Information Design in Crowdfunding under Thresholding Policies

Wen Shen*, Jacob W. Crandall†, Ke Yan‡, and Cristina V. Lopes§

*‡Department of Informatics, University of California, Irvine, Irvine, CA 92697, USA
† Computer Science Department, Brigham Young University, Provo, UT 84602, USA
‡ College of Information Engineering, China Jiliang University, Hangzhou 310018, China

ABSTRACT
Crowdfunding has emerged as a prominent way for entrepreneurs to secure funding without sophisticated intermediation. In crowdfunding, an entrepreneur often has to decide how to disclose the campaign status in order to collect as many contributions as possible. Such decisions are difficult to make primarily due to incomplete information. We propose information design as a tool to help the entrepreneur to improve revenue by influencing backers’ beliefs. We introduce a heuristic algorithm to dynamically compute information-disclosure policies for the entrepreneur, followed by an empirical evaluation to demonstrate its competitiveness over the widely-adopted immediate-disclosure policy. Our results demonstrate that the immediate-disclosure policy is not optimal when backers follow thresholding policies despite its ease of implementation. With appropriate heuristics, an entrepreneur can benefit from dynamic information disclosure. Our work sheds light on information design in a dynamic setting where agents make decisions using thresholding policies.

INTRODUCTION
Crowdfunding reinvents the way that entrepreneurs raise external funding for implementing creative ideas. It has created a rapidly growing market that contributes an annual economic impact of tens of billions of US dollars globally [45]. Unfortunately, not all the crowdfunding campaigns are successful because most campaigns will get funded only if they have reached the fundraising goal within a deadline [37]. In fact, less than 40% of the crowdfunding projects reach the targeted goals and receive the funds within the campaign deadlines [37].

Mounting research has begun to investigate the determinants of the success of crowdfunding projects. Although there might be many factors (e.g., project descriptions [26], product value [1], geography effect [2], reward details [26], entrepreneurs’ reputation [23] and the social network effect [1]) that influence a campaign’s success, a recent study indicates that the number of donations made by early backers of a project is often the only difference between that project being funded or not [39]. A substantial body of both theoretical analyses [1, 3, 28] and empirical evidence [13, 23, 26, 38] demonstrate that the amount of early contributions has a strong positive effect on the success of crowdfunding campaigns. These prior studies unanimously confirm the crucial role of early contributions in the success of crowdfunding projects.

There are two main reasons why the amount of early contributions matters. First, information about contributions received early in the campaign signals to potential backers the quality of the project, which in turn can trigger social learning behavior [7] causing potential backers to also contribute to the campaign [13]. An empirical study on a sample of 25,058 Kickstarter projects indicates that prospective backers usually make their pledging decisions based on how much of the project goal has already been funded by others [23]. Second, backers who have made an early contribution are likely to circulate the information of the project to their friends or families, which may attract additional contributions [13, 38]. Both rationales indicate that it is of interest to entrepreneurs to attract as many contributions from early backers as possible.

In crowdfunding, backers are often reluctant to donate in the early days of a campaign due to high uncertainty [3, 13, 23, 28, 39]. A major source of uncertainty is the probability of success that the campaign will get funded (i.e., Probability of Success, or PoS) [13, 23]. Prospective backers are often uncertain about entrepreneurs’ abilities to collect sufficient contributions to get the project funded. For instance, 64.12% of the crowdfunding projects in Kickstarter failed to reach the target goals [21]. A backer experiences a monetary or non-monetary opportunity cost if the fundraising goal is not achieved (and the project not funded), even if he is refunded upon the failure of the campaign [3].

To attract as many early contributions as possible, an entrepreneur must take appropriate measures to coordinate backers’ actions. To do this, the entrepreneur needs to have prior knowledge about the backers’ arrival process, their valuation of the project (if funded), and how they estimate the probability that the campaign will be funded. However, none of this information is perfectly known to the entrepreneur. Thus, it is challenging for the entrepreneur to figure out what actions will make backers, especially early backers, be more willing to contribute.

If conditions permit, the entrepreneur can manipulate backers’ payoffs by offering appealing discounts to early backers that face high uncertainty [16, 40]. The problem of devising allocation and
payment schemes falls into the field of mechanism design [29]. While illuminating, it requires additional budgets and thus diminishes the entrepreneur’s revenue [16, 41]. Absent from sophisticated or even unrealistic assumptions of the backers’ private types (e.g., valuation, arrival time, departure time), it is rather difficult or even unfeasible for the entrepreneur to implement effective mechanisms [16, 41]. This is particularly the case in online settings where the entrepreneur has little knowledge about how the backers make their projections of the campaign’s PoS.

Alternatively, the entrepreneur can improve backers’ beliefs of the campaign’s PoS by choosing what information backers see. In particular, the entrepreneur can and is permitted to voluntarily disclose the project status (i.e., how many contributions have been collected up to a given timestamp), a critical factor that influences backers’ beliefs of the campaign’s PoS [3, 23]. The problem of determining which pieces of information are disclosed to whom is called information design [43].

Prior work on information design has generally assumed that backers’ strategic behavior was perfectly rational and that games were well-defined (e.g., signaling games) [4, 8–10, 43]. However, studies on consumer purchasing behavior show that buyers usually follow thresholding policies to decide whether to purchase goods or not [18, 20, 46]. They often buy products when the prices are no more than their reserved values. This is particularly the case when consumers face high degrees of uncertainty and have little knowledge about the environment or the future, as frequently observed in clinical decision making [31], crowd-sourcing contests [15], airline ticket sales [46], online shopping [25], management science [42], societies of autonomous machines [34] and crowdfunding [3, 28]. Under certain circumstances, thresholding policies are optimal policies and hence represent rational behavior [30]. We thus consider the scenario where backers follow thresholding policies when they decide whether to contribute to a project or not.

In this paper, we study the information design problem in which an entrepreneur voluntarily reveals the project status to backers to influence their beliefs of the project’s success. Our work contributes to the state of the art in the following ways:

1. We show that excessive information disclosure weakly shrinks the entrepreneur’s revenue. We identify conditions when immediate disclosure is optimal in crowdfunding when agents follow thresholding policies. We demonstrate that immediate disclosure is optimal if the funding goal has been achieved and if the project status increases monotonically by at least one contribution each time.

