Maximizing the Relevant Diversity of Social Swarming Information

Peter Terlecky[1], Yurong Jiang[2], Xing Xu[2], Amotz Bar-Noy[1] and Ramesh Govindan[2]
[1] *CUNY Graduate Center, New York, NY, USA*
[2] *University of Southern California, Los Angeles, CA, U.S.A.*

Keywords: Information Diversity, Social Swarming, Maximum Coverage.

Abstract: In social swarming applications, users are equipped with smartphones and generate data on specific tasks in the form pictures, video, audio, text. A central commander would like to gain access to data relevant to a particular query. Which data wirelessly uploaded to the commander maximizes the amount of diverse information received subject to a bandwidth constraint? We model such a problem in two distinct ways. It is first modeled as a maximum coverage with group budget constraints problem and then as a variant of the maximum edge-weighted clique problem. It is shown that the algorithm for the maximum coverage model outperforms a heuristic for the clique-based model theoretically and practically, with both performing very well experimentally compared to an upper bound benchmark.

1 INTRODUCTION

In social swarming applications, users are equipped with smartphones with which they generate data on specific tasks. The data is in the form of text, video, audio, and pictures. A central commander would like to wirelessly receive information relevant to a query from each of the users that is collectively comprehensive.

Furthermore, he does not wish the users flood the network with all of their data, so he requires each user to send a limited amount of the relevant collected data. The commander would like to avoid duplicate data or data that is similar in nature. That is, the commander would like to maximize the dissimilarity of the received data set, but have the data set be relevant to the query/task. Which data should be uploaded from which users to maximize the diversity of information while maintaining bandwidth constraints?

One way in which we model such a problem is as a maximum coverage with group budgets problem. Chekuri and Kumar (Chekuri and Kumar, 2004) introduce and analyze the maximum coverage with group budgets problem. They give a 2-approximation for the cardinality version of the problem and a 12-approximation for the cost version.

We also be model such a problem as a variant of the maximum edge-weighted clique problem. The maximum edge-weighted clique problem was studied in(Dijkhuizen and Faigle, 1993; Macambira and De Souza, 2000; Hunting et al., 2001; Park et al., 1996).

Receiving a diverse set of high quality information is important in many applications. Consider a shopper is deciding on whether to buy a product. The shopper would like to receive reviews on a product. The reviews may be grouped by category or rating. In particular, the shopper would like to receive a diverse subset of high quality reviews avoiding duplicate or similar reviews. Which subset of reviews should be shown to the shopper? Tsaparas et al. (Tsaparas et al., 2011) focus on the problem of selecting a comprehensive set of high-quality reviews covering many different aspects of the reviewed problem. They model the problem as a maximum coverage problem and provide algorithms which they test on a user study of the Amazon Mechanical Turk. Lappas et al. (Lappas et al., 2012) seek to select a subset of reviews that maintain the statistical properties of the review corpus. Yu et al. (Yu et al., 2013) seek to select a small subset of high quality reviews which are opinion diversified and cover a large set of attributes. Zhuang et al (Zhuang et al., 2006) consider the problem of extracting the features on which movie reviewers express their opinions and determining whether the opinions are positive or negative. They propose an effective multi-knowledge based solution.

Chen et al. (Chen et al., 1997) consider the problem of finding an optimal subset of features, that is, from a large set of candidate features, selecting a sub-

Terlecky P., Jiang Y., Xu X., Bar-Noy A. and Govindan R..
Maximizing the Relevant Diversity of Social Swarming Information.
DOI: 10.5220/0004717503650372
In *Proceedings of the 3rd International Conference on Sensor Networks* (SENSORNETS-2014), pages 365-372
ISBN: 978-989-758-001-7

set of features which are able to represent given examples (samples) consistently. They proves that the problem of finding an optimal subset of features is NP-hard, and presents a heuristic for solution.

Margules et al. (Margules et al., 1988) considers the problem of selecting a set of vendors in the manufacturing environment. This paper proposes a decision support approach to selecting vendors under the conflicting criteria of minimizing the annual material costs, reducing the number of suppliers and maximizing suppliers' delivery and quality performances.

The context of our work is social swarming. This context has also been the setting for the following works. Liu et al. (Liu et al., 2012a) considered maximizing the credibility of social swarming information. In (Liu et al., 2012b), Liu et al. were intersted in maximizing the number of timely reports sent to a commander. Jiang et al. (Jiang et al., 2013) developed a system called Mediascope for selective timely retrieval of media from mobile devices.

