ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-778

Crowdfunding Public Projects with Provision Point: A Prediction Market Approach

Praphul Chandra[1] and **Sujit Gujar**[2] and **Y. Narahari**[3]

Abstract. Crowdfunding is emerging as a popular means to generate funding from citizens for public projects. This is popularly known as civic crowdfunding. In this paper, we focus on crowdfunding public projects with provision point: these are projects in which contributions must reach a predetermined threshold in order for the project to be provisioned. On web based civic crowdfunding platforms, the success of crowdfunding public projects has been somewhat mixed. In this paper, our objective is to design a mechanism that improves the success of crowdfunding public projects. In particular, we propose a class of mechanisms for crowdfunding platforms with sequentially arriving agents. This class of mechanisms induces an extensive form game for agents arriving on the platform and we show that the game has a non-empty set of sub-game perfect equilibria at which the project is fully funded. We call this new class of mechanisms *Provision Point Mechanism with Securities* (PPS). The novelty of PPS lies in the use of a prediction market to incentivize agents to contribute in proportion to their true value for the project and to contribute as soon as they arrive at the crowdfunding platform. Different variations of PPS are possible depending on the underlying prediction market. In this paper, we use a cost function (or equivalently, scoring rule) based prediction market; in fact, we specify the requirements that a cost function should satisfy to be used in PPS. We study and compare two specific instances of PPS: (1) Logarithmic Market Scoring Rule based and (2) Quadratic Scoring Rule based. We also discuss the considerations that should guide the choice of the cost function when deploying our mechanism on crowdfunding platforms.

1 INTRODUCTION

Civic crowdfunding platforms like Spacehive [1], Citizinvestor [11], Neighbourly [20] etc., aim to generate funding for public and community projects from citizens. The success of these platforms has been mixed. For example, in the United Kingdom, Spacehive has generated £4.4 million for public projects from citizen contributions across 68 cities with a 44% *success rate* (the fraction of posted projects that are fully funded)[1]. Thus, less than half the number of projects posted meet their funding targets. In this paper, our objective is to design a mechanism that can markedly improve the success rate of crowdfunding public projects. A typical process that is followed in crowdfunding of public projects is as follows:

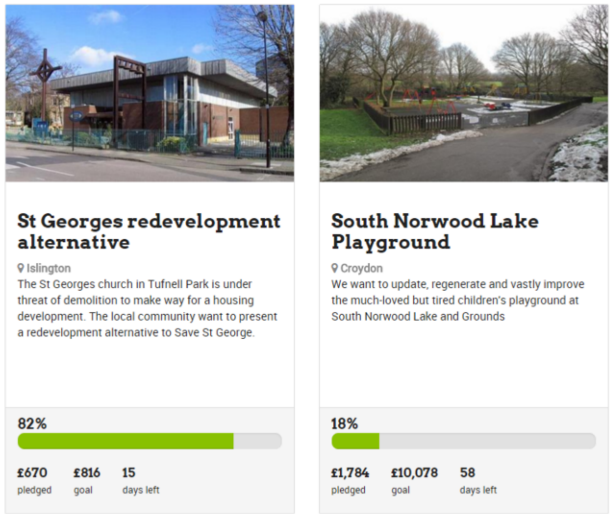

Figure 1. Public projects listed on a crowdfunding platform

1. *Requester posts public project*: A requester, seeking crowdfunding for a public project, posts a proposal. The proposal specifies a target amount of funds to be raised for the project to be provisioned: the target amount is thus also known as the provision point. The requester also specifies a deadline by which the funds need to be raised.
2. *Agents arrive*: Agents arrive over time to view the project and observe (i) the target amount, (ii) the amount pending to be funded, and (iii) the deadline. Figure 1 shows two different ongoing public projects from a crowdfunding platform [1] as they appear to an agent arriving at the platform.
3. *Agents contribute*: Each arriving agent may contribute a certain amount towards funding the project.
4. *Requester provisions or refunds*: If the funding target is achieved by the deadline, the requester provisions the project; otherwise, the contributions of all agents are refunded.

Two features in this process are notable: (i) Crowdfunding relies on voluntary contributions and hence neither coercion nor punishment is an option. (ii) Since contributions arrive over a period of time, an agent is able to observe the contributions of the agents who have contributed so far.

Relying on private contributions to fund a public project is not a new phenomenon and has been studied in the literature extensively [4, 5, 25, 26, 12, 6, 23, 27, 16]. The key challenge in relying on private contributions to fund a public project is the free-riding problem: since public projects are non-excludable and non-rival, agents have

[1] Indian Institute of Science, Bangalore, India, email: praphulcs@gmail.com
[2] International Institute of Information Technology, Hyderabad, India, email:sujit.gujar@iiit.ac.in (Work done while at IISc.)
[3] Indian Institute of Science, Bangalore, India, email: hari@csa.iisc.ernet.in

an incentive to free-ride on the contributions of others. One approach to solve the free riding problem is using *assurance contracts* which allow agents to commit contributions conditional on sufficient contributions from others [5, 25]. The term *dominant assurance contracts* is used to refer to contracts which can ensure that the project gets funded at equilibrium [26]. Several mechanisms have been proposed which implement assurance contracts.

Provision Point Mechanism (PPM) [4] invites contributions for a target amount. If the target is met, the project is provisioned; otherwise, agents' contributions are refunded. Provision Point mechanism with Refund bonus (PPR), proposed by Zubrickas *et. al.* [27], invites contributions for a target amount. If the target is met, the project is provisioned; otherwise, agents' contributions are refunded *and* agents who volunteered to contribute are paid an additional *refund bonus* which depends on the quantum of the agent's contribution.

Neither of the above two mechanisms takes into account the sequential nature of agent contributions in crowdfunding platforms. The mechanisms also do not handle the fact that contributions, once made, are common knowledge. Applying these mechanisms in a sequential setting could hamper their success in crowdfunding of public projects (Section 3.2). This motivates the need for our proposed mechanism, in which, we retain the idea of a refund bonus[4] from [27] but the way we compute the refund bonus makes the mechanism more attractive. Moreover our mechanism explicitly takes into account the sequential nature of arrivals of agent contributions. The novelty in our mechanism is in using a prediction market based approach for computing the refund bonuses. We award contingent securities to the agents who contribute to the public project: if the project is not funded, an agent is paid a unit amount for each unit of security held by the agent. The number of securities awarded to an agent is based on the quantum of the agent's contribution and the time at which the agent makes the contribution. As the securities are allotted only for the outcome that the project is not funded and not for the outcome that the project is funded, the prediction market under consideration is classified as *complex prediction market* [2].

1.1 Contributions and Outline

The following are the main contributions of this paper.

- We propose a class of mechanisms, named *Provision Point Mechanism with Securities* (PPS) for crowdfunding public projects. PPS induces an extensive form game and we show that the game has a non-empty set of sub-game perfect equilibria at which the project is fully funded (Theorem 3).
- PPS solves the free-riding problem when public projects are provisioned using private contributions since agents are incentivized to contribute *in proportion* to their true value for the project and to contribute *as soon as they arrive* at the crowdfunding platform.
- PPS uses a complex prediction market [2]. Different versions of PPS are possible depending on the underlying prediction market and the cost function. We study and compare two specific instances of PPS: (1) Logarithmic Market Scoring Rule (LMSR) [14, 15] based and (2) Quadratic Scoring Rule (QSR) based.