2. We introduce a heuristic algorithm called Dynamic Shrinkage with Heuristic Selection (DDSHS) to help the entrepreneur make decisions on information-disclosure policies.

3. Our experimental results demonstrate that the immediate-disclosure policy is not optimal when agents follow cutoff policies despite that it is more efficient to compute than the heuristic methods. Entrepreneurs can benefit from dynamic information design with appropriate heuristics.

**DECISION MAKING IN CROWDFUNDING**

This section first introduces key notations. It then formalizes backers’ decision model and the entrepreneur’s optimization problem.

---

**Preliminaries**

We consider discrete time $t \in T = \{1, 2, 3, \ldots, T\}$, where $T$ is the deadline for the campaign. Before launching the campaign, the entrepreneur must determine a fundraising goal $G$ to get funded, a deadline $T$ for reaching the goal, the number of rewards $N$, the minimal amount of contributions for a reward $P$, and a detailed description of the project such as motivation, product, milestones, and profiles of the team. All this information is fixed and is disclosed to all the backers. See Figure 1 for the procedure of a typical crowdfunding campaign.

![Figure 1: Procedure of a typical crowdfunding campaign.](image)

After the campaign starts to accept contributions, backers arrive at the campaign sequentially with at most one each time. This is without loss of generality because batch arrivals can be viewed as a special case where the time interval is minimal [3, 36]. Let $b(t) \in \{0, 1\}$ denote the number of arrivals at time $t \in T$.

At the beginning of time $t$, the entrepreneur discloses the state of the campaign (i.e., project status) $s(k)$ to each backer $i$ that is in the campaign. Here, $s(k) = (S_k, k)$ where $S_k$ refers to the percentage of funds that have been raised up to time $k \leq t$ (not included), with respect to the fundraising goal $G$. For simplicity, let $|s(k)| = S_k$. We denote the entrepreneur’s decision on information disclosure for backer $i$ at time $t$ by: $d(i, t) = (s(i), t)$ s.t. $k \in T$, $k \leq t$ . Here, $|s(1)| = 0$. The disclosed project status $s(k)$ must reflect the true state of the project at time $k$, which is enforced by the crowdfunding platform. In real-world crowdfunding campaigns, entrepreneurs are allowed to voluntarily disclose truthful project status [3, 39]. In our work, we assume that any information about the project status observed by the backers is directly revealed by the entrepreneur. Future work should address the scenarios when backers have exogenous information due to information contagion [5].

**Backers’ Decision Model**

It is widely known that backers’ beliefs of PoS are usually correlated with the entrepreneur’s updates of the project status [3, 23, 24, 26]. However, the exact correlation is privately known to a backer himself only and not observed by the entrepreneur. This makes it difficult to accurately model backers’ decision making process. To tackle this problem, we formalize backers’ decision model using the same pattern as that in the work [30] by Ohannessian et al., which also assumes that agents use thresholding policies.

Let $r_i(t, d(i, t)) \in [0, 1]$ represent high-value backer $i$’s estimate of the campaign’s PoS given the report $d(i, t)$, and $\phi_i \in (0, 1]$ be his
threshold on \( r_i(\cdot) \) to contribute. We denote backer \( i \)'s decision on whether to contribute or not at time \( t \) by \( a_i(t) \in \{0, 1\} \), where 0 indicates Not Pledging, and 1 represents Pledging. Backer \( i \)'s expected utility \( u_i \) is determined as follows:

\[
u_i(t, a_i(t), d(i, t)) = \begin{cases} 
    c_i \cdot a_i(t), & \text{if } r_i(t, d(i, t)) \geq \phi_i; \\
    0, & \text{otherwise}. 
\end{cases} \tag{1}
\]

Here, \( c_i > 0 \) is backer \( i \)'s expected utility if he contributes (i.e., \( a_i(t) = 1 \)) when his estimate of PoS is no less than the threshold \( \phi_i \). Note that \( c_i, r_i(\cdot) \) and \( \phi_i \) are all private information known to backer \( i \) only, while his arrival and pledging behavior are observed by the entrepreneur through the platform. In practice, backer \( i \) may adapt to the environment and update his threshold accordingly. In this case, his threshold \( \phi_i \) can be treated like the upper bound of all the updated thresholds. Without loss of generality, we assume that each contributing backer pledges the same amount \( P \) of fund to the project for a reward.

Backer \( i \) stays at the campaign for at most \( l_i \in \{1, 2, 3, \ldots, L\} \) periods, where \( l_i \) is known to backer \( i \) only. This is without loss of generality because although backers may dynamically enter and exit the system and check the progress, these situations can be viewed as the case that the backers stay in the system for a sufficient period.

Let \( I(t) \) denote the group of backers who have arrived at the campaign before or at time \( t \), have at least one time period to leave and have not yet claimed a contribution. At time \( t \), for each backer \( i \in I(t) \), his objective function is

\[
B_i(t) = \max_{a_i(t)} \nu_i(t, a_i(t), d(i, t)) \quad \text{s. t. } G, T, N, P, l_i , \tag{2}
\]

where \( u_i \) is determined by Equation 1. At time \( t \), backer \( i \) will leave the campaign either if he claims a contribution (i.e., \( a_i(t) = 1 \)) or his own deadline \( l_i \) is reached.

**The Entrepreneur’s Optimization Problem**

In crowdfunding, the entrepreneur is interested in attracting as many contributions as possible within the deadline so that her project will get funded. Specifically, her objective is to set the disclosure policy such that the number of contributions is maximized until a given deadline \( T \).

Let \( M(t) \) denote the funds that the entrepreneur has raised up to time \( t \) (if included) when she uses the disclosure policy \( DP(t) \). Here, \( DP(t) = ((d(i, t'))_{i \in I(t')}\}_{t' \leq t} \). The entrepreneur’s expected contributions at time \( t \) is defined as follows:

\[
M(t) = \sum_{t' = 1}^{t} \sum_{i \in I(t')} \alpha_i(t') \cdot P \quad \text{s. t. } G, T, N, P . \tag{3}
\]

Due to the deadline constraint, the entrepreneur’s optimization problem (i.e., optimal information design) is formalized as follows:

**Definition 1 (Optimal Information Design).** An optimal information design in crowdfunding is to find a disclosure policy \( DP_{opt}(T) \), such that \( M(T) \) is maximized, i.e., \( DP_{opt} = \arg\max_{DP(T)} M(T) \).