2 MODEL

A commander has m reporters in the field collecting data in various ways: photos, video, sound recordings, text. The commander would like to receive as much information as possible on a particular event or cirumstance. He would like the reporters to upload some subset of their reports so that the total "information" collected is maximized. Each report has a particular size and there is a limited amount of bandwidth assigned to each reporter. Given bandwidth constraints, he would like to determine which reports should be uploaded so as to maximize the total information received. Here information is modeled as tags which the reporters set to the reports. For example, if a reporter takes a photo, he tags the images in the photo. If a reporter reports via text, he includes keywords as his tags. The tags are the elements in the information universe I.

2.1 Single Format

First, let us assume all reports are of one format, and let us assume that this format is photos. The information universe is comprised of the elements in the set $I = \{i_1, \ldots, i_n\}$. A photo or information set J is a subset of I. Assume there are m users. User k's smartphone has u_k photos. Denote the photo set by P_k. Let photo p_{kj} denote photo j from user k. We wish to select $s_k \leq u_k$ photo's from user k, $k \in \{1, \ldots, m\}$ such that we maximize the information obtained, i.e. the cardinality of the union of the information sets selected. We may assume without loss of generality that $s_k = 1$ for each k for if $s_k > 1$: we may make s_k copies of user k's photo set and select one photo from each copy and if $s_k = 0$, we may ignore the photo set. We call this problem the Single Format Maximum Information Coverage Problem (SFINFOCOVER) and can represent this problem by the following program.

$$\max \ |\cup p_{kj} x_{kj}|$$

$$\text{s.t.} \sum_{j=1}^{u_k} x_{kj} \leq 1 \qquad k \in \{1, \ldots, m\} \tag{1}$$

$$x_{kj} \in \{0,1\} \qquad \forall k, j \tag{2}$$

It is represented as an integer program in the following way:

$$\max \sum_{i=1}^{n} y_i$$

$$\text{s.t.} \sum_{j=1}^{u_k} x_{kj} \leq 1 \qquad k \in \{1, \ldots, m\} \tag{3}$$

$$y_i \leq \sum_{i \in p_{kj}} x_{kj} \qquad i \in \{1, \ldots, n\} \tag{4}$$

$$y_i, x_{kj} \in \{0,1\} \qquad \forall i, j, k \tag{5}$$

The variables y_i, $i = 1, \ldots, n$ are indicator variables for selecting the i-th element. The variables x_{kj} are indicator variables for selecting the j-th set from photo set k. Inequality 3 assures that at most one photo is selected from each photo set. Inequality 4 guarantees that if no set is selected containing element i, then element i is not selected.

2.2 Multi-format

We now turn our attention to the multi-format scenario. In this scenario each reporter has a set of reports of different formats i.e. video, audio, text, etc. that can be uploaded to the commander. Let R_{ij} denote report j from reporter i and let $r_i = \{R_{i1}, \ldots, R_{iu_i}\}$ be the collection of u_i reports from reporter i. Let $I = \{i_1, \ldots, i_n\}$ denote the information universe. Each report R_{ij} is a subset of the information universe $R_{ij} \subseteq I$, for all i, j.

There is a size for each report which represents the file size of that report. Let $s(R_{ij})$ denote the size of report R_{ij}. Certainly reports of different formats have different sizes. Video files tend to be much larger than text files or photos, but they can also offer more information. Even within a format, files can be of different sizes. Consider video, audio, or text files.

They may be of different duration or length and as the duration or length of a file increase as does its size.

Reporter i has a fixed amount of bandwidth with which to upload reports and this is a constraint on the size of the reports that can be uploaded by a reporter. Let b_i denote the total size which can be uploaded by reporter i. Given the information universe $I = \{i_1, \ldots, i_n\}$, bandwidth constraints $b_1 \ldots, b_m$ for reporters $r_1, \ldots, r_m$ respectively, reports $\{R_{ij}\}$, and size of reports $\{s(R_{ij})\}$, which reports should be uploaded to maximize the number of obtained information elements subject to the bandwidth constraints $b_1 \ldots, b_m$?

This problem is called the Multi-Format Maximum Information Coverage Problem (MFINFOCOVER) and it can be represented by the following IP.

$$\begin{aligned} \max \sum_{i=1}^{n} y_i & \\ \text{s.t.} \sum_{t=1}^{u_i} s(R_{it}) x_{it} \leq b_i & \quad i \in \{1, \ldots, m\} \quad (6) \\ y_i \leq \sum_{i \in R_{kj}} x_{kj} & \quad i \in \{1, \ldots, n\} \\ y_i, x_{kj} \in \{0,1\} & \quad \forall i, j, k \end{aligned} \tag{7}$$

The significant difference of the IP for MFINFOCOVER compared to the IP for SFINFOCOVER is inequality 6 which bounds the sum of the sizes of reports uploaded by a reporter by the bandwidth available for that reporter.