The rest of the paper is organized as follows. In Section 2, we summarize the notation we use and present some preliminaries. In Section 3, we position our work in relation to the existing literature. In Section 4, we review complex prediction markets and propose a new class of mechanisms, PPS, for crowdfunding a public project and show the existence of a non-empty set of sub-game perfect equilibria where the project gets fully funded. In Section 5, we compare the performance of PPS mechanism with two popular cost functions and discuss the impact of the cost function in PPS. We conclude in Section 6 with a summary.

[4] Even though PPS relies on a *sponsor* to offer a refund bonus while the funds are being collected, the bonus is not paid out at equilibrium.

2 PRELIMINARIES

We focus on crowdfunding projects which involve private provisioning of a public project without coercion and with agents arriving over time. Table 1 lists the key notation used in this work. Similar to previous work [4, 27], we assume that agents have quasi-linear utility (ASSUMPTION-1) and apart from knowing the history of contributions, agents do not have any information regarding whether the project will get funded or not (ASSUMPTION-2). We model the following sequence of events. At $t = 0$, the requester posts a proposal for funding a public project. This includes the target amount of funds h^0 (the provision point) and a deadline T till which agents may contribute to the project. h^t refers to the target amount that remains to be collected at time t: h^0, T, and h^t are common knowledge. Agent $i \in \{1, 2, \dots, n\}$ arrives at time $a_i \in [0, T]$ and observes the funds that have been collected so far $(h^0 - h^{a_i})$. The value that an agent derives from the public project getting provisioned (θ_i) is his private information. Agent i may decide to contribute funds $x_i \in [0, h^{a_i}]$ to the project at any time $t_i \in [a_i, T]$. We assume that agents contribute only once to the project (ASSUMPTION-3). This assumption is reasonable in civic crowdfunding scenarios where agents typically visit the project website once and contribute if the project has value to them. From an analysis view point, the mechanism we design ensures that agents have no advantage in delaying or splitting up their contributions. We leave it for future work to study effect of spiteful contributions [8].

Symbol	Definition
T	Time at which fund collection concludes
t	Epoch of time in the interval $[0, T]$
h^t	Amount that remains to be funded at t;
h^0	Target amount (provision point)
$i \in \{0, 1, \dots, n\}$	Agent id; $i = 0$ refers to the requester
$\theta_i \in \mathbb{R}_+$	Agent i's value for the project
$x_i \in \mathbb{R}_+$	Agent i's contribution to the project
$a_i \in [0, T]$	Time at which agent i arrives at the platform
$t_i \in [a_i, T]$	Time at which agent i contributes to the project
$\psi_i = (x_i, t_i)$	Strategy of agent i
$\vartheta \in \mathbb{R}_+$	Net value for the project
$\chi \in \mathbb{R}_+$	Net contribution for the project

Table 1. Key notation

The strategy of agent i is $\psi_i = (x_i, t_i)$. $\mathbf{x} = (x_1, \dots, x_n)$ refers to the vector of agent contributions and $\psi = (\psi_1, \dots, \psi_n)$ denotes the strategy profile of agents. We use the subscript $-i$ to represent vectors without agent i; so, for example, x_{-i} refers to the vector of contributions of all agents except i. The net value for the project among the agents is $\vartheta = \sum_{i=1}^{n} \theta_i$ and the net contribution is $\chi = \sum_{i=1}^{n} x_i$. The utility derived by agent i with value θ_i for the project, when agents use strategy profile ψ is $u_i(\psi; \theta_i)$.

2.1 Important Definitions

We seek to design mechanisms in a sequential setting such that a public project gets funded at equilibrium. Such mechanisms induce a game among the agents $\{1, 2, \ldots, n\}$. With ψ_is being agents' strategies and u_is as their utilities, we define Pure Strategy Nash Equilibrium (PSNE) and Sub-Game Perfect Equilibrium (SGPE).

Definition: (Pure Strategy Nash Equilibrium) A strategy profile $\psi^* = (\psi_1^*, \ldots, \psi_n^*)$ is said to be a Pure Strategy Nash Equilibrium (PSNE) if $\forall i, \forall \theta_i$

$$u_i(\psi_i^*, \psi_{-i}^*; \theta_i) \quad \geq \quad u_i(\tilde{\psi}_i, \psi_{-i}^*; \theta_i) \quad \forall \tilde{\psi}_i.$$

Let H^t be the history of the game till time t, that contains the agents' arrivals and their contributions, then we define:

Definition: (Sub-game Perfect Equilibrium) A strategy profile $\psi^* = (\psi_1^*, \ldots, \psi_n^*)$ is said to be a sub-game perfect equilibrium if $\forall i, \forall \theta_i$

$$u_i(\psi_i^*, \psi_{-i|H^{a_i}}^*; \theta_i) \quad \geq \quad u_i(\tilde{\psi}_i, \psi_{-i|H^{a_i}}^*; \theta_i) \quad \forall \tilde{\psi}_i, \forall H^t$$

Here $\psi_{-i|H^{a_i}}^*$ indicates that the agents who arrive after a_i follow the strategy specified in ψ_{-i}^*.

3 RELATED WORK

Our work is related to the literature on provisioning of *public goods*. In this paper, we use the term *public project* instead, since it is more suitable for crowdfunding platforms. The literature deals with two kinds of public projects. For *discrete* public projects, a predetermined target amount must be collected for the project to be provisioned. For *continuous* public projects, the *extent* of project provisioned increases monotonically with net contributions, up to a threshold.

For *continuous*, public projects, one of the simplest mechanisms is the Voluntary Contribution Mechanism (VCM): agents voluntarily contribute and the extent of the public project provisioned corresponds to the aggregate funds collected. VCM induces a simultaneous move game which has multiple equilibria. Many of these equlibiria lead to an under-provisioning of the public project, a result which has been verified empirically [17]. Morgan [19] studies the use of state lotteries for funding continuous public projects. Voluntary contributions are incentivized by offering an opportunity to win a fixed prize and an agent's contribution towards public project also determines the likelihood of his winning the prize. This game has a unique equilibrium which provisions a higher level of the public project than VCM.

In this paper, our focus is on *discrete* public projects (projects with a provision point) which are predominant on crowdfunding platforms. As discussed in Section 1, our work is motivated by the need to non-trivially extend the work of [4] and [27] to the realistic setting where agent contributions arrive *sequentially*. Marx and Matthews[18] consider a sequential setting where agents make repeated contributions to a project, taking turns in a round-robin fashion. They prove the existence of a Nash equilibrium where each agent contributes if and only if all the past agents have contributed their equilibrium contributions. Thus, it is not a sub-game perfect equilibrium. Our work differs from this in that, there is neither a pre-fixed order of contributions nor do agents contribute repeatedly and we look for sub-game perfect equilibria.

