Supporting Online Material for

**Time-Critical Social Mobilization**

Galen Pickard, Wei Pan, Iyad Rahwan, Manuel Cebrian, Riley Crane, Anmol Madan, Alex Pentland*

*To whom correspondence should be addressed. E-mail: pentland@mit.edu

Published 28 October 2011, *Science* 334, 509 (2011)
DOI: 10.1126/science.1205869

This PDF file includes:

Materials and Methods
SOM Text
Figs. S1 to S5
Time Critical Social Mobilization: Supporting Online Material

Galen Pickard,¹,²† Wei Pan,¹† Iyad Rahwan,³,¹† Manuel Cebrian,¹,† Riley Crane,¹ Anmol Madan,¹ Alex Pentland¹,‡

¹Massachusetts Institute of Technology, USA
²Massachusetts Institute of Technology Lincoln Laboratory, USA
³Masdar Institute of Science & Technology, UAE

†These authors contributed equally to the paper.
‡Corresponding author; E-mail: pentland@mit.edu

September 16, 2011

1 Materials and Methods

We implemented the MIT mechanism by building a simple web platform (http://balloon.media.mit.edu). Any user could sign up with an E-mail address to become our team member and report a balloon on the web platform. During the registration and balloon reporting process, we recorded the time-stamp and IP address for each user activity on the platform. After registration, the user received a unique link in an E-mail from us, and the user could refer others by asking their friends to sign up with the MIT team via this link. The link allowed our system to record the referral relationships between users. It was up to the users to decide how to share this unique link. Some users chose to e-mail the link to their friends, but many others shared the link in online social media tools such as Twitter and Facebook. We initialized the diffusion process by sending the link to our web platform to a few close friends of the team members and several online web blogs 2 days before the competition.

Our analysis in Figure 1 and Figure 2 is based on the sign-up time information and referral chains collected from our web platform. We translated all user IP addresses into their locations using a 3rd-party online database smart-ip.net. Based on this location information, we were able to perform the analysis in Figure 3.
All the raw data including time stamps collected by our platform during the DARPA Challenge as well as the translated IP addresses are available as MySQL databases from the authors.

The raw Twitter dataset was collected in a Stanford research project by Yang et al. (31), and this project is unrelated to the DARPA Challenge. The Stanford researchers monitored and recorded the public Twitter feed from June, 2009 to December, 2009, and the crawled Twitter dataset contains 20-30% of all public tweets during their study period. The dataset had been available to public until recently when Twitter requested Stanford to withdraw from public access\(^1\). We used this Twitter dataset to scan for related tweets with keywords described in the main text, and conducted the analysis in Figure 4.

Other teams mentioned in our main text used different strategies to diffuse their involvement and to recruit their members: The GaTech team created their own website \(\text{http://www.ispyaredballoon.com}\), embedded links towards its website on Georgia Tech and Georgia Tech Research Institute websites, and performed search engine optimization to attract traffic. They also received a National Public Radio “Here and Now” broadcast interview four days before the challenge (17). George Hotz simply communicated his participation on his Twitter feeds to his 50,000 followers one day before the challenge. The Geocacher team sent E-mail alerts to all the members in the Geocacher community (roughly several hundred thousand members) one and two days prior to the balloon launch (17). They also set up a blog (\(\text{http://www.10ballonies.com}\)) to post their progress. DeciNena posted their information (\(\text{http://decinena.com}\)) in the comment section of every related DARPA Challenge blog post they could find online starting three days prior to the balloon launch\(^2\).

2 Formal Definition for the MIT Strategy

The MIT team approach was based on the idea that achieving large-scale mobilization towards a task requires diffusion of information about the tasks through social networks, as well as incentives for individuals to act, both towards the task and towards the recruitment of other individuals.

We consider our approach to the DARPA Network Challenge to be an instance of a more general class of mechanisms for distributed task execution. We define a diffusion-based task environment which consists of the following: \(N = \{\alpha_1, \ldots, \alpha_n\}\) is a set of agents; \(E \subseteq N \times N\) is a set of edges characterizing social relationships between agents; \(\Psi = \{\psi_1, \ldots, \psi_m\}\) is a set of tasks; \(P: N \times \Psi \rightarrow [0, 1]\) returns the success probability of a given agent in executing a given task; \(B \in \mathbb{R}\) be the budget that can be spent by the mechanism.