2.3 Clique Model

In this subsection, we present the clique model. Assume the information universe is comprised of the elements in the set $I = \{i_1, \ldots, i_n\}$. Let R_{ij} denote report j from reporter i. In particular, $R_{ij} \subseteq I$. Represent each report R_{ij} $\forall i, j$ by a vertex. Each vertex R_{ij} has a size denoted by $s(R_{ij})$. The set of vertices is partitioned into m classes $r_1, \ldots, r_m$, where a class represents a reporter's set of reports. That is, $r_i = \{R_{i1}, \ldots, R_{iu_i}\}$ where u_i is the number of reports of reporter i. Each class r_i has a capacity b_i, corresponding to the bandwidth allocated to a reporter to upload his reports. This report graph is a complete graph with edge $e_{it,jk}$ having weight $w_{it,jk}$. The weight of an edge is the symmetric difference of the information sets of the vertices connected by the edge. That is,

$$w_{it,jk} = |R_{it} \cup R_{jk}| - |R_{it} \cap R_{jk}| \tag{8}$$

This weight measures how distinct the two reports are by counting the number of elements they differ in collectively. The optimization objective is to select a sub-clique of vertices in which the sum of the edge-weights is maximum over all feasible sub-cliques. A sub-clique is feasible if the sum of the sizes of all vertices selected from a class is at most the capacity of the class, for every class, i.e. $\sum_{t=1}^{u_i} s(R_{it}) x_{it} \leq b_i$, $\forall i = 1, \ldots, m$. We call this problem the Clique Information Coverage Problem (CLIQUEINFOCOVER).

We can formulate CLIQUEINFOCOVER as the following integer program.

$$\begin{aligned} \max & \sum_{it \neq jk} w_{it,jk} y_{it,jk} \\ \text{s.t. } & y_{it,jk} \leq x_{it} && \forall it \neq jk \quad (9) \\ & y_{it,jk} \leq x_{jk} && \forall it \neq jk \quad (10) \\ & x_{it} + x_{jk} - y_{it,jk} \leq 1 && \forall it \neq jk \quad (11) \\ & \sum_{t=1}^{u_i} s(R_{it}) x_{it} \leq b_i && \forall i = 1, \ldots, m \quad (12) \\ & x_{it} \in \{0,1\} && \forall i_t \\ & y_{it,jk} \in \{0,1\} && \forall it \neq jk \end{aligned}$$

In the integer program, variable x_{it} is used as the indicator variable for selecting vertex R_{it} for the clique. Similarly, variable $y_{it,jk}$ is used as the indicator variable for selecting edge $e_{it,jk}$ for the clique. Inequalities 9 and 10 ensure that edge $e_{it,jk}$ is not selected if either vertex it or jk is not selected. Inequality 11 guarantees that $y_{it,jk}$ is selected if both both vertices it and jk are selected. Inequality 12 ensures that the total size of vertices chosen from class r_i does not exceed b_i.

We analyze the integrality gap of the above integer program. Consider the following example with n an even integer. There are two classes, with $b_1 = b_2 = 1$ and $I = \{1, \ldots, n\}$. Class 1 consists of the reports $\{\{1\}, \{2\}, \ldots, \{n/2\}\}$ with each report being size 1. Class 2 consists of the reports $\{\{n/2+1\}, \ldots, \{n\}\}$ with each report also being of size 1. Note that any two reports are disjoint. The optimal integer programming solution chooses any set from class 1 and any set from class 2 and obtains an objective value of 2. A linear programming solution that selects $x_i = 1/(n/2)$ for all i obtains an objective of $2 \cdot \binom{n}{2} \cdot 1/(n/2)$ which is $2(n-1)$. Thus, the integrality gap of such an integer program is at least $n-1$. This result implies that rounding a linear programming relaxation is not very amenable for solution and that a linear programming upper bound on the optimal could be a very loose upper bound. With this in mind, we also consider a quadratic programming formulation of CLIQUEINFOCOVER.

The following is a quadratic programming formulation of CLIQUEINFOCOVER, it is similar to the quadratic program presented in (Alidaee et al., 2007)

$$\max \sum_{it \neq jk} w_{it,jk} x_{it} x_{jk}$$
$$\text{s.t.} \sum_{t=1}^{u_i} s(R_{it}) x_{it} \leq b_i \qquad \forall i = 1, \ldots, m$$
$$x_{it} \in \{0,1\} \qquad \forall it$$

The intuition behind the optimization goal of selecting a sub-clique with largest sum of edge-weights is to select a subset of reports which have high pairwise distinction with the belief that this will maximize the total number of elements covered. Consider the following instance: Let $I = \{1, \ldots, n\}$, reporter 1 has reports $\{1, \ldots, n\}$ and $\{\lfloor n/2 \rfloor - 1\}$ each of size 1 and reporter 2 has report $\{\lfloor n/2 \rfloor, \ldots, n\}$ of size 1. We have that $b_1 = b_2 = 1$ and thus each reporter can upload exactly one report. The reports which maximize the sum of dissimilarity are reports $\{\lfloor n/2 \rfloor - 1\}$ and $\{\lfloor n/2 \rfloor, \ldots, n\}$ giving a dissimilarity of $\lfloor n/2 \rfloor + 1$, while the reports which maximize the amount of information obtained are $\{1, \ldots, n\}$ and $\{\lfloor n/2 \rfloor, \ldots, n\}$ as they give the full information set I. The difference in obtained information is $\lfloor n/2 \rfloor - 2$ elements.