3.1 Provision Point Mechanism (PPM)

PPM [4] for discrete public projects collects voluntary contributions. The project is provisioned if the funding target is achieved. If the funding target is not achieved, the contributions are refunded. Let $\mathcal{I}_X$ be an indicator random variable which takes the value 1 if X is true and 0 otherwise. Thus, for PPM, the project gets funded only if $\chi \geq h^0$, and agent i's ($i > 0$) utility[5] in PPM is:

$$u_i(\mathbf{x}; \theta_i) \quad = \quad \mathcal{I}_{\chi \geq h^0} \times (\theta_i - x_i) + \mathcal{I}_{\chi < h^0} \times 0 \qquad (1)$$

In PPM, an agent's utility consists of a *funded utility* $(\theta_i - x_i)$, which is the agent's utility if the project is provisioned and an *unfunded utility* (zero) which is the agent's utility if the project is not provisioned. PPM has been shown to have multiple equilibria, many of which are inefficient [4]: a result which has been verified empirically too [17].

3.2 Provision Point Mechanism with Refund bonus (PPR)

PPR [27] for discrete public projects collects voluntary contributions. The project is provisioned if the funding target is achieved. If the funding target is not achieved, the contributions are refunded and an additional refund bonus is paid to agents who volunteered to contribute, in proportion to their contribution. The refund bonus is $\frac{x_i}{\chi} B \ \forall i$ where $B > 0$ is the refund budget set aside by the requester at the beginning and is common knowledge among all agents. Thus, for PPR too, the project is provisioned only if $\chi \geq h^0$ and agent i's utility in PPR is:

$$u_i(\mathbf{x}; \theta_i) \quad = \quad \mathcal{I}_{\chi \geq h^0} \times (\theta_i - x_i) + \mathcal{I}_{\chi < h^0} \times \left(\frac{x_i}{\chi} B\right) \quad (2)$$

In PPR, an agent's utility consists of a *funded utility* $(\theta_i - x_i)$, which is the agent's utility if the project is provisioned and a strictly positive *unfunded utility* ($\frac{x_i}{\chi} B > 0$), which is the agent's utility if the project is not provisioned. The set of Pure Strategy Nash equilibria with PPR are characterized as follows:

Theorem 1 *[27] Let $\vartheta > h^0$ and $B > 0$. In PPR, the set of PSNE are $\{(x_i^*) : x_i^* \leq \frac{h^0}{B+h^0} \theta_i \forall i; \chi = h^0\}$ if $B \leq \vartheta - h^0$. Otherwise the set of PSNE is empty.*

Limitations of PPR

PPR considers a setup where agents decide their contributions *simultaneously* without knowledge of contributions made by the other agents. The game induced is thus a simultaneous move game. When applied in a sequential (discrete time) setting where agents can contribute over time and can observe previous contributions, agent strategies consist of the contribution amount and the interval in which they contribute.

Proposition: Let $\vartheta > h^0$ and $B > 0$. If $B \leq \vartheta - h^0$, the set $\{(x_i^*, T) : x_i^* \leq \frac{h^0}{B+h^0} \theta_i \forall i; \chi = h^0\}$ constitutes Pure Strategy Nash equilibria of PPR in the sequential setting.

[5] As the strategy space in PPM consists only of contribution to be made, we drop ψ here.

Proof: In PPR, since the refund bonus does not depend on the time when the contribution is made, no agent has an incentive to invest earlier than the deadline, if all other agents do the same. Effectively, PPR in sequential setting collapses to a one shot simultaneous move game at $t = T$ where x_i^*s are given by Theorem (1). Thus, (x_i^*, T) constitute a set of PSNE of PPR in sequential setting. □

The implication of the above proposition is that all the agents may delay their contribution as close to the deadline as possible in sequential settings and wait to free-ride till the end. This is undesirable since such temporal strategies can lead to the equilibrium not being achieved in practice. This shortcoming of PPR in sequential settings is because early contributions do not receive any advantage.

We seek to design crowdfunding mechanisms by explicitly capturing and taking advantage of the fact that on web based crowdfunding platforms, contributions are sequential rather than simultaneous. The key intuition in our approach is that by giving participants a payoff structure which refunds them more generously if they contribute earlier (in the event the project is not funded), participants have an incentive to contribute early. This overcomes the serious limitation of PPR. We achieve our objective using a novel prediction market approach.

4 OUR APPROACH: PREDICTION MARKET FOR CROWFUNDING MECHANISMS

We incorporate ideas from the literature on prediction markets with the key idea being that contributors actually buy contingent securities which each pay a unit amount if the project is not funded. As these securities are purchased, the price increases, thereby incentivizing the participants to contribute earlier rather than later. Our mechanism achieves an equilibrium at which the project is funded and thus the refund bonus is not paid out at equilibrium. Since our approach leverages prediction markets, we briefly explain important concepts from prediction market literature that are relevant to our crowdfunding mechanism design approach.

4.1 Cost function based Prediction Markets

A Prediction Market seeks to predict the outcome of an event in future. Let Ω be the set of mutually exclusive and exhaustive outcomes of the event. For example, in a prediction market designed to predict the outcome of a political election among two candidates, we would be interested in an outcome set $\Omega = \{\omega_A, \omega_B\}$ where ω_A is the outcome that candidate A wins the election and ω_B is the outcome that candidate B wins the election. Since $|\Omega| = 2$, we refer to this as a *binary* outcome event. A prediction market incentivizes agents to express their belief about the outcome of an event. One approach to realize a prediction market is by associating securities (Arrow-Debreu contracts [3]) with the outcomes of the event. A security associated with outcome ω_j pays a unit amount if ω_j is realized and zero otherwise. An agent with a belief different from the market belief can buy (or sell) securities to modify the market belief. An automated market maker is a software agent which automates pricing and order execution of such securities. An automated market maker can be realized using a cost function $C : \mathbb{R}^{|\Omega|} \to \mathbb{R}$ which is a potential function specifying the amount of money wagered in the market as a function of the number of securities that haven been issued by the market for each outcome. In a market with a binary outcome event, $C : \mathbb{R}^2 \to \mathbb{R}$ is a function of the vector of outstanding securities, $\mathbf{q} \in \mathbb{R}^2$. Several authors [22, 21, 2, 10, 6] have studied conditions that a cost function must satisfy to be used in prediction markets.

Symbol	Definition
$\Omega = \{\omega_j\}_{j\in\{1,\ldots,\|\Omega\|\}}$	Set of possible outcomes of the event
π_{ω_j}	Payoff vector if outcome ω_j is realized
p_{ω_j}	Price of an infinitesimally small amount of security associated with outcome ω_j
$\mathbf{q} = \{q_{\omega_j}\}_{\{1,\ldots,\|\Omega\|\}}$	Vector of securities issued by the market maker
$C : \mathbb{R}^{\|\Omega\|} \to \mathbb{R}$	Cost function used in the prediction market
$\mathbf{r}$	Bundle securities purchased by an agent
$\texttt{Cost}(\mathbf{r}\|\mathbf{q})$	Cost of purchasing a bundle of $\mathbf{r}$ securities when $\mathbf{q}$ securities are outstanding

