now write down the entire LP. For reasons of clarity, I further round the values to 2 digits after the decimal point. Furthermore, all τ are required to be ≥ 0:

LP 3.

$$
\begin{aligned}
min \quad & 0.18\,\tau(l,l) + 0.23\,\tau(l,h) + 0.19\,\tau(h,l) + 0.4\,\tau(h,h); \\
s.t. \quad & 0.43\,\tau(l,l) + 0.57\,\tau(l,h) - 0.43\,\tau(h,l) - 0.57\,\tau(h,h) > 0.15; \\
& -0.33\,\tau(l,l) - 0.67\,\tau(l,h) + 0.33\,\tau(h,l) + 0.67\,\tau(h,h) > 0.15; \\
& 0.43\,\tau(l,l) + 0.57\,\tau(l,h) > 0.1; \\
& 0.33\,\tau(h,l) + 0.67\,\tau(h,h) > 0.1;
\end{aligned}
$$

The optimal payment scheme that is computed by the solver is thus:

		$r(i)$ h	$r(i)$ l
i	h	1.86	0
	l	0	1.13

Please note that the payments are given only for agent 2 as the setting is no longer symmetric.

5.4 Choosing the reference rater

I will rate agent i against the announcement of the preceding agent. The only exception to the rule is the first agent who is rated against the second (i. e. against the following) agent. Besides its simplicity, this procedure has two major advantages: First, only the very first agent has to wait for her payments while all other agents can be scored right away. Second, the information contained in past signal announcements can be released quickly. Generally, we have to keep the current (i. e. updated) type beliefs undisclosed as long as the information contained in the updated beliefs are informative about the signal announced by the reference agent. The immediate release of this information is especially important in the Markov setting where types may change and information thus outdates.

Depending on the actual setting, other choice procedures may be advantageous. For example, one might consider rating against the agent that is closest with regards to the time-steps. Similarily, one may select the neighboring agent that is the cheapest. Both of these procedures have the drawback that the mechanism needs to withhold its payments until the reference rater is chosen. Another intuitive objective is finding the cheapest rating pairs[4]. Since most of the time, the center will not know how many agents will arrive and when, this procedure would require an online approach. I leave this for future work.

5.5 Feasible region of the LP

The LP may be infeasible and this section is devoted to the analysis of these infeasible configurations. That is, settings in which we are unable to construct a mechanism.

As is quickly seen, the objective function is bounded if the constraints are feasible. All factors are positive or zero and thus have a lower bound at zero which corresponds to zero costs. The analysis of the feasible region of the LP is more difficult.

5.5.1 LP feasible ⇔ Stochastic relevance

Suppose for a moment we already have the payment matrix induced by $\tau(\cdot|\cdot)$ (see Appendix A). What is the expected payment to agent i given her signal and announcement (the latter not necessarily truthful)?

Definition 4. *Let $s_j^{t_2}$ be the signal received by agent i and let a_h^i be her announcement. The* generalized expected payment *is the expected payment to agent i given that $r(i)$ is announcing her signal truthfully.*

$$\begin{aligned} E(a_h^i, s_j^{t_2}) &= \sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \cdot \tau(a_h^i, a_k^{r(i)}) \\ &= \sum_{k=1}^{M} \tau(a_h^i, a_k^{r(i)}) \cdot g(s_k^{t_1} \,|\, s_j^{t_2}) \end{aligned}$$

[4]Note that 'pairs' are not necessarily symmetric, i. e. if $r(i)$ is the reference agent to agent i, this does not induce that the relation holds the other way round, as well.

This can be captured in a two-dimensional matrix holding all $E(\cdot|\cdot)$:

$$\begin{aligned}\boldsymbol{E} &= \begin{pmatrix} E(a_1^i, s_1^{t_2}) & \dots & E(a_1^i, s_M^{t_2}) \\ \vdots & \ddots & \vdots \\ E(a_M^i, s_1^{t_2}) & \dots & E(a_M^i, s_M^{t_2}) \end{pmatrix} \\ &= \boldsymbol{\tau} \times \boldsymbol{G}\end{aligned}$$

Proposition 5. *LP 2 is feasible if and only if the signal observation at time t_2 is stochastically informative about the signal observation at time t_1*

Proof.

- ”⇒” I prove the *modus tollens* equivalent of the expression, i. e. given the signal observations are *not* stochastically informative, the LP is *not* feasible.

  The signal observations are not stochastically informative, so that at least two different columns in $\boldsymbol{G}$[5] have the same entries. Without loss of generality, let $g(\cdot|s_h)$ and $g(\cdot|s_j)$ be these two colums with equal entries. Multiplying $\boldsymbol{\tau}$ with $g(\cdot|s_h)$ and $g(\cdot|s_j)$ results in the columns $E(\cdot, s_h)$ and $E(\cdot, s_j)$, respectively. As in both multiplications the factors are equal, the resulting columns $E(\cdot, s_h)$ and $E(\cdot, s_j)$ are equal, as well. The first constraint group in LP 2 (honesty constraints), though, is requiring that the expected payment of the honest announcement is *strictly larger* than any of the dishonest announcements. In particular: $E(a_h, s_h) > E(a_j, s_h) = E(a_j, s_j) > E(a_h, s_j) = E(a_h, s_h)$ ↯

- ”⇐” To show that LP 2 is feasible if all columns in $\boldsymbol{G}$ are different, I construct a $\boldsymbol{\tau}$ that complies with all constraints in LP 2.

  It is simple to scale a $\boldsymbol{\tau}$ that complies with the first group of constraints (honesty) to comply with the second (participation) and third (non-negative transfers), respectively. Therefore, I will first elucidate on the honesty constraints.

  In chapter 3 I already introduced a way to construct a feasible $\boldsymbol{\tau}$: strictly proper scoring rules. Let us construct $\boldsymbol{\tau}$ by applying such a scoring rule $R(\cdot|\cdot)$ to $\boldsymbol{G}$ as mentioned in section 3.2:

$$\tau(a_h^i, a_k^{r(i)}) = R(a_k^{r(i)}|a_h^i)) \qquad (5.7)$$

[5] for definition of these matrices, I again refer to Appendix A)

The generalized expected payment is then

$$E(a_h^i, s_j^{t_2}) = \sum_{k=1}^{M} g(s_k^{t_1} \,|\, s_j^{t_2}) \;\cdot\; R(a_k^{r(i)} | a_h^i)) \tag{5.8}$$

which is solely maximized by the honest announcement (see chapter 3), i. e. the expected payment of the honest announcement is at least $\epsilon > 0$ larger than any announcement with different probability values and thus also strictly larger than any of the dishonest announcements for all $s_j^{t_2}$.