\(^1\)See \(\text{http://snap.stanford.edu/data/twitter7.html}\).

\(^2\)We estimated their start time according to the information on their Facebook post: \(\text{http://www.facebook.com/pages/Team-DeciNena/192350616581}\).
In a diffusion-based task environment, unlike in traditional task allocation mechanisms (e.g. based on auctions), agents are not aware of the tasks a priori. Instead, they become aware of tasks as a result of either 1) being directly informed by the mechanism through advertising; or 2) being informed through recruitment by an acquaintance agent (20).

Another characteristic of diffusion-based task environments is that, when a task is completed, the mechanism is able to identify not only the agent who executed it, but also the information pathway that led to that agent learning about the task. The pathway leading to the successful completion of task $\psi_i$ is captured by the sequence $S(\psi_i) = \langle a_1,\ldots,a_r \rangle$ of unique agents, where $a_r$ is the agent who completed the task, $a_r$ was informed of the task by $a_{r-1}$ and so on up to agent $a_1$ who was initially informed of the task by the mechanism. By slightly overloading notation, let $|S(\psi_i)|$ denote the length of the sequence (i.e. the number of agents in the chain), and let $\alpha_j \in S(\psi_i)$ denote that agent $\alpha_j$ appears in sequence $S(\psi_i)$.

We can now define a class of mechanisms that operate in the above settings. A diffusion-based task execution mechanism specifies the following: $I \subseteq N$ is a set of initial nodes to target (e.g. via advertising); $\rho_i$ is the payment made to agent $\alpha_i$; such that the following constraint is satisfied: $c|I| + \sum_{\alpha_i \in N} \rho_i \leq B$.

In words, the mechanism makes two decisions. First, it decides which nodes to target initially via advertising. Second, it decides on the payment (if any) to be made each agent. The mechanism must do this within its budget $B$.

In the DARPA Network Challenge, each $\psi_i$ represents finding a balloon, and $v(\psi_i) = 4,000$ for all $\psi_i \in \Psi$. Moreover, we assume that the ten tasks are all identical (namely finding a balloon), and all tasks are indistinguishable, $\forall \alpha_i \in N, \forall \psi_k, \psi_l \in \Psi$ we have $P(\alpha_i, \psi_k) = P(\alpha_i, \psi_l)$. That is, the success probability of a particular agent is the same for all balloons.

We are now ready to define our mechanism, referred to as a recursive incentive mechanism. Given $I$ initial targets, and assuming $v(\psi_i) = B/|\Psi|$, divide the budget $B$ such that each task $\psi_i \in \Psi$ has budget $B_i = B/|\Psi|$. If agent $j \in N$ appears in position $k$ in sequence $S(\psi_i)$, then $j$ receives the following payment:

$$\frac{v(\psi_i)}{2(|S(\psi_i)| - k + 1)}$$

Hence, the total payment received by agent $j$ is the sum of payments for all sequences in which $j$ appears:

$$\rho_j = \sum_{\psi_i,j \in S(\psi_i)} \frac{v(\psi_i)}{2(|S(\psi_i)| - k + 1)}$$

The surplus is therefore: $S = B - \sum_{\alpha_j \in N} \rho_j$. Figure S1 illustrates how this mechanism works.
(a) Example social network. (b) Recruitment tree with two paths (shown in thick lines) initiated by $\alpha_1$ led to finding balloons.

**Figure S1:** Recursive incentive mechanism: (a) Suppose that in this network, agent $\alpha_1$ recruits all of his neighbors, namely $\alpha_2$, $\alpha_5$ and $\alpha_8$. Suppose that $\alpha_8$ recruits $\alpha_6$, who finds balloon $\psi_1$. (b) We have a winning sequence $S(\psi_1) = \langle \alpha_1, \alpha_8, \alpha_6 \rangle$ with $|S(\psi_1)| = 3$. The finder receives $\rho_8 = \frac{4,000}{2^{(3-2+1)}} = 2,000$. Since $\alpha_8$ recruited $\alpha_6$, then $\rho_8 = \frac{4,000}{2^{(3-2+1)}} = 1,000$. From this sequence, $\alpha_1$ receives $\frac{4,000}{2^{(3-1+1)}} = 500$. Likewise, looking at the left recruitment path, we have a winning sequence $S(\psi_2) = \langle \alpha_1, \alpha_2, \alpha_3, \alpha_4 \rangle$ with $|S(\psi_2)| = 4$. The finder receives $\rho_4 = \frac{4,000}{2^{(4-2+1)}} = 2,000$. As above, we have $\rho_3 = \frac{4,000}{2^{(4-2+1)}} = 1,000$ and $\rho_2 = \frac{4,000}{2^{(4-2+1)}} = 500$. From this sequence, $\alpha_1$ receives $\frac{4,000}{2^{(4-1+1)}} = 250$. Adding up its payments from the two sequences it initiated, $\alpha_1$ receives a total payment of $\rho_1 = 750$. Assuming there are only two tasks, the surplus in this case is $S = (4,000 - 3,500) + (4,000 - 3,750) = 750$. 
3 Formal Proofs