Table 2. Important terms for prediction market

4.1.1 *Conditions on Cost Function*

- CONDITION-1 (PATH INDEPENDENCE) This condition requires that the cost of acquiring a bundle of $\mathbf{r}$ securities must be the same regardless of how an agent splits up the purchase. This condition implies that in a prediction market, prices can be represented by a cost function such that the cost of purchasing a bundle of $\mathbf{r}$ securities is $\texttt{Cost}(\mathbf{r}|\mathbf{q}) = C(\mathbf{q}+\mathbf{r}) - C(\mathbf{q})$.
- CONDITION-2 (CONTINUOUS AND DIFFERENTIABLE) This condition requires that the gradient of the cost function ($\nabla C(\mathbf{q})$) is well defined everywhere so that it can be treated as a vector of instantaneous prices for securities associated with each outcome. Further $p_{\omega_j} = \partial C(\mathbf{q})/\partial(q_{\omega_j}) \geq 0 \quad \forall \omega_j \in \Omega$ represents the price per security of an infinitesimally small amount of security associated with outcome ω_j.:
- CONDITION-3 (INFORMATION INCORPORATION) This condition requires that a purchase of a bundle of $\mathbf{r}$ securities should never lower the price of $\mathbf{r}$, that is, for any $\mathbf{q}$ and $\mathbf{r} \in \mathbb{R}^{|\Omega|}$ $C(\mathbf{q}+2\mathbf{r}) - C(\mathbf{q}+\mathbf{r}) \geq C(\mathbf{q}+\mathbf{r}) - C(\mathbf{q})$. This condition is required to ensure that participating in the market is incentive compatible for a myopic agent. Further, this condition implies that the cost function used in prediction markets must be convex.
- CONDITION-4 (NO ARBITRAGE) This condition requires that it is never possible for an agent to purchase a bundle of securities $\mathbf{r}$ and receive a positive payoff *regardless* of the outcome. For all $\mathbf{q}$ and $\mathbf{r} \in \mathbb{R}^{|\Omega|}$, $\exists \omega_j \in \Omega$ such that $C(\mathbf{q}+\mathbf{r}) - C(\mathbf{q}) > \mathbf{r} \cdot \pi_{\omega_j}$
- CONDITION-5 (EXPRESSIVENESS) This condition requires that any agent can set the market belief to reflect his belief about the expected outcome. Let Δ_n be the n dimensional probability simplex, then $\forall \mathbf{p} \in \Delta_{|\Omega|}, \exists \mathbf{q} \in \mathbb{R}^{|\Omega|}$ s.t. $\nabla C(\mathbf{q}) = \mathbb{E}_{\omega\sim\mathbf{p}}[\pi(\omega)]$.
- CONDITION-6 (BOUNDED LOSS) This condition requires that an automated market maker using a cost function can only lose a finite amount regardless of the transactions undertaken by the agents, that is, $\sup_{\mathbf{q}}[\max_{\omega_j}(q_{\omega_j}) - C(\mathbf{q})] < \infty$. If the market maker initializes the market with $\mathbf{q} = (0, 0)$, then the worst case loss for the market maker is $\sup_{\mathbf{q}}[\max_{\omega_j}(q_{\omega_j}) - (C(\mathbf{q}) - C(\mathbf{0}))]$.

With a binary outcome event, two popular cost functions that satisfy these conditions are [7]:

$$C_{LMSR}(\mathbf{q}) = b\ln(\exp(q_{\omega_0}/b) + \exp(q_{\omega_1}/b)) \quad (3)$$

$$C_{QSR}(\mathbf{q}) = \frac{q_{\omega_0} + q_{\omega_1}}{2} + \frac{q_{\omega_0}^2 + q_{\omega_1}^2}{4b} - \frac{(q_{\omega_0} + q_{\omega_1})^2}{8b} - \frac{b}{2} \quad (4)$$

where b is a parameter that controls how fast prices change.

4.2 Proposed Mechanism: Provision Point Mechanism with Securities (PPS)

We now introduce a new class of mechanisms which explicitly takes into account the sequential nature of contributions. Similar to PPM and PPR, in this new class of mechanism too, the project gets provisioned only if the net contributions reach the provision point ($\chi \geq h^0$). However, the refund bonus of a contributor is determined using securities from the complex cost based prediction market defined in Section 4.2.1. We refer to our mechanism as Provision Point Mechanism with Securities (PPS).

In PPS, we create a prediction market by associating securities with the binary outcome of the public project getting funded or not. We consider a binary outcome ($\Omega = \{\omega_0, \omega_1\}$) event where ω_0 refers to the (negative) outcome that the project is not funded by the deadline and ω_1 refers to the (positive) outcome that the project is funded by the deadline. The key intuition in PPS is to incentivize agents to contribute to public projects by treating every contribution towards the public project as simultaneously an investment in purchasing securities associated with the negative outcome (project not getting funded). We treat every contribution $x_i > 0$ at t_i towards the public project as simultaneously an investment in purchasing $r_i^{t_i} > x_i > 0$ securities associated with the negative outcome ω_0: each of these securities pays out a unit amount if the project is not fully funded and zero otherwise. Thus, agent i's utility who contributes x_i at t_i is:

$$u_i(\psi; \theta_i) \quad = \quad \mathcal{I}_{\chi \geq h^0} \times (\theta_i - x_i) + \mathcal{I}_{\chi < h^0} \times (r_i^{t_i} - x_i) \quad (5)$$

Equation (5) consists of two terms: the *funded utility* is $(\theta_i - x_i)$ and the *unfunded utility* is $(r_i^{t_i} - x_i)$. The funded utility is a monotonically decreasing function of x_i and is independent of the time the contribution is made (t_i) and the history of the game till time t_i (H^{t_i}).

The unfunded utility $(r_i^{t_i} - x_i)$ depends on $r_i^{t_i}$ which in turn depends on (i) the quantum of the contribution (x_i)[6], (ii) the timing of the contribution (t_i) and (iii) the history of past contributions H^{t_i} via the total number of outstanding securities (q^{t_i}). CONDITION-3 ensures that $r_i^{t_i}$ is a monotonically decreasing function of q^{t_i}. This means that for a given contribution, the number of securities awarded to an agent cannot increase with time: intuitively, this is the reason why agents are incentivized to contribute early. It turns out that the cost function needed to determine $r_i^{t_i}$ in our crowdfunding mechanism needs to be a *complex* cost function [2].

4.2.1 Complex Prediction Markets

Abernethy *et. al.* [2] introduce the distinction between *complete* and *complex* cost function based prediction markets. In a complete cost function based market, the market maker offers a security corresponding to *each* potential outcome and *each* of these securities pays a unit amount if the associated outcome is realized. In a complex cost function based market, the market maker may offer $K < |\Omega|$ securities and/or a security may not necessarily pay a unit amount when an outcome $\omega_j \in \Omega$ is realized.

A key result from [2] imposes additional constraints on cost function to be used in complex prediction markets. Let $\pi_\Omega = \{\pi_{\omega_j} | \omega_j \in \Omega\}$ and let $\mathcal{H}(\pi_\Omega)$ be the convex hull of π_Ω. Then,

Theorem 2 *[2] If $\mathcal{H}(\pi_\Omega)$ is closed, then under* CONDITIONS2-5*, C must be convex with $\{\nabla C(\mathbf{q}) : \mathbf{q} \in \mathbb{R}^K\} = \mathcal{H}(\pi_\Omega)$.*

[6] Even though $r_i^{t_i}$ depends on x_i, we use the notation $r_i^{t_i}$ for simplicity.