Due to the dynamic nature of crowdfunding, the design of disclosure policy \( DP(t) \) cannot be based on backers’ later decisions (\( a_i(\hat{t})_{i \in I(\hat{t})} \)) or use later project status \( (s(\hat{t}))_{\hat{t} > T} \). This constraint is called No Clairvoyance.

**OPTIMAL INFORMATION DESIGN**

After introducing the solution concepts, we show that excessive information weakly shrinks revenue. We further identify conditions under which immediate disclosure is optimal in crowdfunding.

**Solution Concepts**

In his seminal work [11], Blackwell formulated a partial order that compares the quality of two pieces of information (see Theorem 1). According to Blackwell’s theorem, if a piece of information \( \zeta_2 \) is Blackwell-inferior to \( \zeta_1 \), then an agent will always weakly prefer \( \zeta_1 \) to \( \zeta_2 \).

**Theorem 1 (Blackwell’s theorem [11]).** Let \( \zeta_1 \) and \( \zeta_2 \) represent two pieces of information, the following conditions are equivalent:

1. (When the agent chooses \( \zeta_1 \), her expected utility is always at least as big as the expected utility when she chooses \( \zeta_2 \), independent of the utility function and the distribution of the input. (2) \( \zeta_2 \) is a garbling of \( \zeta_1 \).

Blackwell’s theorem implies two types of information: vertical information and horizontal information, which are key solution concepts used in this work. Given two pieces of information, if one is always (weakly) preferred whatever the information receivers’ types are, then they are vertical information (see Definition 2). If the two pieces of information are not comparable without prior knowledge about the receivers’ types, then the information is horizontal (definition omitted since it complements vertical information).

**Definition 2 (Vertical Information).** Given two pieces of information \( \zeta_1 \) and \( \zeta_2 \), where \( \zeta_1 \not\equiv \zeta_2 \), if \( \forall t \in I, u_i(\zeta_1) \geq u_i(\zeta_2) \), then \( \zeta_1 \succeq \zeta_2 \), where \( I \) denotes a set of agents and \( \succeq \), indicating preferred or indifferent to, is independent of agent \( i \)'s private type. If \( \zeta_1 \succ \zeta_2 \), the information is vertical.

In crowdfunding, backer \( i \)'s estimate of the campaign’s PoS (i.e., \( r_i(t, d(i, t)) \)) is both time and state-dependent. Both the amount of funding (measured by \( |s(k)| \)) raised, and the time of the project status (denoted by \( k \)) are important. We identify three scenarios of vertical information. First, a higher state of project status is always more favorable if the time of the state is the same (e.g., \( 20\% > 10\% \)), which is obvious. Second, the earlier report of project status is always weakly preferred if the project status of the two reports are the same (see Proposition 1). Third, the later report of project status is always weakly preferred if the revenue increases by more than \( P \) each time between the timestamps of the two statuses. (see Proposition 2). All the proofs can be found in Appendix A.

**Proposition 1.** An earlier report of project status is always weakly preferred if the project status of the two reports are the same. Formally, given project status \( s(k_1) \) and \( s(k_2) \), where \( k_1 < k_2 \), \( |s(k_1)| = |s(k_2)| \), we have \( \forall t \geq k_2, \forall i \in I(t): (s(k_1), t) \succeq (s(k_2), t) \).

**Proposition 2.** A later report of project status is always weakly preferred if the revenue increases by more than \( P \) each time between the period of the two statuses. Given status reports \( s_1(t_1) \).
and $e_2 : (s(k_2), t_2)$, where $k_1 < k_2$, $t_1 < t_2$ and $|s(k_2)| - |s(k_1)| \geq (k_2 - k_1) \cdot P/G$, we have $\forall t \geq k_2, \forall i \in I(t) : e_2 \succapprox e_1$.

If the conditions of vertical information cannot be identified, the information is horizontal (see Example 1). Without prior information about backers’ private types (e.g., arrival process, valuation, the estimate of the campaign’s PoS, and the correlation between them), it is not feasible for the entrepreneur to identify optimal information design. However, an effective disclosure policy should capture both the vertical and horizontal component, making the information design problem particularly challenging for the entrepreneur.

Example 1. Given $T = 30, P = 0.1G$, $|s(10)| = 30\%$ and $|s(15)| = 40\%$, without prior knowledge about backer $i$’s projection of PoS (i.e., $r_i(t)$), it is unclear which project status is more favorable by $i$. This is because $|s(15)| - |s(10)| = 0.1 < 0.1 \cdot (15 - 10)$.

Excessive Disclosure Shrinks Revenue

Given two project status reports, if their partial order can be identified according to Proposition 1 and 2, then the entrepreneur only needs to disclose the one with higher order. This is because revealing the low-order report does not increase the chance that backers contribute to the campaign (see Lemma 1).

Lemma 1. If the order of two project status reports can be identified, the low-order report does not increase the chance of backers’ contribution. Formally, given two reports $e_1 : (s(k_1), t_1)$ and $e_2 : (s(k_2), t_2)$, if $\forall t \geq \max(k_1, k_2), k_1 \neq k_2, \forall i \in I(t) : e_2 \succapprox e_1$, we have: $E(a_i = 1|t) \geq E(a_i = 1|e_1, e_2)$, where $E(a_i = 1|r)$ denotes the expectation that backer $i$ contributes to the campaign given the information $r$.

If the partial order of the two reports cannot be identified, the entrepreneur should also refrain from disclosing additional information. Depending on how backers estimate the campaign’s PoS, revealing more information than necessary can decrease the revenue. The reason is that excessive information disclosure can decrease backers’ projections of the PoS (See Lemma 2).