3.1 Mechanism Always within Budget

Proposition 1. The recursive incentive mechanism is never in deficit (i.e. never exceeds its budget).

Proof. Recall that each sub-task $\psi_i$ is allocated an equal share of $B_i = B/|\Psi|$ budget. Hence, it suffices to show that the payment for any arbitrary task $\psi_i$ is bounded by $B_i$. Let $S(\psi_i) = \langle a_1, \ldots, a_r \rangle$ be the (finite) sequence leading to the successful completion of $\psi_i$. We need to show that the total payment made to all agents in sequence $S(\psi_i)$ within budget, that is, we need to show that:

$$\sum_{k=1}^{r} \frac{B_i}{2^{(r-k+1)}} \leq B_i \quad \text{or equivalently we need to show that} \quad \sum_{k=1}^{r} \frac{1}{2^{(r-k+1)}} \leq 1$$

We can easily see that:

$$\sum_{k=1}^{r} \frac{1}{2^{(r-k+1)}} = \sum_{k=1}^{r} \left(\frac{1}{2}\right)^{(r-k+1)} = \sum_{k=1}^{r} \frac{1}{2^{r-k+1}} \times \left(\frac{1}{2}\right)^{(r-k)}$$

Defining $i = r - k$, we can rewrite:

$$\sum_{k=1}^{r} \frac{1}{2^{(r-k+1)}} = \sum_{i=0}^{r-1} 0.5 \left(\frac{1}{2}\right)^i$$

This is a finite geometric series, with a well-known closed form:

$$\sum_{i=0}^{r-1} 0.5 \left(\frac{1}{2}\right)^i = 0.5 \frac{1 - \left(\frac{1}{2}\right)^{(r-1)+1}}{1 - \frac{1}{2}} = 1 - \left(\frac{1}{2}\right)^r = \frac{2^r - 1}{2^r} \leq 1 \quad \text{(for} \ r \geq 1)$$

We continue to discuss our theoretical analysis on the mechanism. When agent $\alpha_i$ becomes aware of a set of tasks $\{\psi_1, \ldots, \psi_m\}$, it needs to select a (possibly empty) set of neighbors $T(\alpha_i) \subseteq \{\alpha_j \in N : (\alpha_i, \alpha_j) \in E\}$ to recruit (i.e. to inform them about $\psi$), assuming the team will win the challenge. The diffusion of information about the task relies crucially on such recruitment choices among agents. One can perform this incentive analysis under two different assumptions:
3.2 Incentives With Uniform Independent Success Probability Among All Population

We now discuss the properties of our mechanism under the assumption that the probability of each person finding a balloon is an independent (and very small) constant, \( \forall i, k, P(\alpha_i, \psi_k) = \epsilon \), such that \( n \epsilon \leq 1 \), i.e. the sum of these probabilities over the entire population (including those not recruited) is bounded by 1. In this case, it is trivial to show that recruiting all of one’s peers is the best strategy. Without recruiting, one achieves an expected reward of \( \sum_i \epsilon v(\psi_i) \). With recruiting, on the other hand, one’s expected rewards is \( \sum_i \left( \epsilon^2 v(\psi_i) + \sum_j \epsilon^j x_j v(\psi_i) \right) \), where \( x_j \) is the number of individuals at depth \( j \) of the recruiter’s tree. Clearly, this expected reward increases monotonically in the number of directly recruited nodes.