So far, $\boldsymbol{\tau}$ does not necessarily comply with the external benefits from lying, $\Delta(s_j, a_h)$. From chapter 3 we know that neither multiplying $\boldsymbol{\tau}$ with a scalar λ nor adding a constant changes its incentive properties. A multiplication with λ results in a $\boldsymbol{\tau}'$ whose honesty bounds (i. e. the difference between the honest and dishonest announcements) are scaled by λ, as well. These honesty bounds are actually a constraint on $\boldsymbol{E}$: For every honest announcement we have $M-1$ dishonest announcements and the corresponding expected payments are noted down in $\boldsymbol{E}$. So for every column in $\boldsymbol{E}$ we look at the (expected payment) difference between the honest announcement s_j and every dishonest announcement a_h and divide it to the lower bound it is supposed to conform to, i. e. $\Delta(s_j, a_h)$. If we pick λ to be the maximum of all these fractions, we receive a $\boldsymbol{\tau}'$ conforming with the external benefits:

$$\max_{j,h} \frac{E(a_j, s_j) - E(a_h, s_j)}{\Delta(s_j, a_h)}. \tag{5.9}$$

To make $\boldsymbol{\tau}'$ complying with the second group of constraints, i. e. voluntary participation, we may simply add $|\min(\tau(\cdot,\cdot)')| + C$ to all entries in $\boldsymbol{\tau}'$. That way, all $\tau(\cdot,\cdot)'$ are positive, as well.

□

5.5.2 Stochastic relevance

As shown in the preceding subsection, finding the feasible region of the LP comes down to analyzing the stochastic relevance of the observations made by agent i and her reference agent $r(i)$. Finding an exact description of the stochastically informative settings is very difficult even in the base case without the Markov Process[6]. Thus, I will

[6]Neither MRZ nor Jurca and Faltings fully analyze these settings. In fact, while the larger part of Radu Jurca's PhD-thesis is about this model, for the infeasibility analysis, he simply refers to the original paper by MRZ who use scoring rules. In addition, the models of MRZ and Jurca and Faltings slightly differ in that the latter allow different types to have the same signal distribution and the prior probability types to be 0. This not only questions a simple reference but further complicates the analysis. On the other hand, in the final version of MRZ's paper, the analysis is substituted by a reference to a paper never written (by the same authors). This technique was coined *proof by phantom reference*.

restrict myself to the somewhat intuitive description of these settings made by MRZ [15], give a matrix formulation of it and explain a special type of stochastic irrelevance that comes with the introduction of the MP.

I will describe the set of parameters that *fail* stochastic relevance. All sets of parameters that do not fail stochastic relevance obey it and will thus make the LP feasible. Let $s_j^{t_2}$ and $s_h^{t_2}$ with $j \neq h$ be two signal observations that generate the same $r(i)$ signal posterior distribution, i. e. $g(s_k^{t_1} \mid s_j^{t_2}) = g(s_k^{t_1} \mid s_h^{t_2})$ for all $k \in \{1, \ldots, M\}$.

Expanding this gives us:

$$
\begin{aligned}
g(s_k^{t_1} \mid s_j^{t_2}) - g(s_k^{t_1} \mid s_h^{t_2}) &= 0 \qquad \forall k \\
\sum_{\theta \in \Theta} f(s_k|\theta) \cdot Pr(\theta^{t_1}|s_j^{t_2}) - \sum_{\theta \in \Theta} f(s_k|\theta) \cdot Pr(\theta^{t_1}|s_h^{t_2}) &= 0 \qquad \forall k \\
\sum_{\theta \in \Theta} f(s_k|\theta) \left(Pr(\theta^{t_1}|s_j^{t_2}) - Pr(\theta^{t_1}|s_h^{t_2})\right) &= 0 \qquad \forall k
\end{aligned}
\tag{5.10}
$$

Rewriting these calculations as matrix multiplications (linear transformations) has two advantages: First, it is more compact and natural to deal with matrix multiplications than with arrays of regular calculations and, second, we can take advantage of theorems proven in Linear Algebra and apply them to our setting.
Let

$$
Pr(\boldsymbol{\theta^{t_1}}|s_j^{t_2}) = \begin{pmatrix} Pr(\theta_1^{t_1}|s_j^{t_2}) \\ Pr(\theta_2^{t_1}|s_j^{t_2}) \\ \vdots \\ Pr(\theta_{|\Theta|}^{t_1}|s_j^{t_2}) \end{pmatrix}
$$

and $Pr(\boldsymbol{\theta^{t_1}}|s_h^{t_2})$ analogue. In addition, let $\boldsymbol{u}$ denote the difference between the two type posteriors:

$$
\begin{aligned}
\boldsymbol{u} &= \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{|\Theta|} \end{pmatrix} \\
&= Pr(\boldsymbol{\theta^{t_1}}|s_j^{t_2}) - Pr(\boldsymbol{\theta^{t_1}}|s_h^{t_2}).
\end{aligned}
$$

Then, the infeasible region is the solution to

$$\boldsymbol{F} \times \boldsymbol{u} = \mathbf{0} \tag{5.11}$$

while $\mathbf{0}$ denotes the null vector (compare equation 5.10).

Therefore, there are two cases that can make LP 2 infeasible:

1. $\boldsymbol{F}$ is such that for two different type distributions, it generates the same (posterior) signal distribution.

2. The (posterior) type belief at t_1 is the same for two distinct signal observation at t_2.

According to MRZ it is straightforward to show that these restrictions are only satisfied non-generically and that the set of distributions $\boldsymbol{F}$ and $\boldsymbol{\theta}$ that satisfy them have Lebesgue measure zero. Note that if only one condition fails, the LP becomes infeasible. Despite this being a strong limitation from a mathematical point of view, MRZ argue that these cases are rare in practical implementations and even small perturbations of the believes make the setting stochastically informative again.

Intuitively, the first restriction should not be given very often if different types generate different probability distributions (as we demanded). If $\boldsymbol{F}$ is quadratic (i. e. $M = |\Theta|$), this only is the case for (non-invertible) matrices with rank strictly smaller than M.

The second situation is especially interesting with regards to the Markov Process: First, $\boldsymbol{P}$ may may have a format, such that it loses information in between two *neighboring* time steps and, second, $\boldsymbol{P}^t$ may converge to a steady state, i. e. the MP forgets where it came from resulting in type posteriors that—after some t—are identical in their floating point representation. Regarding the latter, I refer to section 6.2.2.

From equation 5.4 we know how to compute the unconditional type update, i. e. the change in type belief that occurs simply through the time passing by without any additional information through signal announcements[7]. If $\boldsymbol{P}$ is such that *after* the update information about the type beliefs *before* the update is lost, O^i cannot be stochastically informative about $O^{r(i)}$ and the LP will thus be infeasible, as well.

[7] How to update the type beliefs taking into account signal announcements, see section 7.

The analysis is similar to that of equation 5.11. Let $\boldsymbol{v}$ be the difference of two type vectors at t_1 and t_2, respectively:

$$\begin{aligned} \boldsymbol{v} &= \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_{|\Theta|} \end{pmatrix} \\ &= Pr(\boldsymbol{\theta^{t_1}}) - Pr(\boldsymbol{\theta^{t_2}}). \end{aligned}$$

Then, the MP breaks the stochastic connection between two neighboring time steps if and only if for $\boldsymbol{v} \neq 0$ it holds that

$$\boldsymbol{P} \times \boldsymbol{v} = \mathbf{0}. \tag{5.12}$$

Since $\boldsymbol{P}$ is quadratic by definition, this is given if and only if $\boldsymbol{P}$ has rank lower than $|\Theta|$. This is why the setting demands that $\boldsymbol{P}$ is non-singular which in the case of quadratic matrices is an equivalent statement.