Lemma 2. If the partial order of the two project status reports cannot be identified, excessive information disclosure weakly decrease backers’ projections of PoS. Formally, given two reports $e_1 : (s(k_1), t_1)$ and $e_2 : (s(k_2), t_2)$ where $k_1 \neq k_2$, if the partial order of the two cannot be identified by the entrepreneur, then $\forall t \geq \max(k_1, k_2), \forall i \in I(t)$, we have: $r_i(t, (e_1, e_2)) \leq \max(r_i(t, e_1), r_i(t, e_2))$.

In our model of crowdfunding, excessive disclosure weakly diminishes a backer’s willingness to contribute (see Theorem 2) and thus weakly shrinks the revenue as well as the entrepreneur’s ability to implement optimal information-disclosure policies. The entrepreneur should not disclose more information about the project status than necessary in order to collect as many contributions as possible.

Theorem 2. Excessive information disclosure weakly diminishes the chance that a backer will contribute. Formally, given two project status reports $e_1 : (s(k_1), t_1)$ and $e_2 : (s(k_2), t_2)$ where $k_1 \neq k_2$, let $a_i'$ denote backer $i$’s decision on pledging if given report either $e_1$ or $e_2$, and $a_i''$ denote his decision on contribution if given $(e_1, e_2)$. $\forall t \geq \max(k_1, k_2), \forall i \in I(t)$, we have: $E(a_i' = 1) \geq E(a_i'' = 1)$.

Immediate Disclosure is not Always Optimal

The immediate-disclosure policy (see Definition 3) is widely adopted by entrepreneurs on major crowdfunding platforms (e.g., Kickstarter, Indiegogo) due to its ease of implementation [3]. It is thus important to investigate if immediate disclosure is optimal.

Definition 3 (Immediate Disclosure). An immediate-disclosure policy always reveals the current project status to all the backers in the campaign. Formally let $DP_{im}$ denote immediate disclosure, we have $DP_{im}(t) = (((d(j, t')))_{j \in I(t')}|t' \leq t \text{ s.t. } d(j, t') = (s(t'), t')$.

If the entrepreneur and the backers have identical information, immediate disclosure is optimal [6, 19, 32]. It still holds if the entrepreneur has some unique information provided that such information does not affect the backers’ decisions. Unfortunately, in our model, all information about the campaign (e.g., $G, T, P, N$) except the project status is known to both the entrepreneur and the backers. Backer $i$’s estimate of PoS (i.e., $r_i(t, (d(j, t)))$) is influenced by the project status $s(k)$ that the entrepreneur reveals. Thus, it is critical for the entrepreneur to identify conditions when immediate disclosure is optimal.

From Proposition 2, we see that to improve backer $i$’s belief of the campaign’s PoS, the entrepreneur should always disclose the project status that is preferred by all the backers if available. Otherwise, the information design is not optimal. With this intuition in mind, we show that before the campaign reaches the fundraising goal, immediate disclosure is optimal if and only if the project status increases monotonically in time by at least one contribution each time (see Lemma 3). This condition characterizes the relationship between the growth rates of the revenue and the maximum possible arrival rate of the backers. Though the entrepreneur does not have prior knowledge of the backers’ types (e.g., beliefs, thresholds), she can observe the progress of the project and determine if immediate disclosure is optimal given the tracking record of project status.

Lemma 3. Before the campaign reaches the fundraising goal, immediate disclosure is optimal if and only if the project status increases monotonically in time by at least one contribution each time. Formally, if $M(t) < G$, then we have:

$$DP_{im}(t) = \argmax M(t) \iff \forall t' \leq t : |s(t')| - |s(t' - 1)| \geq P/G.$$ 

After successfully reaching the funding objective, it is certain that the campaign will get funded, so immediate disclosure is optimal (see Lemma 4).

Lemma 4. After the campaign reaches the fundraising goal, Immediate disclosure is optimal. Formally, if $M(t) \geq G$, then we have:

$$DP_{im} = \argmax M(t).$$

Lemma 3 shows that immediate disclosure is not always optimal during the crowdfunding campaign when backers follow a thresholding policy. According to Definition 1 and Equation 3, in order to compute the optimal solution, the entrepreneur must have a prior knowledge of the sequence of decisions $((a_i(t))_{j \in I(t)})_{t \in T}$ in advance. However, such assumption violates the No Clairvoyance constraint and is not implementable in practice.
DYNAMIC INFORMATION DESIGN

Instead of restricting our attention to optimal information design, we introduce a heuristic algorithm, called Dynamic Shrinkage with Heuristic Selection (DSHS), to help the entrepreneur make decisions on information disclosure.

DSHS treats the two conditions separately: before and after project success (see Algorithm 1). Before the campaign reaches the fundraising goal, the algorithm determines the disclosure policy according to two processes: dynamic shrinkage (see Algorithm 2) and heuristic selection. After the campaign reaches the fundraising goal (if it happens), the algorithm discloses information immediately.

**Algorithm 1 DSHS**

**Input:** $t$ - time; $s(t)$ - project status at time $t$; $I(t)$ - backers in the campaign.

**Output:** $(d(i,t))_{i \in I(t)}$ - the entrepreneur’s decisions on information disclosure for backers in the campaign at time $t$.

1. if $t \leq T$ then
2. for each backer $i \in I(t)$ do
3. if current revenue $M(t) < G$ then ▷ before success
4. Include all the available project status into $H_t(t)$
5. Sort $H_t(t)$ in the ascending order of $|s(k)|$
6. Remove the least promising candidates in $H_t(t)$
7. Select the project status $s(k_{sel})$ using heuristics
8. Finalize disclosure decision $d(i,t) \leftarrow s(k_{sel},t)$
9. else ▷ after success
10. Disclosure current status, i.e., $d(i,t) \leftarrow (s(t),t)$
11. end if
12. Update revenue $M(t+1) = M(t) + a_i(t) \cdot P$
13. Update project status $s(t+1) = M(t+1)/G$
14. end for
15. end if

**Dynamic Shrinkage.** In the dynamic-shrinkage process, DSHS ranks all the available choices, and removes the least promising choices which are less preferred by the backers according to Propositions 1 and 2. By doing so, the entrepreneur avoids excessive information disclosure that weakly shrinks revenue.