3.3 Incentives With Uniform Success Probability Among Recruited Individuals

We can also analyze incentives under the assumption that the probability of an individual finding a balloon is uniformly distributed across the recruited individuals. Formally we have:

**Definition 1 (Uniformity).** Each recruited individual is equally likely to accomplish task \( \psi \), and the probability of each individual accomplishing the task is \( \frac{1}{n} \), where \( n \) is the number of all recruited individuals.

Intuitively, it means that a fixed-size group of recruited individuals is guaranteed to find the balloon eventually, even if no other individuals are recruited. This assumption is realistic if the set of recruited individuals is sufficiently large (e.g. thousands). We will continue to show that, under fairly broad assumptions on the structure of the society, it is also in the best interest of each individual to recruit all their friends. In particular, here we show that if no individual controls \( n/3 \) of the population (i.e. is able to prevent them from learning about the task), then the strategy profile in which all individuals recruit all their friends is a Nash equilibrium. In the following results, we consider the situation that there is only one task \( \psi \) that is being diffused in the social network with a total budget of 1.

3.3.1 All-or-None Recruitment on Fixed-Forest Social Networks

We consider the case in which the social network takes the form of a forest of rooted trees, and the roots of these trees form the set of initially-recruited nodes \( I \).

Given this forest \( F \), which contains a total of \( n \) nodes, each node chooses whether or not to recruit all of its children. This induces a “recruited subforest” \( F' \) of size \( n' \), consisting of all nodes which can trace a path of recruitment to a root node of \( F \).
3.3.2 Game Definition

We demonstrate the definition of the game by example, recreating the “prisoner’s dilemma” using a 5-node forest under the uniformity assumption.

Consider the forest $F$ shown in Figure S3. There are two players, $R_1$ and $R_2$, each of which has the option to recruit a single child or not. If neither recruits, both receive an expected payment of $\frac{1}{3}$. If one recruits but the other does not, the recruiter has an expected payment of $\frac{1+\frac{1}{2}}{4} = \frac{3}{8}$, while the other has an expected payment of $\frac{1}{4}$. If both recruit, both have an expected payment of $\frac{1+\frac{3}{4}}{5} = \frac{3}{10}$. This gives a payment matrix approximated by:

<table>
<thead>
<tr>
<th></th>
<th>$N$</th>
<th>$Y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>.33, .33</td>
<td>.25, .37</td>
</tr>
<tr>
<td>$Y$</td>
<td>.37, .25</td>
<td>.3, .3</td>
</tr>
</tbody>
</table>

Figure S2: Nodes $R_1$, $b_1$, and $a_2$ choose to recruit; the rest not. The recruited subforest $F'$ is shown in red. Note that $a_2$’s choice to recruit is rendered moot by $R_2$’s choice not to recruit.
Figure S3: The game played by $R_1$ and $R_2$ is equivalent to the “prisoner’s dilemma.”

Clearly, choosing to recruit is a strictly dominant strategy for each player, so the only Nash equilibrium that both players recruit – even though this is Pareto inefficient.

### 3.3.3 Nash Equilibrium of Larger Forests

Consider a game in which all actors have two options: “recruit all” or “recruit none.” For any given agent $a$, let all other agents choose “recruit all,” and consider $a$’s optimal strategy. If choosing “recruit all” is optimal for $a$, then no agent can benefit by deviating from a strategy of “recruit all,” if all other agents choose “recruit all.” This, by definition, makes the uniform choice to “recruit all” a Nash equilibrium.

**Theorem 1.** A node $a$ will prefer recruitment to non-recruitment predicated on all other nodes choosing to recruit if and only if sufficiently many nodes in the forest $F$ are not descendants of $a$ under the uniformity assumption.

**Proof.** For a node $a$ in a forest $F$ of size $n$, let the tuple $X = \{x_1, x_2, x_3, \ldots\}$ be defined as the number of children, grand-children, great-grand-children, etc. of node $a$. If $F$ is finite, each $x_i$ is finite and there exists some $j$ such that $x_i = 0$ for all $i > j$. Let $k$ be the number of nodes in $F$ that are not descendants of $a$, noting that $k = n - \sum i x_i$. Since we assume all nodes other than $a$ choose to recruit, the expected payment received by $a$ if $a$ chooses to recruit is $\frac{1}{k}$. If $a$ does choose to recruit, then $a$ will receive expected payment $\frac{1 + \sum i x_i}{n}$. Therefore, we have that $a$ will choose to recruit predicated on all other nodes recruiting if and only if $\frac{1 + \sum i x_i}{n} > \frac{1}{k}$, or, equivalently, when $k > \frac{\sum i x_i}{\sum i x_i}$. \hfill $\Box$

**Corollary 1.** In any forest $F$ of size $n$ for which no tree contains more than $\frac{n}{3}$ nodes, all nodes choosing to recruit is a Nash equilibrium under the uniformity assumption.