Chapter 6

Experimental Results

If not stated otherwise, the parameters for the experimental settings are created as described in Appendix B.

6.1 Running Time

Linear Programs can be solved in weakly polynomial running time. For example, the Ellipsoid algorithm finds the optimal solution (or technically a very good approximation of it) in $O(n^4 L)$ where n is the number of variables (M^2 in our setting) and L is the number of input bits (i. e. the number of bits needed to encode the LP).

While having an exponential worst-case running time, the Simplex method usually performs better in practice. Thus, in order to tell whether the feedback mechanism is applicable in real-world settings, I empirically evaluate it on a customary Notebook with Intel Core2Duo (1.6GHz) and 2GB of RAM running Windows XP. The solving process is essentially twofold: First, a small Python 2.5 program computes all conditional probabilities $g(s_k^{t_1} \mid s_j^{t_2})$ that are needed to build LP 2 while in the second step, the LP is composed and passed to an external solver (lp_solve 5.5.10 via its Python API).

6.1.1 Running time dependent on M

Figure 6.1 shows the running time for different values of M. I used gnuplot 4.2.3 to make a least-square-fit against the runtime data. The fitted curve f(x) corresponds to $0.0235369x^3 - 0.186207x^2 + 1.19075x + 0.274942$ and fitting curves with a polynomial of order lower than 3 results in bad approximations.

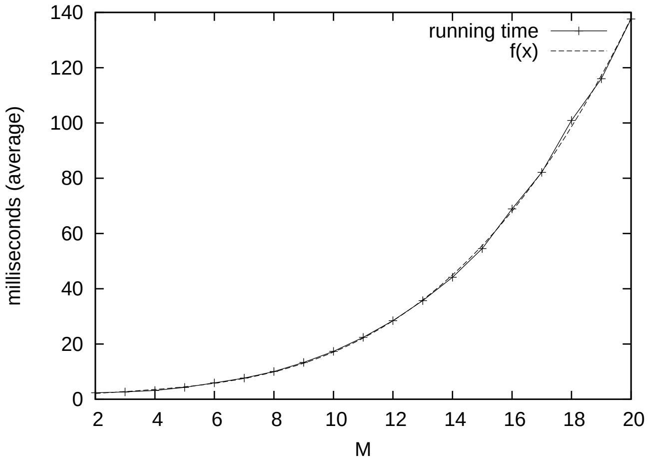

Figure 6.1: Running time for different values of M

6.1.2 Running time depending on $\Delta(t)$

We are also interested in the runtime that comes with the introduction of the MP. Let the unary $\Delta(t)$ denote $t_2 - t_1$. Figure 6.2 shows the running time depending on the time steps that lie in between the two agents while $\Delta(t) = 0$ corresponds to 100[1].

Interestingly, the runtime not only grows with $\Delta(t)$ but becomes larger for large M, as well. This is due to the matrix exponentiation algorithm whose runtime is $O(|\Theta|^3 \cdot \log t)$ while in our settings, $|\Theta|$ equals M and t corresponds to $\Delta(t)$. For larger $\Delta(t)$, the factor $|\Theta|^3 = M$ gets more influence on the running time if only multiplied by $\log t$.

Nonetheless, the running time of the Markov Process is unlikely to be the limiting factor for application. Values for $\Delta(t)$ will rarely be much higher than 40 so that the slowdown will be within 10% even for settings with a large signal set. Secondly, for some $\boldsymbol{P}$, the setting may be infeasible by then (compare figure 6.4) and finally, the expected costs we aim to minimize are growing at a higher pace with every time step in between the two ratings (compare figure 6.3).

6.2 Payment behavior

6.2.1 Expected Costs

What are the expected costs depending on $\Delta(t)$? Figure 6.3 shows the expected costs for different values of stochastic movement inherent in $\boldsymbol{P}$[2] as reflected by ϵ (see Ap-

[1] Please note that for $M = 5$ I took 10000 settings instead of 1000.

[2] For the expected payment analysis I focus on the behavior regarding $\Delta(t)$ and $\boldsymbol{P}$. Simply applying the random setting as described in Appendix B to compare the behavior for different values of M and $|\Theta|$ may lead to wrong conclusions. This is due to the fact that the noise level of $\boldsymbol{F}$ and $\boldsymbol{P}$ does not naturally extend

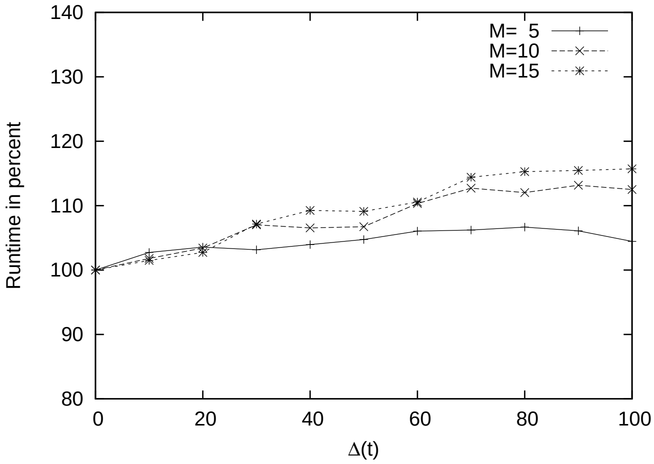

Figure 6.2: Normalized running time depending on $\Delta(t)$

pendix B for a description of the random setting).

As expected, the more time lies in between the two ratings the higher are the mechanism's cost. A larger ϵ corresponds to more type perturbations through the MP (also see section 6.2.2). For larger $\Delta(t)$ the probabilities for a certain type at t_2 conditional on the types at t_1 become more alike as the Markov Process converges to its steady state and so do the columns of $\boldsymbol{G}$. As these are the coefficients of the LP and they are getting closer, the solver needs larger τ to separate the honest from the dishonest announcements. Note that our external benefits from lying are constant and thus independent of $\Delta(t)$.

6.2.2 Stochastic Relevance

As mentionend in section 5.5, if $\boldsymbol{P}$ is non-trivial (i. e. it adds uncertainty to the setting), large $\Delta(t)$ may break the stochastic connection between the signal observations because of floating point inaccuracies. That is, the LP becomes infeasible because two minimally different vectors in $\boldsymbol{G}$ are represented by the same floating point numbers resulting in a stochastically irrelevant $\boldsymbol{G}$.

The point at which a setting becomes infeasible obviously depends on the magnitude of stochastic movement. Figure 6.4 shows the fraction of infeasible settings depending on $\Delta(t)$ for different values of ϵ and 5000 settings. As one would expect, higher perturbations on the type vectors result in quicker convergence of $\boldsymbol{G}$.

to different numbers of types or signals.