The following corollary, which can be derived from Theorem 2 and its proof in [2], will be useful in determining the price of securities awarded to agents who contribute to the public project.

Corollary: If $\mathcal{H}(\pi_\Omega)$ is closed, then under CONDITIONS2-4, C must be convex with $\{\nabla C(\mathbf{q}) : \mathbf{q} \in \mathbb{R}^K\} \subseteq \mathcal{H}(\pi_\Omega)$.

4.2.2 A Complex Prediction Market for Crowdfunding

Previous work has used complex cost function based markets in scenarios where the outcome space ($|\Omega|$) is very large [2, 9, 13]. Our use of complex prediction markets in the context of crowdfunding (binary outcome event) is motivated not by a large outcome space but with the explicit objective of limiting the expressiveness of agents.

For a binary outcome event, a *complete* cost function based prediction market offers $K = |\Omega| = 2$ securities where the securities associated with the negative outcome (q_{ω_0}) pay a unit amount of the project is not funded ($\pi_{\omega_0} = (1, 0)$) and securities associated with the positive outcome (q_{ω_1}) pay a unit amount of the project is funded ($\pi_{\omega_1} = (0, 1)$). Thus, $\pi_\Omega = \{(1, 0), (0, 1)\}$.

For a binary outcome event, we propose a *complex* cost function based prediction market that offers $K = |\Omega| = 2$ securities where the securities associated with the negative outcome (q_{ω_0}) pay a unit amount if the project is not funded ($\pi_{\omega_0} = (1, 0)$) but securities associated with the positive outcome (q_{ω_1}) *never* payout ($\pi_{\omega_1} = (0, 0)$). Thus, $\pi_\Omega = \{(1, 0), (0, 0)\}$ and $\mathcal{H}(\pi_\Omega) = [0, 1]$. Furthermore, agents are not allowed to sell securities. Thus, the design of our complex prediction market for crowdfunding has the following implications.

1. FIXED POSITIVE OUTCOME SECURITIES Since the payoff associated with the positive outcome are zero ($\pi_{\omega_1} = (0, 0)$) and since CONDITION-2 requires that the price of the positive security is non-negative ($\partial C(\mathbf{q})/\partial q_{\omega_1} \geq 0$), no agent will purchase securities associated with the positive outcome. For the rest of the paper we will assume that the market is initialized with $\mathbf{q} = (0, 0)$[7]. Thus, for the duration of the market $q_{\omega_1} = 0$ and $\mathbf{q} = (q_{\omega_0}, 0)$.
2. LIMITED EXPRESSIVENESS Since agents are not allowed to sell securities and since they can purchase securities associated with the negative outcome only, it follows that agents may not be able to express their true beliefs about the event and thus our complex market violates CONDITION-5.
3. LOW RANK PRICE SPACE Using the previous implication and Corollary from Section 4.2.1, we have that $\partial C(\mathbf{q})/\partial q_{\omega_0} \in [0, 1]$ and if the market is initialized with $\mathbf{q} = (0, 0)$, then $\partial C(\mathbf{q})/\partial q_{\omega_0} \in [0.5, 1]$.

To emphasize that securities related with the positive outcome are fixed at initialization and not traded for the duration of the market ($[0, T]$), we will refer to the cost function used in this proposed prediction market as C_0. Such a cost function can be obtained by taking any cost function which satisfies CONDITIONS 1-4,6 and setting $q_{\omega_1}^t = 0 \quad \forall t \in [0, T]$. Thus, $C_0 : \mathbb{R} \rightarrow \mathbb{R}$. For the rest of the paper, since we will be using prediction market involving *only* negative outcome securities, we will use the following simplified notation:

$$\begin{aligned} q &\equiv q_{\omega_0} \\ \pi &\equiv \pi_{\omega_0} \\ p &\equiv p_{\omega_0} = \frac{\partial C_0(q)}{\partial q_{\omega_0}} \end{aligned}$$

[7] If the market is initialized with a different number of positive securities, z, then $\mathbf{q} = (q_{\omega_0}, z)$. Section 5.4 discusses this scenario.

Proposition: Let C be a cost function which satisfies CONDITION-2 and C_0 be the corresponding cost function obtained by setting $q^t_{\omega_1} = 0 \quad \forall t \in [0, T]$, then C_0 is invertible.

Proof: Since $q^t_{\omega_1} = 0 \quad \forall t \in [0, T]$, $C_0 : \mathbb{R} \to \mathbb{R}$ is a one-to-one function of a single variable. By CONDITION-2, C_0 is continuous and differentiable. Thus, by the inverse function theorem [24], the inverse of C_0 exists. □

We use the notation C_0^{-1} to refer to the inverse of the function C_0. In PPS, the cost of purchasing $r_i^{t_i}$ securities at t_i when q^{t_i} securities are outstanding is:

$$\texttt{Cost}(r_i^{t_i}|q^{t_i}) \quad = \quad C_0(q^{t_i} + r_i^{t_i}) - C_0(q^{t_i})$$

Thus, an agent who contributes x_i at t_i receives $r_i^{t_i}$ securities where:

$$\begin{aligned} x_i &= C_0(q^{t_i} + r_i^{t_i}) - C_0(q^{t_i}) \\ r_i^{t_i} &= C_0^{-1}(x_i + C_0(q^{t_i})) - q^{t_i} \end{aligned} \tag{6}$$

Since the cost function must be path independent (CONDITION-1), the number of securities issued by a single contribution of h^0 is the same as the number of securities allocated if the contribution is split into any number of smaller contributions ($\sum_i x_i = h^0$). Hence, in PPS the total number of securities that will be issued for crowdfunding a project with a target amount h^0 is:

$$\sum_i r_i^{t_i} = C_0^{-1}(h^0 + C_0(0)) \tag{7}$$

4.2.3 *Equilibrium Analysis of PPS*

We now specify an additional condition that a cost function needs to satisfy to be used in PPS.

- CONDITION-7 (SUFFICIENT LIQUIDITY) This condition requires that a cost function should ensure that $\forall \theta_i < h^0$, an agent's unfunded utility $(r_i^{t_i} - x_i)$ is monotonically increasing in x_i, that is, $\forall q^{t_i}, \forall x_i < h^0$, $\quad \frac{\partial}{\partial x_i}(r_i^{t_i} - x_i) > 0 \Rightarrow \frac{\partial r_i^{t_i}}{\partial x_i} > 1$.

Theorem 3 *Let $C : \mathbb{R}^2 \to \mathbb{R}$ be a cost function that satisfies* CONDITIONS 1-4,6 *and $C_0 : \mathbb{R} \to \mathbb{R}$ be the cost function obtained from C by fixing the number of positive outcome securities. If C_0 satisfies* CONDITION-7 *and is used in PPS for crowdfunding a project with provision point h^0 when $\vartheta > C_0^{-1}(h^0 + C_0(0))$, the strategies in the set $\Big\{(\psi_i^* = \{x_i^*, a_i\}) : x_i^* \le (C_0(\theta_i + q^{a_i}) - C_0(q^{a_i}))$ if $h^{a_i} > 0$, otherwise $x_i^* = 0$; $\chi = h^0\Big\}$ are sub-game perfect equilibria.*

Proof: First we claim in Step 1 that, at equilibrium, $\chi = h^0$. In Step 2, we characterize the equilibria strategy of agent i (ψ_i^*). Step 3 proves the upper bound on b. We show that these equilibria strategies are sub-game perfect in Step 4.