**Proof.** Consider forest $F$ with $n$ nodes, and a node $a$ which has $m$ descendants, taking a shape described by a tuple $X = \{x_1, x_2, x_3, \ldots\}$. The expected payment for $a$ for not recruiting is $\frac{1}{n-m}$, and the expected payment for $a$ for recruiting is $\frac{1 + \sum i x_i}{n}$. Therefore, we have that $a$ will choose to recruit predicated on all other nodes recruiting if and
only if \( n - m > \frac{m}{\sum_i \frac{x_i}{2}} \). We note that the definition of \( X \) yields that no non-zero value can follow a zero value (i.e. one must have grand-children in order to have great-grand-children). It follows that, if we fix \( m \), the setting of \( X \) which maximizes \( \frac{m}{\sum_i \frac{x_i}{2}} \) is \( X = \{1, 1, \ldots, 1, 0, 0, \ldots\} \), which gives \( \frac{1}{2} \leq \sum_i \frac{x_i}{2} < 1 \) for \( m > 0 \). Thus, \( a \) will choose to recruit if (but not only if) \( n - m > 2m \). This condition holds for all nodes if and only if no tree in \( F \) contains more than \( \frac{2}{3} \) nodes. In this case, all nodes will choose to recruit predicated on all other nodes recruiting, so all nodes choosing to recruit is a Nash equilibrium.

\[\square\]

### 3.3.4 Selective Recruitment on Fixed-Forest Networks

We now consider the same social graph structure, but allow a node to selectively recruit any subset of its children.\(^3\)

**Definition 2 (Weight).** We define the weight of a node \( a \), \( W_a \), as the sum of the rewards that would be received by \( a \) in the event that each of its descendants were to complete the task. We note the following properties of \( W_a \):

- If \( a \) is a leaf, then \( W_a = 1 \).
- If \( a \) has children \( c_1, c_2, \ldots \) with weights \( W_{c_1}, W_{c_2}, \ldots \), then \( W_a = 1 + \frac{1}{2} \sum_i W_{c_i} \).
- If node \( a \) has descendants described by shape \( X = \langle x_1, x_2, \ldots, 0 \rangle \), then \( W_a = 1 + \sum_i \frac{x_i}{2} \).
- In a forest with \( n \) nodes, the expected payment to node \( a \) is \( U(a) = \frac{W_a}{n} \).

**Lemma 1.** Assuming all other nodes recruit all their children, a node \( a \) with children \( C = \{c_1, \ldots, c_m\} \) recruits a child \( c_x \) regardless of the shape of \( F_{c_x} \) of the other children \( c_i \neq c_x \), if and only if the weight of \( c_x \) is large relative to the number of its descendants:

\[
\frac{1}{2} \frac{W_{c_x}}{k + m - 1} \leq \frac{1}{2} \frac{W_{c_x}}{|c_x|}
\]

**Proof.** Let \( k > 2 \) be the number of nodes in the forest that are not descendants of \( a \). The marginal benefit of recruiting a child \( c_x \) of node \( a \) depends on the other children \( a \) recruits. Assuming \( a \) recruits a subset of its children \( S \subset (C \setminus \{c_x\}) \) and does not recruit any other children \( C \setminus (S \cup \{c_x\}) \), child \( c_x \) is recruited if and only if

\[
\frac{1 + \frac{1}{2} \sum_{c_i \in S} W_{c_i}}{k + (\sum_{c_i \in S} |c_i|)} \leq \frac{1 + \frac{1}{2} \sum_{c_i \in S} W_{c_i} + \frac{1}{2} W_{c_x}}{k + (\sum_{c_i \in S} |c_i|) + |c_x|}
\]