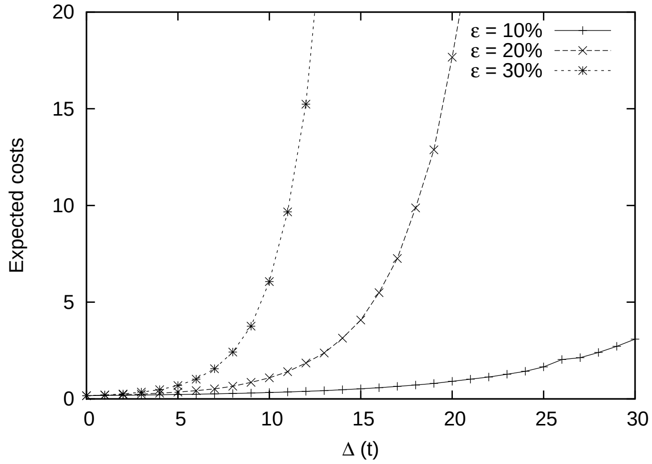

Figure 6.3: Expected costs for different values of ϵ

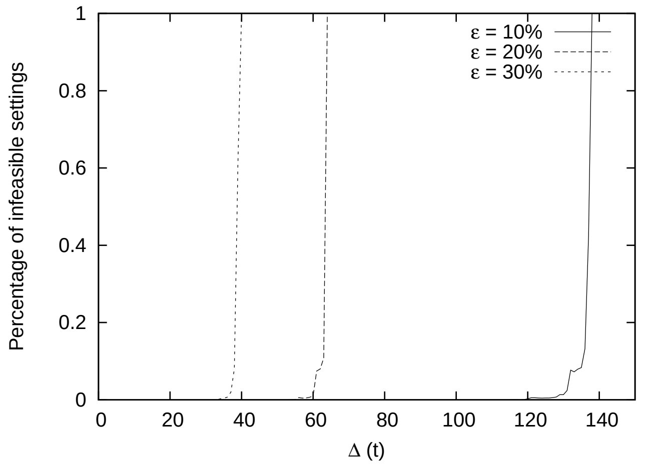

Figure 6.4: Percentage of infeasible settings for different values of ϵ

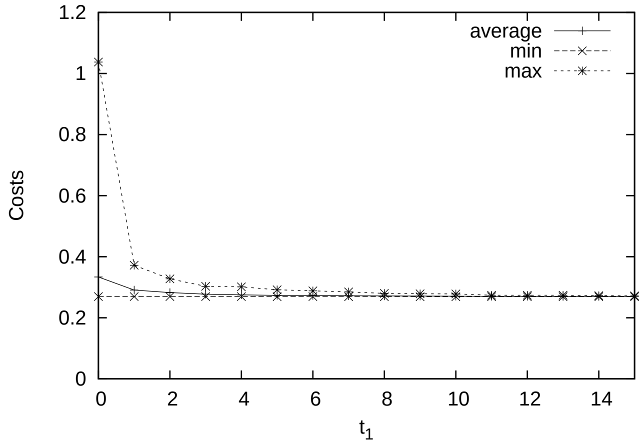

Figure 6.5: Costs for fixed $\Delta(t) = 10$ at different positions of t_1

6.2.3 Convergence

Since we usually rate every agent against her preceding agent (compare section 5.4), our t_1 is mostly set to 0 (also compare Appendix B for the description of the random settings), so that $\Delta(t)$ corresponds to the interval from 0 to t_2 and is thus fixed. For other pairing methods, though, it is insightful to study the behavior for fixed $\Delta(t)$ at different positions for t_1. Figure 6.5[3] shows the costs in both expectation and range for different t_1.

One can see that the payments rapidly converge. The level they converge to corresponds to the payments that are computed for the steady state type vector for fixed $\Delta(t)$. Most transition matrices $\boldsymbol{P}$ that are applicable in our setting have a unique steady state (they are irreducible[4] and aperiodic[5] [13]). The intuition is that seen from $t = 0$ (i. e. the mechanism's position) the type beliefs at t_1 and t_2 with fixed $\Delta(t)$ become more alike for large t_1, t_2. In fact, as the MP perturbes them, they become closer to the steady state vector so that the mechanism computes payments that are converging to these steady state payments. For further explanations on steady states in Markov processes, I refer to [13, p. 194-216].

[3]For some reason, LaTeX had problems including an eps graphic with gnuplot's errorbars, so that I used pointlines instead.

[4]it is possible to get to any state from any state

[5]A state q_i has period k if any return to state q_i must occur in multiples of k time steps. If $k = 1$, then the state is said to be aperiodic. If all states are aperiodic, the MP is said to be aperiodic.

Chapter 7

Updating the Type Beliefs

7.1 Motivation

Up to this point, we have constructed payments that induce agents to give honest feedback about the signals they receive. What is supposed to be published, though, is the probability that the product has a certain type.

The literature on hidden Markov models[1] describes three basic algorithms answering the following questions:

1. What is the probability of a certain signal sequence given the model?

2. Given a signal sequence, what is the most likely hidden sequence of states?

3. Given only the topology of the model, what are the parameters that maximize the probability for the observed signal sequence?

Unfortunately, none of these match our situation. Note that the most likely hidden sequence of states is not what we are looking for. Potential customers are only interested in the quality the product has right now (or in the future) but not in the quality the product most likely once had. Given the topology and the model, the future of the MP only depends on the most recent state. Thus, what we are interested in is the probability distribution of the most recent state given the signal sequence announced by the agents.

Note that simply taking the last state of the most likely hidden sequence of states will not help us for two reasons: First, it only gives us a single state and not a probability distribution. Second, it may well be that the most likely last state is not at the end of the most likely *sequence* of states.

Let $N = |\Theta|$ be the number of states and T the most recent time step. The naive approach goes through all N^{T+1} possible state sequences (the index begins with 0) and

[1] see for example [7, p.409-429]

is thus infeasible for large T. However, we can use Dynamic Programming to come up with an algorithm linear in T.

7.2 One Announcement per Time Step

Let us first consider a setting with exactly one observation per time step. In a next step, we will extend this to incorporate both *null transitions* and *multiple observations*.

The outer part of the algorithm consists of computing the conditional probability that the most recent state q_T is θ given the vector of signal announcements $\boldsymbol{Y}$. Extending this conditional probability gives us

$$Pr(q_T = \theta|\boldsymbol{Y}) = \frac{Pr(q_T = \theta \cap \boldsymbol{Y})}{Pr(\boldsymbol{Y})}. \tag{7.1}$$

We begin with iteratively calculating $Pr(q_T = \theta \cap \boldsymbol{Y})$. Let s^t denote the signal announcement at time step t while $\boldsymbol{Y^t} = (s^0, s^1, \ldots, s^t)$ is the timely-ordered vector of all these signals up to time step t. Calculating the base case $t = 0$ is straightforward:

$$Pr(q_0 = \theta \cap s^0) = Pr(\theta) \cdot f(s^0|\theta). \tag{7.2}$$

Computing the joint probability at time $t + 1$ can be reduced to computing N joint probabilities at time t:

$$Pr(q_{t+1} = \theta \cap \boldsymbol{Y^{t+1}}) = f(s^{t+1}|\theta) \cdot \sum_{i=1}^{N} \left[Pr(q_t = \theta_i \cap \boldsymbol{Y^t}) \cdot Pr(\theta_i \to \theta)\right] \tag{7.3}$$

What is left, is the denominator of equation 7.1. Following the law of *total probability*, summing up all joined probabilities at $t = T$ for all $\theta \in \Theta$ suffices and we receive

$$Pr(\boldsymbol{Y}) = \sum_{i=1}^{N} Pr(q_T = \theta_i \cap \boldsymbol{Y}). \tag{7.4}$$

Extending equation 7.1 results in:

$$Pr(q_T = \theta|\boldsymbol{Y}) = \frac{Pr(q_T = \theta \cap \boldsymbol{Y})}{\sum\limits_{i=1}^{N} Pr(q_T = \theta_i \cap \boldsymbol{Y})}. \tag{7.5}$$

The runtime of the algorithm is $O(N^2\,T)$. Please see section 7.3.1 for a detailed discussion of the runtime.