Initially, DSHS includes all the project status disclosures $s(k)$ since the last disclosure for backer $i$ into a set $H_i(t)$. That is,

$$H_i(t) = \{ s(k) \}_{k \in [k_0,k_0+1,\ldots,t]} \text{ s.t. } d(i,t') = s(k_0),t' \} \tag{4}$$

where $t' \in [1,2,\ldots,t]$. It then sorts $H_i(t)$ in the ascending order of $|s(k)|$. This sorting problem can be easily solved by Quicksort [17] with time complexity $O(|H_i(t)| \log |H_i(t)|)$. Since $|H_i(t)| \leq T$, the worst-case complexity for the function is $O(T \log T)$.

After the sorting process, DSHS removes the least promising candidates through the function SHRINK (see Algorithm 2). While there are at least two choices available, the SHRINK algorithm removes the project status with later time if the two statuses have the same progress (see line 4, Algorithm 2) according to Proposition 1. This process is equivalent to removing duplicates in a sorted array, which can be solved in $O(T)$ time. Given two project statuses, if they satisfy the relation in Proposition 2, then the algorithm removes the project status with the earlier time (see line 7, Algorithm 2). This step takes $O(T \log T)$ time in the worst case. The algorithm does nothing if only one disclosure strategy exists.

**Heuristic Selection.** After the shrinkage process, if there are still at least two choices available (i.e., $|H_i(t)| \geq 2$), then the remaining set $H_i(t)$ is horizontal. The entrepreneur has to select some $s(k_{sel}) \in H_i(t)$ to attract as many contributions as possible. This optimization problem is similar with the renowned restless bandit problem [44], which is not solvable due to incomplete information. However, simple heuristics such as random selection, greedy selection, ε-greedy exploration, and softmax exploration can be used to produce acceptable results. See Appendix B for details of each algorithm.

We further introduce a meta algorithm (see Algorithm 3). The intuition is that the algorithm can improve the quality of decisions by only using the experts that have a satisfying performance for producing the final results [14]. Besides, we take an ensemble approach to calculate the final selection instead of directly applying the results produced by the selected experts. The benefit is that the algorithm can further reduce potential performance loss due to biases of a single individual expert [22, 35].

Before describing the meta algorithm, we first introduce the notations used. Let $X$ denote the set of experts, where $x \in X$ is one of the four heuristics. We write $z^x(t)$ for expert $x$’s expected revenue at time $t$ and write $w^X(x)$ for expert $x$’s revenue prospect. Here, $z^x(t)$ is computed by: $z^x(t) = \sum_{s(k) \in H_t} \sum_{i \in I(t)} \rho^x_t(s(k)) \cdot Y_x(s(k),i,t)$, where $\rho^x_t(s(k))$ denotes the probability that $s(k)$ is selected as the targeted project status by expert $x$ at $t$, and $Y_x(s(k),i,t)$ is the entrepreneur’s expected increase of revenue given $(s(k),t)$ for backer $i \in I(t)$ by using expert algorithm $x$. Details of computing $Y_x(s(k),i,t)$ for each expert $x$ is described in Appendix B. Initially, $w^X = \max_{x \in X} z^X(x)$.

Each time the algorithm selects a subset $X'$ of experts whose expected revenue is higher than a learned threshold—the minimum learned prospect $w^X(x)$, this step eliminates the experts that fail to produce better expected revenue than the threshold. The algorithm then performs a majority vote from the results generated by each expert $x \in X'$. The selected project status $s(k_{sel})$ is the one with the most votes. Ties are broken by choosing the result generated by the expert with the highest $z^x(t)$. This step aims to improve the robustness of selection by reducing the performance loss caused by biases of a single expert.

![Algorithm 2 SHRINK](image-url)
When a new expert algorithm is selected, the prospect for the expert is updated by $w^t(x) = (1 - \sigma)q^t(x) + \sigma w^t-\delta(x)$. Here, $\delta$ is the number of periods that expert $x$ has been used, and $\sigma \in [0, 1]$ is the learning rate (in our paper), $q^t(x)$ is the entrepreneur’s average revenue gain per time by using expert algorithm $x$ in the last $\delta$ periods. It is calculated by: $q^t(x) = \sum_{t'=t-\delta}^{t} \sum_{j \in I(t')} \alpha_j(t') \sigma_j(t')$.

Algorithm 3 META

Input: $H'$-remaining status disclosures after shrinkage; $X$-the set of experts
Output: $s(k_{x,t})$-the selected project status disclosure

1: Compute $z^t(x)$ for $x \in X$
2: Initialize $w^t(x) = \max_{x \in X} z^t(x)$
3: while $t < T$ do
4: \hspace{1em} $X' = \{x : z^t(x) \geq \min w^t(x)\}$
5: \hspace{1em} Perform a majority vote for $s(k) \in H_{X'}$
6: \hspace{1em} Select $s(k_{x,t})$ as the $s(k)$ with the majority rule
7: \hspace{1em} Update $w^t(x) = (1 - \sigma)q^t(x) + \sigma w^t-\delta(x)$ if a new expert $x$ is selected
8: \hspace{1em} Update $q^t(x)$ and $z^t(x)$ for each $x \in X$
9: end while

$DSHS$ is highly flexible in the sense that it allows the entrepreneur to easily customize both the shrinkage process and the selection process with different methods.

**EMPIRICAL EVALUATION**

This section describes the experimental settings and the results.

**Experimental Setup.** We collected the campaign data from a randomly selected subset of Kickstarter projects using a web crawler. The dataset contains 1,569 projects which satisfy the following conditions: (1) they were all-or-nothing, reward-based campaigns; (2) the campaigns lasted for exactly 1,440 hours (60 days) with both the starting time and the ending time falling between 07/15/2016 and 10/15/2016. Each campaign includes hourly project status, the fundraising goal, the deadline, the minimal amount of contribution, and the number of contributions every hour. The data samples allow us to mimic the operation of real-world crowdfunding projects when the underlying factors and correlations that impact them are yet to be identified [3, 28].