\(^3\)We are grateful to Victor Naroditskiy for comments that helped refine the results in this section.
This inequality is equivalent to
\[
\frac{1 + \frac{1}{2} \sum_{c_i \in S} W_{c_i}}{k + \left( \sum_{c_i \in S} |c_i| \right)} \leq \frac{1}{2} \frac{W_{c_x}}{|c_x|}
\]
We are looking for a condition that guarantees \( c_x \) is recruited for any subset of \( S \) and shapes of trees \( F_{c_i} \) rooted at children other than \( c_x \). Thus, we want the left-hand side to be as high as possible. For \( k > 2 \), the left-hand side is maximized when \( S = C \setminus \{c_x\} \) and each child \( c_i \in S \) has no descendants. This worst-case value is
\[
\frac{1 + \frac{1}{2} (m - 1)}{k + (m - 1)} = \frac{\frac{1}{2} (m + 1)}{k + m - 1}
\]

**Corollary 2.** Node \( a \) recruits all of its children if Lemma 1 holds for each child.

**Theorem 2.** Assuming all other nodes recruit all their children, a node \( a \) recruits all of its children regardless of the shape of \( F_a \) if sufficiently many nodes are not \( a \)'s descendants: \( k > |a|^2 \)

**Proof.** The condition \( k > |a|^2 \) is equivalent to \( \frac{1}{|a|} > \frac{|a|}{k} \). Since \( |c_x| \leq |a| \), we have \( \frac{1}{|c_x|} > \frac{|a|}{k} \).

For \( a \geq m + 1 \) (for the case \( a = m \), all children of \( a \) have no descendants, and it is easy to see that they are all recruited if \( k > 2 \)), we obtain \( \frac{1}{|c_x|} > \frac{m+1}{k} \). Trivially, \( W_{c_x} > 1 \), and thus \( \frac{W_{c_x}}{|c_x|} > \frac{m+1}{k} > \frac{m+1}{k+m-1} \) and by Lemma 1 \( c_x \) should be recruited. \( \square \)

### 3.3.5 Recruitment on Graphs

We consider now the case in which the social graph is not a forest, but is instead a general graph. In this case, the mechanism of recruitment itself plays a non-trivial role, since it is possible for a node to be recruited by two different potential parents, and must choose between them. There is significant literature on diffusion processes on graphs, and wide varieties of such processes are seen in practice. We will not investigate the properties of specific diffusion mechanisms, but instead we will define a property of a diffusion mechanism that guarantees that recruitment is Nash.

**Definition 3 (Monotonic Diffusion).** Consider a diffusion process on a social graph, and a set of seed nodes \( R_1, R_2, \ldots, R_n \). Let \(|R_1|, |R_2|, \ldots, |R_n|\) be the number of nodes whose recruitment leads back to \( R_1, R_2, \ldots, R_n \), respectively. We call the diffusion process monotonic if removing a seed node \( R_x \) causes the sizes of \(|R_1|, |R_2|, \ldots, |R_{x-1}|, |R_{x+1}|, \ldots, |R_n|\) to either increase or stay constant (i.e. if \( R_x \) does not participate, this does not cause another seed node to recruit fewer children).
Monotonicity holds for most “well-behaved” diffusion processes, but is notably violated by various “complex contagion” processes in which, for example, a node adopts after receiving two signals.

**Theorem 3.** Under the monotonic diffusion assumption, if each node can recruit less than $\sqrt{k}$ nodes, where $k$ is the number of all recruited nodes that are not descendant of this node when all nodes recruit, then all nodes recruiting is a Nash equilibrium.

**Proof.** Consider a node $a$, which can choose whether or not to recruit, and suppose all other nodes recruit. Suppose it were the case that if $a$ were to not choose recruitment, then all nodes that would have been recruited by $a$ would end up un-recruited, instead. In this case, the graph reduces to the same fixed forest we analyzed previously. Suppose instead that some of these nodes end up recruited by a different node. In this case, not recruiting is strictly less desirable, since the size of the network grows without any increase in potential payout. Hence, it follows Theorem 2 that recruiting all children is more desirable in either case under the assumption that $a$ cannot recruit more than $\sqrt{k}$. If diffusion is monotonic, the two cases considered are collectively exhaustive, so recruiting all children is always the more desirable option.

3.4 Summary

The two assumptions above differ in their treatment of how the addition of a new member to the network affects the probability that each other member succeeds in finding a balloon. The first assumption is that new members have no effect on existing members, perhaps because they are searching mutually exclusive areas, and if the new member were to find a balloon, that implies that no-one would have found that balloon in his absence. The second assumption is that each member’s probability decreases from $\frac{1}{n}$ to $\frac{1}{n+1}$, perhaps because all members share the same search space. Unless “network effects” are present (e.g. working with a new member makes us both more likely to succeed than working alone), these two assumptions represent best- and worst-case assumptions. There are certainly intermediate assumptions that could be made, and the strategies that motivate diffusion at both extremes will do so for these intermediate assumptions as well.