7.3 Multiple Announcements per Time Step

Since the agents do not necessarily arrive one every time step, we need to incorporate both *null transitions* and *multiple observations*. I will first interpret the announcements made in a single time step as a vector. Thereafter, I will show how compute the conditional probability that interprets the announcements in a single time step as a set (i. e. without ordering inside a single time step). We will see that the results are equal.

7.3.1 Ordered Tuple Case

Let $\boldsymbol{o^t}$ denote the *ordered tuple* of signal announcements in time step t and $\boldsymbol{O^t} = (\boldsymbol{o^0}, \boldsymbol{o^1}, \ldots, \boldsymbol{o^t})$ the timely-ordered tuple of these announcement tuples up to time step t. Then, $h(\boldsymbol{o^t}|\theta)$ is the probability of $\boldsymbol{o^t}$ conditional on state θ.

$$h(\boldsymbol{o^t}|\theta) = \begin{cases} 1, & \text{if } |\boldsymbol{o^t}| = 0 \\ \prod\limits_{s \in \boldsymbol{o^t}} f(s|\theta), & \text{else.} \end{cases} \tag{7.6}$$

Analogue to equation 7.2, the base case is:

$$Pr(q_0 = \theta \cap \boldsymbol{o^0}) = Pr(\theta) \cdot h(\boldsymbol{o^0}|\theta) \tag{7.7}$$

Adapting the iteration step (equation 7.3) yields:

$$Pr(q_{t+1} = \theta \cap \boldsymbol{O^{t+1}}) = h(\boldsymbol{o^{t+1}}|\theta) \cdot \sum_{i=1}^{N} \left[Pr(q_t = \theta_i \cap \boldsymbol{O^t}) \cdot Pr(\theta_i \rightarrow \theta) \right]. \tag{7.8}$$

Similar to equation 7.5, the outer equation for the ordered case is

$$\begin{aligned} Pr(q_T = \theta|\boldsymbol{O}) &= \frac{Pr(q_T = \theta \cap \boldsymbol{O})}{Pr(\boldsymbol{O})} \\ &= \frac{Pr(q_T = \theta \cap \boldsymbol{O})}{\sum\limits_{k=1}^{N} Pr(q_T = \theta_k \cap \boldsymbol{O})}. \end{aligned} \tag{7.9}$$

Running time

The computation of $h(\boldsymbol{o^t}|\theta)$ for a single θ in a single time step is in $O(|\boldsymbol{o^t}|)$. It needs to be computed for all θ in all time steps, resulting in

$$\underbrace{N \cdot |\boldsymbol{o^0}| + N \cdot |\boldsymbol{o^1}| + \cdots + N \cdot |\boldsymbol{o^T}|}_{(T+1)\times} = N \cdot \left(|\boldsymbol{o^0}| + |\boldsymbol{o^1}| + \cdots + |\boldsymbol{o^T}|\right) = N \cdot |\boldsymbol{O^T}|$$

for this part. Note that the $h(\boldsymbol{o^t}|\theta)$ can be computed independently of the Dynamic Programming part.

The runtime of the latter is the same as that of section 7.2. The base case has one multiplication for every N while every iteration has N multiplications for every of the N types. For this part, this results in a runtime of $O(N^2\,T)$. Putting together the two parts results in a runtime of

$$O(N^2\,T + N\,|\boldsymbol{O^T}|). \tag{7.10}$$

7.3.2 Unordered Tuple Case

In this part, I am going to interpret the announcements made in a single time step as an *unordered* tuple as this is closer to our setting's intuition that announcements are made concurrently. Fortunately, we can reduce the computation of the conditional probability in the unordered case to the respective computation in the ordered case. In fact, the values are identical.

Slightly abusing the notation, let $\boldsymbol{s^t}$ be the unordered tuple representing the signal announcements in t and let $\boldsymbol{Y^t} = (\boldsymbol{s^0}, \boldsymbol{s^1}, \ldots, \boldsymbol{s^t})$ now denote the timely-ordered vector made up of the unordered tuples up to time step t.

Proposition 6. *The conditional probability in the unordered multiset case,* $Pr(q_T = \theta|\boldsymbol{Y})$*, equals the conditional probability in the ordered tuple case,* $Pr(q_T = \theta|\boldsymbol{O})$*.*

Proof. The difference to the ordered case is that the joined probabilities for ordered tuples need to be multiplied with the number of possible orderings for $\boldsymbol{s^t}$. The number of possible orderings in time step t is computed by the multinomial coefficient:

$$m(\boldsymbol{s^t}) = \binom{|\boldsymbol{s^t}|}{|s_1 \in \boldsymbol{s^t}|, \ldots, |s_M \in \boldsymbol{s^t}|} = \frac{|\boldsymbol{s^t}|!}{|s_1 \in \boldsymbol{s^t}|! \cdot \ldots \cdot |s_M \in \boldsymbol{s^t}|!}. \tag{7.11}$$

The base case using equation 7.7 is:

$$Pr(q_0 = \theta \cap \boldsymbol{s^0}) = m(\boldsymbol{s^0}) \cdot Pr(q_0 = \theta \cap \boldsymbol{o^0}). \tag{7.12}$$

Similarily, the iteration refers to equation 7.8:

$$
\begin{aligned}
Pr(q_{t+1} = \theta \cap \boldsymbol{Y^{t+1}}) &= m(\boldsymbol{s^{t+1}}) \cdot h(\boldsymbol{o^{t+1}}|\theta) \cdot \\
&\quad \sum_{i=1}^{N} \left[Pr(q_t = \theta_i \cap \boldsymbol{Y^t}) \cdot Pr(\theta_i \rightarrow \theta) \right] \\
&= m(\boldsymbol{s^{t+1}}) \cdot h(\boldsymbol{o^{t+1}}|\theta) \cdot \prod_{j=1}^{t} m(\boldsymbol{s^j}) \cdot \\
&\quad \sum_{i=1}^{N} \left[Pr(q_t = \theta_i \cap \boldsymbol{O^t}) \cdot Pr(\theta_i \rightarrow \theta) \right] \\
&= h(\boldsymbol{o^{t+1}}|\theta) \cdot \prod_{j=1}^{t+1} m(\boldsymbol{s^j}) \cdot \\
&\quad \sum_{i=1}^{N} \left[Pr(q_t = \theta_i \cap \boldsymbol{O^t}) \cdot Pr(\theta_i \rightarrow \theta) \right]
\end{aligned}
\tag{7.13}
$$