Most of the projects on Kickstarter offer several tiers of perks for the backers to choose. We only selected the early bird pledges and the regular pledges that would offer a product to the backers to compensate backers’ financial support. We adjusted the funding goal for each project accordingly. The early bird pledges were proportioned to the regular pledges. For instance, an early bird pledge with value 8 is equivalent to 0.8 regular pledge with value 10.

Due to Kickstarter’s API constraints, we were unable to track the number of backers who visited the campaign per hour. We simulated backers’ arrivals using Poisson [33] distribution for each project. The arrival process was independently and identically distributed with a mean $\theta(t)$ across time, where $t \in T$. The mean of $\theta(t)$ is 0.1 (consistent with the empirical arrival rates of backers in crowdfunding projects [27]).

We used the anticipating random walk [3] model for simulating backers’ projections of PoS because it is tailored for computing backers’ estimates of PoS in crowdfunding. For each project, the backers’ valuation of a reward was Gaussian [33] distributed with a mean equivalent to the value of the reward $P$ and a randomly selected standard deviation ranging 0.05 of the mean to 0.5 of the mean.

We performed six groups of experiments: immediate disclosure (immediate), and $DSHS$ with five heuristics (random, greedy, $\epsilon$-greedy, softmax, and meta). Each group was run 30 times with the same 2.9GHz quad-core machine.

**Results.** Figure 2a shows the average revenue (normalized by the highest revenue achieved in all experiments) obtained in the end by each group. The actual revenue excluded the projects that failed ($M(T) < G$), while the expected revenue included all the projects regardless of whether they succeeded to meet the funding goal or not. Not surprisingly, the expected revenue of each group was significantly higher than their respective actual revenue. This is because the majority of the campaigns failed due to not having met the funding goal within the deadline (see Figure 2c). Among the six groups, the meta group scored the best for both the expected revenue (mean = 0.7435, std = 0.0244) and the actual revenue (mean = 0.3722, std = 0.0092), followed by the softmax group, and the $\epsilon$-greedy group. The greedy group and the immediate group performed better than the immediate group in the expected revenue, but not in the actual revenue due to a lower success rate. The random group received the lowest scores in terms of both the expected revenue (mean = 0.3210, std = 0.0273) and the actual revenue (mean = 0.1004, std = 0.0329).

At the beginning, the immediate, the meta, the greedy, and $\epsilon$-greedy groups performed better than the other two (see Figure 2b). As time progressed, the meta and the $\epsilon$-greedy groups continued to lead the way until the later left behind the former at around $t = 400$. The $\epsilon$-greedy group kept the second until at time $t = 900$ that it was surpassed by the softmax group. One explanation is that the $\epsilon$-greedy algorithm initially encouraged exploration to a higher degree than the softmax algorithm. However, it acted more greedily than the softmax exploration over time, which was not favorable since better choices were rarely explored. The meta group used a set of refined policies to produce more robust decisions than the others. The random group performed the worst possibly because the random algorithm completely ignored the history of backers’ responses.

The meta algorithm took the most computation time (mean = 0.2300 std = 0.0290 ), while the random method required the least time (mean = 0.1037, std = 0.0034) (see Figure 2d). Immediate disclosure required no additional time.

In summary, although the meta group required the most computation time, it performed consistently the best among all the groups in terms of both actual and expected revenue. This echoes our previous findings that immediate disclosure is not always optimal in crowdfunding.

https://www.kickstarter.com
Random
Random
Random
Random
(b) Expected revenue over time

\[ \epsilon \]
Greedy
Greedy
Greedy
Greedy

(c) Project success rate (%)
(d) Computation time

Figure 2: A comparison of performance.

CONCLUSION

In this paper, we present the very first study on information design where a sender interacts with multiple receivers that follow thresholding policies. Our work demonstrates that excessive information disclosure weakly shrinks the revenue in crowdfunding when backers use cutoff policies. It further shows that the widely-adopted immediate-disclosure policy is not optimal. We also present how the entrepreneur can benefit from dynamic information disclosure with appropriate heuristics. To further evaluate the performance of the heuristic algorithm, user studies and real-world deployment are needed.

Although our analysis is in the context of crowdfunding, extensions to other domains (e.g., transportation systems, smart grids, and online shopping) where agents typically use thresholding policies can be straightforward. For instance, online shopping marketplace can employ DSFS variants to dynamically reveal the number of products available or the number of products sold to attract potential buyers to buy the products. Further research is required to assess the variants’ performance in these domains.

ACKNOWLEDGMENTS

This work was supported by National Science Foundation through grant no. CCF-1526593. Yan was supported by National Science Foundation of China under grant no. 61602431. The authors wish to thank the anonymous reviewers for their helpful comments.

A PROOFS

Proof of Proposition 1

By condition, \( |s(k_1)| = |s(k_2)| \). That means the revenue does not increase between time \( k_1 \) to time \( k_2 \). Since \( k_1 < k_2 \), we have \( T - k_2 < T - k_1 \), which means less time is left to achieve the fundraising goal \( G \). Thus, the campaign’s PoS decreases or stays the same, regardless of backers’ arrivals from time \( k_1 \) to \( k_2 \). That is, \( r(t, (s(k_1), t)) \geq r(t, (s(k_2), t)) \).

By Equation 1, the utility of all the subsequently arriving backers weakly decreases, i.e., \( \forall t \geq k_2, \forall i \in I(t) : u_i((s(k_1), t)) \geq u_i((s(k_2), t)) \). By Definition 2, we have \( (s(k_1), t) \succ (s(k_2), t) \) for all \( t \geq k_2 \), and for all \( i \in I(t) \).

Proof of Proposition 2

By assumption \( \forall t \in T, b(t) \in [0, 1] \), we have that the number of arrivals from time \( k_1 \) to \( k_2 \) is: \( \sum_{t=k_1}^{k_2} b(t) \leq k_2 - k_1 \). Since \( |s(k_2)| - |s(k_1)| \geq (k_2 - k_1) \cdot PoS/G \), we have that at least \( k_2 - k_1 \) backers have contributed from time \( k_1 \) to \( k_2 \). Without loss of generality, one can assume that each time from \( k_1 \) to \( k_2 \), at least one backer arrives at the campaign and makes a contribution. In other words, the revenue of the campaign grows faster than or equal to the arrival of backers between \( k_1 \) and \( k_2 \). Thus, the campaign’s PoS increases or stays the same. That is, \( r(t, (s(k_2), t)) \geq r(t, (s(k_1), t)) \).