Step 1: In equilibrium, $\chi > h^0$ cannot hold since the requester stops collecting the funds at $\chi = h^0$ if this happens before the deadline T. In equilibrium, $\chi < h^0$ cannot hold since an agent can increase his utility by contributing more and receiving a higher r_i due to CONDITION-7. Thus, in equilibrium $\chi = h^0$.

Step 2: Due to ASSUMPTION-2, agents do not have any bias in believing whether the project will be funded, other than the contributions. From Step 1, the contributions would be such that the project is funded in equilibrium. Thus, at equilibrium, an agent will contribute such that his funded utility is no less than the highest possible unfunded utility. That is, if (x_i^*, t_i^*) is agent's equilibrium strategy, $r_i^* - x_i^* \le \theta_i - x_i^* \Rightarrow r_i^* \le \theta_i$. Expressing r_i^* in terms of x_i^* and t_i^* using Equation (6), we get the condition:

$$\begin{aligned} & C_0^{-1}(x_i^* + C_0(q^{t_i})) - q^{t_i} \le \theta_i \\ \text{or equivalently,} \quad & x_i^* \le C_0(\theta_i + q^{t_i}) - C_0(q^{t_i}) \end{aligned}$$

Note that (i) the RHS of Equation (8) is a monotonically decreasing function of $q^{t_i^*}$ and (ii) q^t, the number of securities allotted by the market at time t, is a monotonically non-decreasing function of t. Thus, an agent with value θ_i minimizes the RHS at $t_i^* = a_i$, that is, he contributes as soon as he arrives[8]. Thus, $\psi_i^* = (x_i^*, a_i)$ and at equilibrium:

$$x_i^* \le C_0(\theta_i + q^{a_i}) - C_0(q^{a_i}) \tag{8}$$

Step 3: Summing up $r_i^* - x_i^* \le \theta_i - x_i^*$ for all agents leads to the condition $\sum_{i=1}^n r_i^* \le \vartheta$. Since securities are allocated using a path independent cost function (CONDITION-1), using Equation (7) the condition for Nash Equilibrium becomes:

$$C_0^{-1}(h^0 + C_0(0)) < \vartheta \tag{9}$$

Step 4: These equilibria, specified as a function of the aggregate history (h^{a_i}), are also sub-game perfect. Consider agent j who arrives last at a_j. If $h^{a_j} = 0$, then his best strategy is $x_j^* = 0$. If $h^{a_j} > 0$, irrespective of H^{a_j} and h^{a_j}, his funded and unfunded utility are the same at x_j^*, defined in the theorem and still it is best response for j to follow the equilibrium strategy. With backward induction, by similar reasoning, it is best response for every agent to follow the equilibrium strategy irrespective of history. Thus, these equilibria are also sub-game perfect equilibria. □

The above theorem characterizes a set of sub-game perfect equilibria at which crowdfunding projects using PPS gets fully funded. Since no agent, without any additional information regarding the project getting funded or not (ASSUMPTION-2), should invest no more than the bound of Equation (8), we believe that this is the only set of sub-game perfect equilibria of induced game at which the project gets fully funded. We are yet to identify any other Nash equilibria for the game induced by PPS.

5 PPS WITH DIFFERENT COST FUNCTIONS

In this section, we undertake a comparison of PPS instantiated using two popular cost functions: logarithmic scoring rule based and quadratic scoring rule based. In both cases, the cost functions satisfy CONDITIONS 1-4,6. A well known criterion for choosing the cost function in prediction markets is the the trade off between the worst case loss and market liquidity [7]. In PPS, CONDITION-7 explicitly lower bounds the liquidity in the market. Interestingly, this upper bounds the refund budget and thus the worst case loss. Thus for PPS, the key consideration in choosing the cost function comes from the trade off between satisfying CONDITION-7 and the maximum bonus that can be offered to incentivize agents to contribute to the public project earlier.

[8] For an intuitive explanation, See Section 5.3

5.1 LMSR-PPS

In our complex cost function based prediction market, LMSR-PPS, which uses the cost function specified in Equation (3) is specified as:

$$
\begin{aligned}
C_0(q^t) &= b\ln(1+\exp(q^t/b)) \\
p^t &= \frac{\exp(q^t/b)}{1+\exp(q^t/b)} \\
\texttt{Cost}(r^t|q^t) &= C_0(q^t+r^t) - C_0(q^t) \\
&= b\ln\left(\frac{1+\exp(\frac{q^t+r^t}{b})}{1+\exp(\frac{q^t}{b})}\right)
\end{aligned}
$$

An agent who contributes x_i at t_i receives $r_i^{t_i}$ securities where:

$$
\begin{aligned}
x_i &= b\ln\left(\frac{1+\exp(\frac{q^{t_i}+r_i^{t_i}}{b})}{1+\exp(\frac{q^{t_i}}{b})}\right) \text{ and} \\
r_i^{t_i} &= b\ln\left(\exp\left(\frac{x_i}{b} + \ln(1+\exp(\frac{q^{t_i}}{b}))\right) - 1\right) - q^{t_i}
\end{aligned}
$$

Proposition: LMSR-PPS satisfies CONDITION-7.

Proof: CONDITION-7 requires that $\forall q^{t_i}, \forall x_i < h^0, \quad \frac{\partial r_i^{t_i}}{\partial x_i} > 1$. With LMSR-PPS,

$$
\frac{\partial r_i^{t_i}}{\partial x_i} = \frac{\exp\left(\frac{x_i}{b} + \ln(1+\exp(\frac{q^{t_i}}{b}))\right)}{\exp\left(\frac{x_i}{b} + \ln(1+\exp(\frac{q^{t_i}}{b}))\right) - 1}
$$

Since the RHS is always greater than 1, CONDITION-7 is always satisfied for LMSR-PPS. We note that this is an immediate implication of the infinite liquidity of LMSR based prediction markets [14]. □

Corollary: If $\vartheta > h^0$ and $b > 0$, in LMSR-PPS, the strategies in the set $\left\{(\psi_i^* = \{x_i^*, a_i\}) : x_i^* \leq b\ln\left(\frac{1+\exp\left(\frac{\theta_i+q^{a_i}}{b}\right)}{1+\exp(\frac{q^{a_i}}{b})}\right) \text{ if } h^{a_i} > 0, \text{ otherwise } x_i^* = 0; \chi = h^0\right\}$ are sub-game perfect equilibria if $b < \frac{\vartheta - h^0}{\ln 2}$.