Putting together the outer calculation (similar to equation 7.1), we see that the multinomials can be cancelled out (also compare equation 7.8 and 7.9):

$$
\begin{aligned}
Pr(q_T = \theta|\boldsymbol{Y}) &= \frac{Pr(q_T = \theta \cap \boldsymbol{Y})}{\sum_{k=1}^{N} Pr(q_T = \theta_k \cap \boldsymbol{Y})} \\
&= \frac{h(\boldsymbol{o^T}|\theta) \cdot \prod_{j=1}^{T} m(\boldsymbol{s^j}) \cdot \sum_{i=1}^{N} \left[Pr(q_{T-1} = \theta_i \cap \boldsymbol{O^{T-1}}) \cdot Pr(\theta_i \rightarrow \theta) \right]}{\sum_{k=1}^{N} h(\boldsymbol{o^T}|\theta_k) \cdot \prod_{j=1}^{T} m(\boldsymbol{s^j}) \cdot \sum_{i=1}^{N} \left[Pr(q_{T-1} = \theta_i \cap \boldsymbol{O^{T-1}}) \cdot Pr(\theta_i \rightarrow \theta_k) \right]} \\
&= \frac{h(\boldsymbol{o^T}|\theta) \cdot \sum_{i=1}^{N} \left[Pr(q_{T-1} = \theta_i \cap \boldsymbol{O^{T-1}}) \cdot Pr(\theta_i \rightarrow \theta) \right]}{\sum_{k=1}^{N} h(\boldsymbol{o^T}|\theta_k) \cdot \sum_{i=1}^{N} \left[Pr(q_{T-1} = \theta_i \cap \boldsymbol{O^{T-1}}) \cdot Pr(\theta_i \rightarrow \theta_k) \right]} \\
&= Pr(q_T = \theta|\boldsymbol{O})
\end{aligned}
\tag{7.14}
$$

□

Thus, if we need to compute the updated type probabilities with *multiple observations* and *null transitions* and we have the announcements per time step in an unordered tuple, we can spare computing multinomials and use the easier ordered tuple case of section 7.3.1.

7.4 Example

I continue with the example from section 2.3 and its extension from section 5.3. As before, we have two types $\theta_1 = B$ and $\theta_2 = G$ emitting two signals $s_1 = l$ and $s_2 = h$. The probabilities given by the base setting are $Pr(G) = 0.7$, $Pr(B) = 1 - Pr(G) = 0.3$ and $f(h|G) = 0.75$, $f(h|B) = 0.35$. The transition matrix[2] is

$$\boldsymbol{P} = \begin{pmatrix} 0.95 & 0.1 \\ 0.05 & 0.9 \end{pmatrix}$$

and I will refer to the transition probabilities as exemplified by $p_{GB} = Pr(G \to B) = 0.1$. Furthermore, we are in $t = 2$ and the agent announcements are $\boldsymbol{Y} = (\{h\}, \{\}, \{l, h\})$.

What is the updated type belief in $T = 2$ given $\boldsymbol{Y}$? From the previous section we know that it is sufficient computing

$$Pr(q_2 = \theta|\boldsymbol{Y}) = Pr(q_2 = \theta|\boldsymbol{O})$$

with $\boldsymbol{O} = ((h), (), (l, h))$, i. e. $\boldsymbol{Y}$ in a fixed (but arbitrary) order. Please note, that I will omit some of the vector brackets for better readability.

We begin with computing the base cases:

$$\begin{aligned} Pr(q_0 = G \cap h) &= Pr(G) \cdot h(h|G) \\ &= Pr(G) \cdot f(h|G) \\ &= 0.7 \cdot 0.75 \\ &= 0.525 \end{aligned}$$

$$\begin{aligned} Pr(q_0 = B \cap h) &= Pr(B) \cdot h(h|B) \\ &= Pr(B) \cdot f(h|B) \\ &= 0.3 \cdot 0.35 \\ &= 0.105 \end{aligned}$$

[2] remember that we use a *left* stochastic matrix, see 5.1

$$
\begin{aligned}
Pr(q_1 = G \cap (h, ())) &= h(()|G) \cdot \Big(Pr(q_0 = G \cap h) \cdot p_{GG} \\
&\quad + Pr(q_0 = B \cap h) \cdot p_{BG}\Big) \\
&= 1 \cdot \Big(0.525 \cdot 0.9 + 0.105 \cdot 0.05\Big) \\
&\simeq 0.478
\end{aligned}
$$

$$
\begin{aligned}
Pr(q_1 = B \cap (h, ())) &= h(()|B) \cdot \Big(Pr(q_0 = G \cap h) \cdot p_{GB} \\
&\quad + Pr(q_0 = B \cap h) \cdot p_{BB}\Big) \\
&= 1 \cdot \Big(0.525 \cdot 0.1 + 0.105 \cdot 0.95\Big) \\
&\simeq 0.152
\end{aligned}
$$

$$
\begin{aligned}
Pr(q_2 = G \cap ((h), (), (l, h))) &= h((l, h)|G) \cdot \Big(Pr(q_1 = G \cap (h, ())) \cdot p_{GG} \\
&\quad + Pr(q_1 = B \cap (h, ())) \cdot p_{BG}\Big) \\
&\simeq 0.25 \cdot 0.75 \cdot \Big(0.478 \cdot 0.9 + 0.152 \cdot 0.05\Big) \\
&\simeq 0.188 \cdot 0.438 \\
&\simeq 0.082
\end{aligned}
$$

$$
\begin{aligned}
Pr(q_2 = B \cap ((h), (), (l, h))) &= h((l, h)|B) \cdot \Big(Pr(q_1 = G \cap (h, ())) \cdot p_{GB} \\
&\quad + Pr(q_1 = B \cap (h, ())) \cdot p_{BB}\Big) \\
&\simeq 0.65 \cdot 0.35 \cdot \Big(0.478 \cdot 0.1 + 0.152 \cdot 0.95\Big) \\
&\simeq 0.228 \cdot 0.192 \\
&\simeq 0.044
\end{aligned}
$$

Before we are able to put the outer part together, we need to compute $Pr(\boldsymbol{O})$:

$$
\begin{aligned}
Pr((h), (), (l, h)) &= Pr(q_2 = G \cap ((h), (), (l, h))) + Pr(q_2 = B \cap ((h), (), (l, h))) \\
&\simeq 0.082 + 0.044 \\
&\simeq 0.126
\end{aligned}
$$

Eventually, we can compute the (conditional) probability that $\theta^2 = G$ and $\theta^2 = B$, respectively:

$$\begin{aligned} Pr(q_2 = G|((h),(),(l,h))) &= \frac{Pr(q_2 = G \cap ((h),(),(l,h)))}{Pr((h),(),(l,h))} \\ &\simeq \frac{0.082}{0.126} \\ &\simeq 0.651 \end{aligned}$$

$$\begin{aligned} Pr(q_2 = B|((h),(),(l,h))) &= \frac{Pr(q_2 = B \cap ((h),(),(l,h)))}{Pr((h),(),(l,h))} \\ &\simeq \frac{0.044}{0.126} \\ &\simeq 0.349 \end{aligned}$$

Chapter 8

Conclusion & Discussion

Adverse selection repuation mechanisms are a way to create the right incentives for agents giving feedback. I introduced an extension to the model of Jurca and Faltings [11, 12, 9, 10] that is able to cope with scenarios in which the quality of the product changes over time. To model the change, I used a Markov Process, so that the resulting structure includes hidden Markov models (HMM) as special cases (i. e. those in which there is exactly one observation every time step). As HMMs have found wide application in diverse areas, such as Robotics, Pattern Recognition, Bioinformatics and Finance, it may well be that similar applications of the mechanism can be found in these areas, as well.