By Equation 1, the utility of all the subsequently arriving backers weakly increases, i.e., \( \forall t \geq k_2, \forall i \in I(t) : u_i((s(k_2), t_2)) \geq u_i((s(k_1), t_1)) \). By Definition 2, we have \( (s(k_1), t) \succ (s(k_2), t) \).

Proof of Lemma 1

By condition, we have \( e_2 \sim e_1 \). By relation of preferences and utility\[ u_{i}(e_2) \geq u_{i}(e_1) \]. According to backers’ utility function (Equation 1), \( r_{i}(t, e_2) \geq r_{i}(t, e_1) \). Depending on the order of the backer’s threshold \( \phi_i \), his belief \( r_{i}(t, e_1) \) when given report \( e_1 \) and the belief \( r_{i}(t, e_2) \) given report \( e_2 \), there are the following three cases:

- \( r_{i}(t, e_1) \leq r_{i}(t, e_2) < \phi_i \): in this case, backer \( i \) will not contribute to the campaign given either \( e_2 \) or \( e_1 \). That is, \( E(a_i = 1|e_2) = E(a_i = 1|e_1, e_2) = 0 \).
- \( r_{i}(t, e_1) \leq \phi_i \leq r_{i}(t, e_2) \): given report \( e_2 \), backer \( i \) will contribute to the campaign and leave the system. An additional signal about the project status cannot improve his possibility of pledging, i.e, \( E(a_i = 1|e_2) \geq E(a_i = 1|e_1, e_2) \).
- \( \phi_i < r_{i}(t, e_1) \leq r_{i}(t, e_2) \): under this condition, backer \( i \) will contribute to the campaign and leave the system given either of the two reports. That is, \( E(a_i = 1|e_2) = E(a_i = 1|e_1, e_2) = 1 \).

Proof of Lemma 2

We prove it by contradiction. Assume, to the contrary, that \( \exists t \geq \max\{k_1, k_2\}, j \in I(t) : r_j(t, (e_1, e_2)) > \max\{r_j(t, e_1), r_j(t, e_2)\} \). Depending on the order of backer \( j \)’s threshold \( \phi_j \), \( r_j(t, (e_1, e_2)) \), and \( \max\{r_j(t, e_1), r_j(t, e_2)\} \), we have the following three cases:

- \( r_j(t, (e_1, e_2)) > \max\{r_j(t, e_1), r_j(t, e_2)\} \geq \phi_j \): if \( r_j(t, e_1) > r_j(t, e_2) \), we have \( r_j(t, (e_1, e_2)) > r_j(t, e_2) \geq \phi_j \). By utility function (Equation 1), \( u_{j}(e_1, e_2) = u_{j}(e_2) = c_j \cdot \alpha_j(t) \). By the relation of preferences and utility \[ e_2 \sim e_1 \], where \( \sim \) denotes indifferent to. This contradicts that \( r_j(t, (e_1, e_2)) > r_j(t, e_2) \).
- \( \phi_j < r_j(t, (e_1, e_2)) > \max\{r_j(t, e_1), r_j(t, e_2)\} \): in this case, by Equation 1, \( u_{j}(e_1, e_2) = u_{j}(e_1) = u_{j}(e_2) = 0 \). By the relation of preferences and utility \[ e_2 \sim e_1 \]. This contradicts that \( r_j(t, (e_1, e_2)) > \max\{r_j(t, e_1), r_j(t, e_2)\} \).
- \( r_j(t, (e_1, e_2)) \geq \phi_j > \max\{r_j(t, e_1), r_j(t, e_2)\} \): by Equation 1, we have \( u_{j}(e_1, e_2) = c_j \cdot \alpha_j(t) \) and \( u_{j}(e_1) = u_{j}(e_2) = 0 \).
We prove it by contradiction. Assume, to the contrary, that:

\[ E(\alpha'_i) = 1 \geq E(\alpha''_i) = 1 \] by Lemma 1. If it is horizontal, \( r_i(t, (e_1, e_2)) = \max \{ r_i(t, (e_1)), r_i(t, (e_2)) \} \) by Lemma 2. Equation 1, there are three cases:

- \( \phi_1 \leq r_1(t, (e_1, e_2)) \leq \max \{ r_1(t, (e_1)), r_1(t, (e_2)) \} \): in this case, backer \( i \) will not contribute for both conditions, \( E(\alpha'_i) = 1 \).
- \( r_2(t, (e_1, e_2)) \leq \max \{ r_1(t, (e_1)), r_1(t, (e_2)) \} < \phi_1 \): in this case, backer \( i \) will not contribute for both conditions, \( E(\alpha''_i) = 1 \).
- \( r_1(t, (e_1, e_2)) < \phi_1 \leq \max \{ r_1(t, (e_1)), r_1(t, (e_2)) \} \): in this case, backer \( i \) will not contribute if given \( (e_1, e_2) \), i.e., \( E(\alpha''_i) = 0 \). If given either \( e_1 \) or \( e_2 \), backer \( i \) will either contribute or not contribute, i.e., \( E(\alpha'_i) = 0 \).

**Proof of Theorem 3**

\( \Rightarrow \) If the right side holds for all \( t' \leq t, \forall i \in I' \), then \( \forall t' \leq t, |s(t')| = |s(t' - 1)| \geq P/G \) such that \( DP_{im}(t) \) is optimal for some \( i, t' \). Without loss of generality, we let \( |s(t')| = |s(t' - 1)| + |\Delta s| \), where \( |\Delta s| < P/G \). Since \( |s(t')| = |s(t' - 1)| + \sum_{i \in I(t')} \alpha_i(t') \cdot P/G \), where \( \alpha_i \in [0, 1] \), we have \( |\Delta s| = \sum_{i \in I(t')} \alpha_i(t') \cdot P/G < P/G \). Therefore, \( |\Delta s| = 0 \), which indicates that \( |s(t')| = |s(t' - 1)| \). According to Proposition 1, \( \forall t' \geq t : (s(t' - 1), t) \geq (s(t'), t) \), which contradicts the supposition that \( DP_{im}(t) \) is optimal.