Proof: Since LMSR-PPS satisfy CONDITIONS 1-4,6-7, Theorem 3 is applicable and Equation (8), the equilibrium contribution of agent i with value θ_i who arrives at a_i is:

$$
x_i^* \leq C_0(\theta_i + q^{a_i}) - C_0(q^{a_i}) = b\ln\left(\frac{1+\exp\left(\frac{\theta_i+q^{a_i}}{b}\right)}{1+\exp(\frac{q^{a_i}}{b})}\right)
$$

The condition for attaining this equilibrium corresponding to Equation (9) is:

$$
\begin{aligned}
C_0^{-1}(h^0 + C_0(0)) &= b\ln\left(\exp\left(\frac{h^0}{b} + \ln(2) - 1\right)\right) \\
&< b\ln\left(\exp\left(\frac{h^0}{b} + \ln(2)\right)\right) \\
&< \vartheta \\
\Rightarrow b < \frac{\vartheta - h^0}{\ln 2}
\end{aligned}
$$

□

5.2 QSR-PPS

In our complex cost function based prediction market, the QSR-PPS, which uses the cost function specified in Equation (4) is specified as:

$$
\begin{aligned}
C_0(q^t) &= \frac{q^t}{2} + \frac{(q^t)^2}{8b} - \frac{b}{2} \\
p^t &= \frac{1}{2} + \frac{q^t}{4b} \\
\texttt{Cost}(r^t|q^t) &= C_0(q^t+r^t) - C_0(q^t) \\
&= r^t\left(\frac{1}{2} + \frac{q^t}{4b} + \frac{r^t}{8b}\right)
\end{aligned}
$$

An agent who contributes x_i at t_i receives $r_i^{t_i}$ securities where:

$$
\begin{aligned}
x_i &= r_i^{t_i}\left(\frac{1}{2} + \frac{q^{t_i}}{4b} + \frac{r_i^{t_i}}{8b}\right) \text{ and} \\
r_i^{t_i} &= \sqrt{(q^{t_i}+2b)^2 + 8bx_i} - (q^{t_i}+2b)
\end{aligned}
$$

Proposition: QSR-PPS satisfies CONDITION-7 if $b > \frac{2}{3}h^0$

Proof: CONDITION-7 requires that $\forall q^{t_i}, \forall x_i \leq h^0, \quad \frac{\partial r_i^{t_i}}{\partial x_i} > 1$. With QSR-PPS,

$$
\frac{\partial r_i^{t_i}}{\partial x_i} = \frac{4b}{\sqrt{(q^{t_i}+2b)^2 + 8bx_i}}
$$

The RHS obtains its minimum value with $q^{t_i} = 0$ and $x_i = h^0$: this corresponds to the condition when the first agent who arrives contributes the whole amount for the public project. Ensuring that this minimum is greater than 1 leads to the condition $b > \frac{2}{3}h^0$. □

Corollary: If $\vartheta > h^0$ and $b > 0$, in QSR-PPS, the strategies in the set $\left\{(\psi_i^* = \{x_i^*, a_i\}) : x_i^* \leq \theta_i\left(\frac{1}{2} + \frac{q^{a_i}}{4b} + \frac{\theta_i}{8b}\right) \text{ if } h^{a_i} > 0, \text{ otherwise } x_i^* = 0; \chi = h^0\right\}$ are sub-game perfect equilibria if $b < \frac{\vartheta^2}{8h^0}$.

Proof: With $b > \frac{2}{3}h^0$, since QSR-PPS satisfies CONDITIONS 1-4,6-7, Theorem 3 is applicable and Equation (8), the equilibrium contribution of agent i with value θ_i who arrives at a_i is:

$$
x_i^* \leq C_0(\theta_i + q^{a_i}) - C_0(q^{a_i}) = \theta_i\left(\frac{1}{2} + \frac{q^{a_i}}{4b} + \frac{\theta_i}{8b}\right)
$$

The condition for attaining this equilibrium corresponding to Equation (9) is:

$$
\begin{aligned}
C_0^{-1}(h^0 + C_0(0)) &= \sqrt{(2b)^2 + 8bh^0} - 2b \\
&< \sqrt{8bh^0} \\
&< \vartheta \\
\Rightarrow b < \frac{\vartheta^2}{8h^0}
\end{aligned}
$$

□

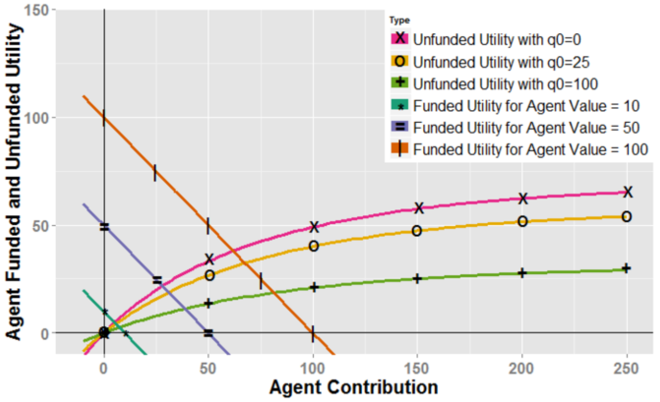

Figure 2. LMSR based PPS: utility vs. contribution

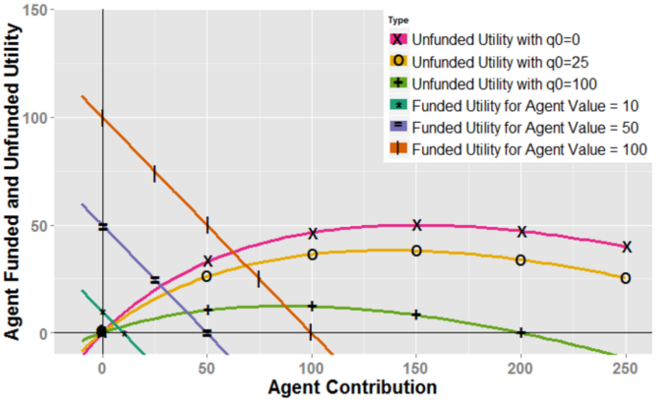

Figure 3. QSR based PPS: utility vs. contribution

5.3 Choosing a Cost Function for PPS

Sections 5.1 and 5.2 show that both LMSR and QSR based PPS can achieve successful crowdfunding of a public project under the right conditions. In both cases, the condition that $\vartheta > C_0^{-1}(h^0 + C_0(0))$ (Theorem 3) leads to an upper bound on b. For LMSR-PPS, $b < \frac{\vartheta - h^0}{\ln 2}$ and for QSR-PPS, $b < \frac{\vartheta^2}{8h^0}$. This upper bound ensures that the the refund bonus is not so high that it exceeds the net value of the project.

A lower bound on b (and hence the refund bonus) comes from CONDITION-7. In LMSR-PPS, the unfunded utility is always a monotonically increasing function of the contribution and CONDITION-7 is always satisfied. Thus, the lower bound on b is trivial ($b > 0$) and all agents have an incentive (no matter how small) to contribute. In practice, if the refund bonus in LMSR is too small to incentivize agents, then CONDITION-7 can be modified to require that $\forall q^{t_i}, \forall x_i < h^0, \quad \frac{\partial r_i^{t_i}}{\partial x_i} > (1+\epsilon)$ and we can show that $b > h^0/\ln(\frac{1}{2\epsilon})$.

In QSR-PPS, the unfunded utility first increases then decreases and thus satisfying CONDITION-7 creates a lower bound on the refund budget ($b > \frac{2}{3}h^0$). Intuitively, if b is too low than agents who arrive after the contributions have crossed a certain threshold (but not yet reached the provision point) will not have an incentive to contribute.