Depending on the actual application at hand, there may be a need for further extensions. For example, there are settings in which a single agent makes multiple observations at possibly different points in time. In order to cope with these scenarios, one could probably adapt the technique described by Jurca and Faltings [11] for making the mechanism robust against colluding agents. In other settings, some or all of the probability data may be either unknown or only known within a certain range. This situation may especially arise in the setting described in this work. For most applications, the assumption that the center perfectly knows the transition matrix is too strict. Instead, he may have an approximate estimation of it. Jurca and Faltings [12] provided a formulation of LP 1 as a robust optimization problem. This readily extends to the setting described in this work but it only addresses deviations with regards to the signal posteriors that make up the coefficients of the final LP (our $\boldsymbol{G}$). This approach has the disadvantage that one cannot precisely control the robustness level of the different probabilistic parameters given with the setting. If, for example, the general probabilistic setting is well known (and the center's believe is shared by the agents) but one wants to incorporate noisy perceptions *up to a certain level*, this does not simply translate to a robust optimization problem. Rather, one would have to propagate these noise intervals through the entire sequence of calculations to see how it is reflected in the coefficients of the LP.

Instead of rewriting the LP into a robust optimization problem, one may also try to truthfully elicit the probabilistic parameters. If—as in our setting—the center knows when an agent perceives her signal because he acts as an intermediary, the truthful elicitation of prior beliefs should be possible. I am thinking of a three-step process in which the rating agent first announces her belief about the type distribution, receives her signal and then makes a signal announcement to the center. Here, the agent receives two scores, one depending on how good her prior announcement matches with the reference agent's signal and the other for her actual signal announcement. The tricky part is to scale the first score in such a manner that the expected profit that can be gained by a false prior announcement (the second score is computed with the announced priors) is dominated by the expected profit of an honest prior announcement. To the best of my knowledge this has not been done so far and I believe this is an interesting field for future work.

Similarily, the non-probabilistic parameters given with the setting may be unknown by the center. While it will be difficult to truthfully elicit the external benefits from lying, the costs of participation C_i may be a different thing. Depending on the certainty level the center requires with regards to the product's type, only a limited number of agents are needed to give feedback. If the center's information solely depends on the *number* of agents while he is indifferent about *who* is giving it (as it was assumed so far), it is reasonable to only let the agents with lowest C_i do the rating. In order to achieve this, it may be possible to apply an *ascending reverse auction* with sealed bids. The auction starts at the level of expected payment that is guaranteed by the honesty constraints and the level is raised until the needed (updated) type belief accuracy is reached and the auction ends. The online environment facilitates the application of sealed bid auctions.

Depending on the context, the process of *acquiring* the information may require costly effort. Right after buying a product or service, the agent might not have a precise idea about its quality. Instead, she must invest time to test it thoroughly and compare it to other products. If not factored into the mechanism appropriately, an agent might therefore be better off sparing this investment and simply report some random signal. Zohar and Rosenschein [27] discuss these constraints in a non-reputation setting for a single agent.

As we have seen, the mechanism presented in this work may have multiple equilibria, some of which correspond to higher payoffs for the feedback-giving agents. This leads us to the topic of equilibrium selection. It is not clear and depends on the actual situation which equilibrium is chosen [8, p. 18-23]. As mentioned, MRZ argue that the honest equilibrium is a focal point and will thus be chosen. Unfortunately, this is far from being certain. On the contrary, there is reason to believe that human agents coordinate on a pareto-optimal lying equilbrium, especially if it is composed by symmetric pure strategies as it is mostly the case in the matrices induced by our mechanism. Ironically, through the introduction of payments we may actually receive a worse outcome than in a system without any payments in which some selfish agents (those whose external benefits are low compared to their participation costs) may not engage in the process at all.

However, I believe the presented mechanism can be applied to *software* agents. Reputation issues arise in distributed systems—such as Peer-To-Peer networks[1]—and the standard software is usually provided by the same actor that also develops the protocol. In this scenario, the goal of the mechanism or protocol designer is to construct a network that is robust against modified programs[2] which allow free-riding on the cost of users of the standard software. If the standard software can be programmed on playing honestly, most users will begin using this honest software. Therefore, the objective would be to maximize the fraction of modified software that is needed to make dishonest play a best response. Potentially, an evolutionary framework with a protocol implementing an honest evolutionary stable strategy (or a generalization thereof) may prove useful. In addition, for two reasons the prior elicitation outlined above is especially interesting for distributed networks made up of software agents: First, the prior beliefs may differ vastly depending on the locality of the network node and, second, it is much easier for a software as opposed to a human agent to learn and announce probability distributions.

Except for a single work by Dellarocas [4] that ignores the feedback part, all reputation mechanisms are either combatting *adverse selection* or *moral hazard* but cannot cope with combined scenarios. Real-world applications on the contrary are made up mostly by the latter, i. e. the perceived quality depends on both the selling agent's *ability* and *will*. In a P2P network, for example, the speed of a file transfer depends on both the physical connection between the agents and the sharing parameters at the sender's side (i. e. how much bandwith is open for upload). A similar argument can be made for auction sites of all kind. Creating such mechanisms that are able to cope with both *moral hazard* and *adverse selection* remains an important part of future work.

[1] depending on the type of network one may need a distributed mechanism

[2] such as BitThief in the BitTorrent network or Kazaa Lite in the Kazaa network

Appendix A

Probability Calculations by Matrix Multiplication

$$\boldsymbol{\tau} = \begin{pmatrix} \tau(s_1, s_1) & \dots & \tau(s_1, s_M) \\ \vdots & \ddots & \vdots \\ \tau(s_M, s_1) & \dots & \tau(s_M, s_M) \end{pmatrix}$$

$$\boldsymbol{F} = \begin{pmatrix} f(s_1 | \theta_1) & \dots & f(s_1 | \theta_{|\Theta|}) \\ \vdots & \ddots & \vdots \\ f(s_M | \theta_1) & \dots & f(s_M | \theta_{|\Theta|}) \end{pmatrix}$$

$$\begin{aligned} \boldsymbol{G} &= \begin{pmatrix} g(s_1^{t_1} | s_1^{t_2}) & \dots & g(s_1^{t_1} | s_M^{t_2}) \\ \vdots & \ddots & \vdots \\ g(s_M^{t_1} | s_1^{t_2}) & \dots & g(s_M^{t_1} | s_M^{t_2}) \end{pmatrix} \\ &= \boldsymbol{F} \times \boldsymbol{A}^{\theta^{t_1} \times s^{t_2}} \end{aligned}$$

$$
\begin{aligned}
\boldsymbol{s}^t &= \begin{pmatrix} Pr(s_1^t) \\ \vdots \\ Pr(s_M^t) \end{pmatrix} \\
&= \boldsymbol{F} \times \boldsymbol{\theta}^t
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{A}^{s^{t_2} \times \theta^{t_1}} &= \begin{pmatrix} Pr(s_1^{t_2} \,|\, \theta_1^{t_1}) & \dots & Pr(s_1^{t_2} \,|\, \theta_{|\Theta|}^{t_1}) \\ \vdots & \ddots & \vdots \\ Pr(s_M^{t_2} \,|\, \theta_1^{t_1}) & \dots & Pr(s_M^{t_2} \,|\, \theta_{|\Theta|}^{t_1}) \end{pmatrix} \\
&= \boldsymbol{F} \times \boldsymbol{A}^{\theta^{t_2} \times \theta^{t_1}}
\end{aligned}
$$