4 Comparison with Query Incentive Networks

We consider how our scheme relates to Kleinberg and Raghavan pioneering work on “Query Incentive Networks” (QINs) (21). A QIN is a network of agents which one agent seeks information which is known to some subset of other agents. This agent broadcasts to all of his neighbors that he is offering a reward of value $r^*$ in exchange for the information. Assuming that none of his neighbors have the information themselves, each can broadcast
to its own neighbors that it is offering a lesser reward (e.g. $r^* - 1$) for the same information. If it receives the information from any of its neighbors, it pays out $r^* - 1$ and receives $r^*$ from the root node, making a profit of 1 by acting as a conduit for the information.

Kleinberg et al. analyze the properties of these networks, and show that they are an efficient method for retrieving uncommon information in a network of agents. QINs are very similar in spirit to our recursive incentives, and we show how recursive incentives can be created through a modification to the QIN strategy. In addition, we argue that recursive incentives effectively solve two issues that would make practical implementation of a QIN difficult in a time-critical situation.

When comparing recursive incentives and QINs, two differences are paramount. First, in a recursive incentive network, upon receipt of the information, the root node directly pays each node on the path to the agent who supplied the information, while in a QIN, the root node makes a single payment to one of its neighbors, who then makes a smaller payment to one of its neighbors, and so on. In the construction of the recursive incentive network, then, is the implicit assumption that any node can communicate (and transfer payment) to any other, but that the propagation of queries can only follow a limited number of links (i.e. the network structure). This assumption is a natural model of real social networks (especially on-line social networks like Facebook or Twitter) in which it is possible to explicitly communicate with any user by name, but the default “broadcast” mechanism only reaches a limited, socially-defined subset of users. A second difference is that in a QIN, intermediate nodes on the chain to the information supplier receive a fixed reward, and the node at the end of the chain receives an amount that decreases as its distance from the root increases. By contrast, in a recursive incentive network, the node that supplies the information receives a fixed reward, while intermediate nodes receive a variable reward depending on their distance from the information supplier.

4.1 Making Recursive Incentives out of Query Incentive Networks

We borrow terminology from Kleinberg et al. to show how to transform the QIN framework into a recursive incentive. We say that the root node offers a “contract” of value $r^*$ for a piece of information, effectively promising “I will pay $r^*$ to the first node to provide me with this information.” Other nodes that do not themselves value the information offer “subcontracts” of value less than $r^*$, hoping to receive reward by acting as an intermediary. Consider if the root node adds the following to the contract: “In addition to paying $r^*$ for the information, if you obtain this information though subcontracts and your subcontracts are identical to this contract, I will reimburse you for half of the costs you pay.” This results in a payment scheme that is identical to a recursive incentive which pays the final node $r^*$, as we will show.

The base case is the situation in which a neighbor ($n_1$) of the root has the information. The root node pays $r^*$ to $n_1$. If $n_1$ does not have the information, but offers the same
contract to its neighbor \((n_2)\), who does have the information, \(n_1\) pays \(r^*\) to \(n_2\). The root then pays \(r^*\) to \(n_1\) for fulfilling the original contract, and an additional \(\frac{r^*}{2}\) to reimburse half of the costs paid by \(n_1\). Consider the case in which \(n_2\) does not have the information, but offers the same contract to \(n_3\), who does. Working from the leaf node up to the root:

- \(n_2\) pays \(r^*\) to \(n_3\) for fulfilling the contract. \(n_3\) makes no payments, so receives a net of \(r^*\).
- \(n_1\) pays \(r^*\) to \(n_2\) for fulfilling the contract, and also pays \(\frac{r^*}{2}\) to \(n_2\), since \(n_3\) paid out \(r^*\) in expenses. \(n_2\) received \(\frac{3r^*}{2}\) and paid out \(r^*\), for a net profit of \(\frac{r^*}{2}\).
- The root pays \(r^*\) to \(n_1\) for fulfilling the contract, and also pays \(\frac{3r^*}{4}\) to \(n_1\), since \(n_1\) paid out a total of \(3r^*\). \(n_2\) received \(\frac{7r^*}{4}\) and paid out \(\frac{3r^*}{2}\), for a net profit of \(\frac{r^*}{4}\).