Figure 2 and 3 compare the performance of LMSR-PPS and QSR-PPS with $b = 100$. In both the figures, the three straight lines correspond to funded utility of agents with different types. Since the unfunded utility depends both on the contribution and the timing of the contribution (via the number of outstanding securities at the time of the contribution), the unfunded utility as a function of the contribution for three different histories (different number of outstanding negative outcome securities) are shown. Note that the unfunded utility is independent of agent type.

If more securities have been sold at the time agent i contributes, he must contribute more to obtain the same unfunded utility. For a given history (number of outstanding securities), the point where the curve (unfunded utility) intersects the line (funded utility) is the contribution amount where the agent derives the same utility independent of whether or not get the project is funded. The set of equilibria x_i^* lie to the left of this intersection.

5.4 Price and Probability in PPS

In any cost function based prediction market, the gradient of the cost function is interpreted as both the instantaneous price of a security and the market probability of the associated event. In PPS however, this needs to be reinterpreted. For every contribution x_i towards the public project, PPS allocates securities associated with the project *not* getting funded. As the contributions near the target ($\chi \to h^0$), the price of the project *not* getting funded nears one ($p \to 1$): this is desirable since PPS is designed to incentivize early contributions. However, interpreting p as the probability of the project *not* getting funded is counter intuitive. Instead we propose to interpret $1 - p$ as the market probability that the project will not be funded. The range of instantaneous price (p) and probability ($1 - p$) also depend on the number of securities with which the market is initialized. If $\mathbf{q} = (0, 0)$, then $p \in [0.5, 1]$. If the market is initialized with $\mathbf{q} = (q_{\omega_1}, 0)$ then the price space expands or shrinks depending on the value of q_{ω_1}. Thus, initialization of $\mathbf{q}$ can be used to control the price space in PPS.

6 Conclusion

In this work, we have proposed a class of provision point mechanisms, PPS for civic crowdfunding. PPS induces an extensive form game among the agents who arrive on the crowdfunding platform and achieves equilibria at which the project is funded. These equilibria have the desirable property that agents do not free ride but instead contribute in proportion to their true value for the project and do so as soon as they arrive. PPS achieves this by incentivizing agents with a refund greater than their contribution if the project is not funded. In PPS, securities issued in a cost function based prediction market determine the refund bonus. Even though PPS relies on a *sponsor* to offer a refund bonus while the funds are being collected, the bonus is not paid out at equilibrium. As these securities are purchased, the price increases, thereby incentivizing participants to contribute earlier. We specified the conditions that a cost function must satisfy to be used in PPS and compared PPS under two popular cost functions. Using these as the benchmark, we provided considerations to choose an optimal cost function. We believe that our work can significantly improve the success rate of provisioning public projects using private funds in scenarios like civic crowdfunding.

REFERENCES

[1] Spacehive, 2015. Available online at `http://www.spacehive.com`.

[2] Jacob Abernethy, Yiling Chen, and Jennifer Wortman Vaughan, 'Efficient market making via convex optimization, and a connection to online learning', *ACM Transactions on Economics and Computation*, **1**(2), 12, (2013).

[3] Kenneth J Arrow, 'The role of securities in the optimal allocation of risk-bearing', *The Review of Economic Studies*, **31**(2), 91–96, (1964).

[4] Mark Bagnoli and Barton L Lipman, 'Provision of public goods: Fully implementing the core through private contributions', *The Review of Economic Studies*, **56**(4), 583–601, (1989).

[5] Earl R Brubaker, 'Free ride, free revelation, or golden rule?', *The Journal of Law & Economics*, **18**(1), 147–161, (1975).

[6] Yan Chen, 'Incentive-compatible mechanisms for pure public goods: a survey of experimental research', *The Handbook of Experimental Economics Results, Amsterdam: Elsevier Press, forthcoming*, (1999).

[7] Yiling Chen, 'A utility framework for bounded-loss market makers', in *In Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*, (2007).

[8] Yiling Chen, Stanko Dimitrov, Rahul Sami, Daniel M Reeves, David M Pennock, Robin D Hanson, Lance Fortnow, and Rica Gonen, 'Gaming prediction markets: Equilibrium strategies with a market maker', *Algorithmica*, **58**(4), 930–969, (2010).

[9] Yiling Chen, Lance Fortnow, Nicolas Lambert, David M Pennock, and Jennifer Wortman, 'Complexity of combinatorial market makers', in *Proceedings of the 9th ACM conference on Electronic commerce*, pp. 190–199. ACM, (2008).

[10] Yiling Chen and Jennifer Wortman Vaughan, 'A new understanding of prediction markets via no-regret learning', in *Proceedings of the 11th ACM conference on Electronic commerce*, pp. 189–198. ACM, (2010).

[11] citizinvestor.com. citizinvestor, 2015. Available online at `http://www.citizinvestor.com`.

[12] Theodore Groves and John Ledyard, 'Optimal allocation of public goods: A solution to the" free rider" problem', *Econometrica: Journal of the Econometric Society*, 783–809, (1977).

[13] Mingyu Guo and David M Pennock, 'Combinatorial prediction markets for event hierarchies', in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pp. 201–208. International Foundation for Autonomous Agents and Multiagent Systems, (2009).

[14] Robin Hanson, 'Combinatorial information market design', *Information Systems Frontiers*, **5**(1), 107–119, (2003).

[15] Robin Hanson, 'Logarithmic markets coring rules for modular combinatorial information aggregation', *The Journal of Prediction Markets*, **1**(1), 3–15, (2012).

[16] Robin Hanson et al., 'Shall we vote on values, but bet on beliefs?', *Journal of Political Philosophy*, (2003).

[17] Paul J Healy, 'Learning dynamics for mechanism design: An experimental comparison of public goods mechanisms', *Journal of Economic Theory*, **129**(1), 114–149, (2006).

[18] Leslie M Marx and Steven A Matthews, 'Dynamic voluntary contribution to a public project', *The Review of Economic Studies*, **67**(2), 327–358, (2000).

[19] John Morgan, 'Financing public goods by means of lotteries', *The Review of Economic Studies*, **67**(4), 761–784, (2000).

[20] neighbourly.com. neighbourly, 2015. Available online at `http://www.neighbourly.com`.

[21] A Othman and T Sandholm, 'Homogeneous risk measures and liquidity-sensitive automated market makers', in *Proceedings of the 7th Workshop on Internet and Network Economics*, pp. 314–325, (2011).

[22] Abraham Othman, David M Pennock, Daniel M Reeves, and Tuomas Sandholm, 'A practical liquidity-sensitive automated market maker', *ACM Transactions on Economics and Computation*, **1**(3), 14, (2013).

[23] Thomas R Palfrey and Howard Rosenthal, 'Participation and the provision of discrete public goods: a strategic analysis', *Journal of public Economics*, **24**(2), 171–193, (1984).

[24] Walter Rudin, *Real and complex analysis*, Tata McGraw-Hill Education, 1987.

[25] David Schmidtz. The limits of government (boulder, 1991.

[26] Alexander Tabarrok, 'The private provision of public goods via dominant assurance contracts', *Public Choice*, **96**(3-4), 345–362, (1998).

[27] Robertas Zubrickas, 'The provision point mechanism with refund bonuses', *Journal of Public Economics*, **120**, 231–234, (2014).