For $t_2 \geq t_1$:

$$
\begin{aligned}
\boldsymbol{A}^{\theta^{t_2} \times \theta^{t_1}} &= \begin{pmatrix} Pr(\theta_1^{t_2} \,|\, \theta_1^{t_1}) & \dots & Pr(\theta_1^{t_2} \,|\, \theta_{|\Theta|}^{t_1}) \\ \vdots & \ddots & \vdots \\ Pr(\theta_{|\Theta|}^{t_2} \,|\, \theta_1^{t_1}) & \dots & Pr(\theta_{|\Theta|}^{t_2} \,|\, \theta_{|\Theta|}^{t_1}) \end{pmatrix} \\
&= \boldsymbol{P}^{t_2 - t_1}
\end{aligned}
$$

Appendix B

Random setting

Since every agent but the first is rated against her preceding agent and the types are updated immediately, I set $t_1 = 0$ and $t_2 \geq t_1$ (which is the case for all but the first agent). Furthermore, I consider settings in which every type corresponds to a signal, i. e. $M = |\Theta|$. Default is $M = 5$.

The observation matrix $\boldsymbol{F}$ is set according to the routine also used by Jurca and Faltings [10]:

$$f(s_m|\theta_l) = \begin{cases} 1-\epsilon & m = l \\ \epsilon/(M-1) & m \neq l \end{cases} \tag{B.1}$$

Similarily, $\boldsymbol{P}$ is generated:

$$Pr(\theta_i \rightarrow \theta_j) = \begin{cases} 1-\epsilon & j = i \\ \epsilon/(|\Theta|-1) & j \neq i \end{cases} \tag{B.2}$$

For both $\boldsymbol{F}$ and $\boldsymbol{P}$, I take $\epsilon = 10\%$.

The type vector at $t = 0$ is uniformely distributed, all $\Delta(s_j|s_h)$ ($h \neq j$) are set to 0.15 and C is set to 0.1. I averaged over 1000 randomly generated settings.

Bibliography

[1] George A. Akerlof. The market for 'lemons': Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics*, 84(3):488–500, August 1970.

[2] Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. In *Proceedings of the Uncertainty in Artificial Intelligence Conference (UAI), Edmonton, Canada.*, 2002.

[3] Roger M. Cooke. *Experts in Uncertainty*. Oxford University Press, 1991.

[4] Chrysanthos Dellarocas. Efficiency through feedback-contingent fees and rewards in auction marketplaces with adverse selection and moral hazard. In *Proceedings of the ACM Conference on Electronic Commerce (EC 03)*, pages 11–18, 2003.

[5] Chrysanthos Dellarocas. Reputation mechanisms. In T. Hendershott, editor, *Handbook on Information Systems and Economics*. Elsevier Publishing, 2006.

[6] Chrysanthos Dellarocas and Charles A. Wood. The sound of silence in online feedback: Estimating trading risks in the presence of reporting bias. *Management Science*, 54(3):460–476, March 2008.

[7] Warren J. Ewans and Gregory R. Grant. *Statistical Methods in Bioinformatics - An Introduction*, volume 2. Springer-Verlag, 2005.

[8] Drew Fudenberg and Jean Tirole. *Game Theory*. MIT Press, 1991.

[9] Radu Jurca and Boi Faltings. Enforcing truthful strategies in incentive compatible reputation mechanisms. In *Internet and Network Economics*, volume 3828 of *LNCS*, pages 268 – 277. Springer-Verlag, 2005.

[10] Radu Jurca and Boi Faltings. Minimum payments that reward honest reputation feedback. In *Proceedings of the ACM Conference on Electronic Commerce (EC 06)*, pages 190–199, 2006.

[11] Radu Jurca and Boi Faltings. Collusion resistant, incentive compatible feedback payments. In *Proceedings of the ACM Conference on Electronic Commerce (EC 07)*, pages 200–209, 2007.

[12] Radu Jurca and Boi Faltings. Robust incentive-compatible feedback payments. In M. Fasli and O. Shehory, editors, *Trust, Reputation and Security: Theories and Practice*, volume LNAI 4452, pages 204–218. Springer-Verlag, 2007.

[13] Ulrich Krengel. *Einführung in die Wahrscheinlichkeitstheorie und Statistik*, volume 8. Vieweg, 2005.

[14] Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, June 1995.

[15] Nolan Miller, Paul Resnick, and Richard Zeckhauser. Eliciting Informative Feedback: The Peer-Prediction Method. *Management Science*, 51(9):1359–1373, 2005.

[16] Michael Mitzenmacher and Eli Upfal. *Probability and Computing*. Cambridge University Press, 2005.

[17] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[18] Theo Offerman, Joep Sonnemans, Gijs van de Kuilen, and Peter P. Wakker. A truth-serum for non-bayesians: Correcting proper scoring rules for risk attitudes. Technical report, Econometric Institute, Erasmus University Rotterdam, 2007.

[19] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[20] David Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency*. PhD thesis, Department of Computer and Information Science, University of Pennsylvania, 2001.

[21] Paul Resnick and Richard Zeckhauser. Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system. In *The Economics of the Internet and E-commerce*, volume 11, pages 127–157. JAI, 2002.

[22] Paul Resnick, Richard Zeckhauser, John Swanson, and Kate Lockwood. The value of reputation on ebay: A controlled experiment. *Experimental Economics*, 9(2):79–101, June 2006.

[23] Mark S. Roulston and Leonard A. Smith. Evaluating probabilistic forecasts using information theory. *Monthly Weather Review*, 130(6):1653–1660, 2002.

[24] Tuomas Sandholm. Automated mechanism design: A new application area for search algorithms. In *Proceedings of the International Conference on Principles and Practice of Constraint Programming (CP 03)*, 2003.

[25] Gilbert Strang. *Linear Algebra*. Springer Verlag, 2003.

[26] Robert Winkler, Javier Mu noz, José Cervera, José Bernardo, Gail Blattenberger, Joseph Kadane, Dennis Lindley, Allan Murphy, Robert Oliver, and David Ríos-Insua. Scoring rules and the evaluation of probabilities. *TEST: An Official Journal of the Spanish Society of Statistics and Operations Research*, 5:1–60, 1996.

[27] Aviv Zohar and Jeffrey S. Rosenschein. Robust mechanisms for information elicitation. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems (AAMAS 06)*, pages 1202–1204, New York, NY, USA, 2006. ACM.