**Proof of Lemma 4**

We prove it by contradiction. Assume, to the contrary, that: \( DP_{im} \) is not optimal for some \( d(i, t') = (s(t'), t') \). This indicates that at time \( t' \leq t \), the entrepreneur could possibly profit by delaying information disclosure. Without loss of generality, we assume that \( d(i, t') = (s(k'), t') \) is the disclosure strategy used in the optimal disclosure policy for backer \( i \in I(t') \), where \( k' < t' \). Depending on the relation between \( t \) and \( k' \), there are two cases (since \( t' \geq t \)):

- \( i \leq k' < t' : \) at time \( t' \), the project has already succeeded, which means there is no uncertainty for backer \( i \) since the campaign’s PoS is 1. Therefore, delaying disclosure does not increase backer \( i \)’s estimate on PoS, which contradicts the proposition that \( d(i, t') = (s(t'), t') \) is not optimal.
- \( k' < i \leq t' : \) at time \( t' \), backer \( i \)’s estimate \( r_i(t', d(i, t')) \) is not lower than 1, while \( r_i(t, d(i, t')) = 1 \). That is, \( d(i, t') \geq d(i, t') \) (\( \forall i \in I(t') \)), which contradicts the proposition that \( DP_{im} \) is not optimal.

**B EXPERT ALGORITHMS**

**Random Selection**

The random algorithm picks the project status \( s(k_{sel}) \) at random from \( s(k) \in H_i(t) \) with equal probability. The expected increase of revenue \( Y(s(k), i, t) \) is computed by averaging the revenue when the entrepreneur disclosed the project status \( s(k) \).

**Greedy Selection**

The greedy selection algorithm chooses the project status \( s(k_{sel}) \) based on the empirical responses that the entrepreneur has received from the backers, using a one-step-look-ahead approach.

At time \( t \), given the disclosure strategy \( d(i, t) = (s(k), t) \), the entrepreneur establishes a historical belief \( Y_{old}(s(k), i, t) \) of the expected increase in the revenue, where \( i \in I(t) \), and \( s(k) \in H_i(t) \):

\[ Y_{old}(s(k), i, t) = \frac{1}{n_k(t)} \sum_{t=1}^{t-1} \sum_{i \in I} \frac{\alpha_i(t')}{\eta(t)/\eta(t)} \] where \( n_k(t) \) denotes the times that \( s(k) \) has been revealed to backers up to time \( t \). \( \alpha_i(t') \) is backer \( i \)’s action given disclosure strategy \( s(k), t' \). \( \eta(t') = |s(t')|/|t'| \) and \( \eta(t) = |s(t)|/|t| \) are the revenue growth rates up to time \( t' \) and \( t \), respectively. Note that \( Y_{old}(0) = 0 \) if \( n_k(t) = 0 \) or \( t = 1 \).

The historical belief represents the entrepreneur’s estimate of the average revenue she receives by revealing project status \( s(k) \) to the backers, with discounting of the revenue growth rates. It is a rough estimate of revenue increase that \( d(i, t) = (s(k), t) \) brings.

At time \( t \), the entrepreneur’s temporal belief \( Y_{tmp} \) of the expected revenue increase given the disclosure decision \( d(i, t) = (s(k), t) \), is determined as follows: \( Y_{tmp}(s(k), i, t) = \sum_{i \in I(t-1)} \frac{\alpha_{i}(t-1)}{|I(t-1)|} \) where \( I(t-1) \) denotes the set of backers in the campaign at time \( t-1 \) and \( \alpha_{i}(t-1) \) is backer \( i \)’s action at time \( t-1 \). If \( |I(t-1)| = 0 \) or \( t = 1 \), then \( Y_{tmp} = 0 \). The temporal belief captures the latest decisions of the backers that will most probably stay in the campaign at time \( t \).

The entrepreneur estimates the belief of the expected increase of revenue, given \( d(i, t) = (s(k), t) \) for backer \( i \in I(t) \), with the following equation:

\[ Y(s(k), i, t) = (1 - \lambda) Y_{old}(s(k), i, t) + \lambda Y_{tmp}(s(k), i, t) \]

where \( \lambda \in [0, 1] \) is the learning rate (we use \( \lambda = 0.1 \)).

The greedy algorithm then selects the project status \( s(k_{sel}) \) by using the following equation: \( s(k_{sel}) = \arg \max \{ Y(s(k), i, t) \} \).

The probability for selects the project status \( s(k_{sel}) \) is 1 and 0 for others.

**e-Greedy Exploration**

In e-greedy exploration, with probability \( e \) the algorithm selects a random choice \( s(k) \). Otherwise, with probability \( 1 - e \) it selects the greedy choice, given the greedy choice use the greedy selection. That is, \( Pr(s(k)) = \frac{e}{n_k(t)} \) if \( s(k) \) is a random selection; \( Pr(s(k)) = 1 - e + \frac{e}{n_k(t)} \) if \( s(k) \) is a greedy selection. Here, \( n_k(t) \) is the times that \( s(k) \) has been revealed to backers up to time \( t, e = c/n_k(t) \), and \( c \in [0, 1] \) is a constant.

Note that \( e = c = n_k(t) = 0 \).

**Softmax Exploration**

Softmax selects the choice using a Boltzmann distribution [33]. At time \( t \), the algorithm selects choice \( s(k) \) with the probability:

\[ \text{Prob}(s(k)) = \frac{e^{\frac{	ext{Val}(s(k))}{k_b}}}{\sum_{s(k) \in H_i(t)} e^{\frac{	ext{Val}(s(k))}{k_b}}} \]

where \( k_b = \max \mu, C^f / \log n_k(t) \) is the temperature parameter. Here, \( \mu = 0.0001, \tau = 1 \) when \( n_k(t) = 0 \), and \( C^f \) is determined by \( C^f = \max_{s(k)} P(s(k), s(k) \in H_i(t), \text{Val}(s(k)) = Y(s(k), i, t)) \). The softmax exploration selects each choice which has a probability that is proportional to the average \( Y \).