Having now explicitly demonstrated how small chains produce the same payouts as recursive incentives, let us now consider the case where the chain is of length \(k\):

- \(n_{k-1}\) pays \(r^*\) to \(n_k\) for fulfilling the contract. \(n_k\) makes no payments, so receives a net of \(r^*\).
- For \(i\) between 1 and \(k-1\), inclusive, define \(p_1\) as the total amount paid out by \(n_i\). Note that the net profit of node \(n_i\) is \(p_{i+1} - p_i\).
- To fulfill all aspects of the contract, \(n_i\) pays \(\frac{p_{i+1}}{2} + r^*\) to \(n_{i+1}\), so \(p_i = \frac{p_{i+1}}{2} + r^*\).
- Letting \(p_{k-1} = r^*\), \(p_{k-i} = \frac{r^*2^{i-1}}{2^{i-1}}\) is the unique solution to this recursive definition of \(p_i\), which leads to node \(n_i\) making net profit \(\frac{r^*}{2^{i-1}}\).

### 4.2 Practical Considerations

Having shown that recursive incentives can be presented as a modification to the concept of QINs, we argue that the recursive incentive formulation has significant benefits when implemented in practice. Consider a traditional QIN operating over a real-world online social network like Facebook or Twitter. When a node sees a contract and wants to offer a subcontract, it has two practical difficulties. First, it cannot allow any of its subcontractors to know about its original (and more valuable) contract, since any would-be subcontractor who knows about the original would surely prefer to fulfill it instead. Also, it must distribute the nature of the request and establish payment infrastructure (including building the trust in the subcontractors that payment will happen). Since it cannot use the original contract as a reference, it is effectively starting from scratch, which significantly increases the cost of initiating the subcontract. By contrast, propagation in a recursive incentive network involves simply passing a link to the original "contract," allowing the
root node to bear all the burden of fully explaining the request and establishing payment infrastructure.

A second real-world issue faced by a QIN (especially in a time-critical situation) occurs when communication between nodes incurs a time delay or is lossy. In a QIN, the information in its entirety must pass through a series of intermediaries. If any of these corrupts the information or significantly delays propagation (e.g., goes offline), the information could potentially not reach the root node in a timely and correct fashion. By contrast, a recursive incentive network allows the node that has the information to pass it directly to the root node, bypassing all intermediaries (while still allowing them to be paid by the root). Again, this assumes “broadcast” communication is limited by social connection, but that point-to-point communication is possible between any two nodes in the network – but this assumption underlies the core dynamics of most online social networks.

5 Quantifying the “MIT” Effect

In the paper, we argued that the “MIT brand” did not play a major role in the success of the MIT team. In particular, we showed that the burst of tweets about the MIT team was more sustained compared to other strategies, including those based on celebrity following, which experienced very short-lived bursts. We attributed this qualitative difference to the MIT incentive mechanism.

To further support this claim, bearing in mind the limited data available, we compared the MIT red balloon team’s tweet count with another MIT-related event, namely launching a hybrid electrical bicycle. This event took place in the same month, received significant mass media attention, and was the only MIT news exceeding 50 tweets in December in our Twitter dataset. While it is difficult to conduct a systematic comparison, Figure S4 suggests that this event also sustained a short-lived burst even with major media coverage, while the MIT team achieved sustaining burst with our mechanism.

\[\text{\footnotesize 4See http://senseable.mit.edu/copenhagenwheel/}.\]

\[\text{\footnotesize 5E.g. see http://www.nytimes.com/2009/12/15/science/earth/15bike.html} \]
Figure S4: Tweet count over time of MIT red balloon team versus another MIT-related event during the same month. For easy comparison, we shifted data temporally by matching the day when the MIT team launched the online campaign with the day when the MIT bike news was released: (a) Daily Twitter counts for both events (Data are scaled in this figure so that both peaks have the same value); (b) Raw daily increase in Twitter counts. The vertical blue dash line indicates the day of the DARPA Challenge competition.

6 Examples of recruitment trees
Figure S5: (a) A tree with the root is shown in green, and the successful path highlighted in red. (b) and (c) Two additional networks that did not lead to balloons.