3 ALGORITHMS AND HEURISTICS

In this section we provide approximation algorithms and heuristics for the problems. It follows that defined problems are NP-Hard. The hardness of SFINFOCOVER and MFINFOCOVER follow from a trivial reduction from the Maximum Coverage problem. It also holds that CLIQUEINFOCOVER is NP-hard even for one class with 0/1 edge weights. The hardness of CLIQUEINFOCOVER follows from a reduction from the Maximum Clique problem. It also holds that there is no constant factor approximation for CLIQUEINFOCOVER, from this reduction.

The following greedy approximation algorithm called GREEDY for SFINFOCOVER gives a 2-approximation. The analysis is given in Chekuri and Kumar (Chekuri and Kumar, 2004). The idea behind GREEDY is as follows: for each photo set from which a photo has not already been selected, select the photo which covers the maximum number of uncovered elements, breaking ties arbitrarily. For a given round, select the photo out of these maximal photos which covers the maximum number of uncovered elements. Remove these elements from consideration in future rounds, and remove this photo set from consideration in future rounds.

Algorithm 1: GREEDY.

$H \leftarrow \emptyset$, $I' \leftarrow I$, $S \leftarrow \emptyset$
while $S \neq \{1, \ldots, m\}$ and $I' \neq \emptyset$ **do**
  for all Reporters $k \in \{1, \ldots, m\}$ **do**
    if a $k \notin S$ **then**
      $l \leftarrow \text{argmax}_j \{|p_{kj} \cap I'|\}$
      $G_k \leftarrow p_{kl}$
    else
      $G_k \leftarrow \emptyset$
    end if
  end for
  $c \leftarrow \text{argmax}_i |G_i|$
  $H \leftarrow H \cup \{G_c\}$, $S \leftarrow S \cup \{c\}$, $I' \leftarrow I' - G_c$
end while
OUTPUT: H, $I - I'$

Next, we present an approximation algorithm for MFINFOCOVER called MFGREEDY. It is a 12-approximate algorithm (Chekuri and Kumar, 2004). This algorithm adds reports greedily, in the following sense. In a given round, it computes the coverage per size of every remaining feasible report. It adds a feasible report with the largest coverage per size.

Algorithm 2: MFGREEDY.

$H \leftarrow \emptyset$, $I' \leftarrow I$, $b'_k \leftarrow b_k \; \forall k \in \{1, \ldots, m\}$, $G_c = R_{11}$
while $G_c \neq \emptyset$ **do**
  for all Reporters $k \in \{1, \ldots, m\}$ **do**
    for all reports R_{kt} **do**
      if $s(R_{kt}) > b'_k$ **then**
        $R_{kt} \leftarrow \emptyset$
      end if
      $l \leftarrow \text{argmax}_j \{|R_{kj} \cap I'| / s(R_{kj})\}$
      $G_k \leftarrow R_{kl}$
    end for
  end for
  $c \leftarrow \text{argmax}_i |G_i \cap I'| / s(G_i)$
  $H \leftarrow H \cup \{G_c\}$, $I' \leftarrow I' - G_c$, $b'_r \leftarrow b'_r - s(G_c)$
end while
OUTPUT: H, $I - I'$

We propose the following heuristic for CLIQUEINFOCOVER which we call CLIQUE-MAXSUM. For each reporter i, for each report R_{it}, $t \in \{1, \ldots, u_i\}$ belonging to reporter i, compute the ratio of sum of edge weights to size of the report. Until capacity b_i would be exceeded, add the feasible report with largest ratio from reporter i (and remove this report from r_i).

Algorithm 3: CLIQUE-MAXSUM.

$C \leftarrow \emptyset$
for all reporters $i \in \{1,\ldots,m\}$ **do**
  for all Reports R_{it}, $t \in \{1,\ldots,u_i\}$ **do**
    $S_{it} \leftarrow \sum_{jk} w_{it,jk}$
    Compute the ratio $k_{it} = S_{it}/s(R_{it})$
  end for
  while there is a report which can be added within capacity b_i **do**
    $C \leftarrow C \cup \{\text{feasible report } R^*_{it} \text{ with maximum } k_{it}\}$
    Remove R^*_{it} from r_i
  end while
end for

OUTPUT: C and sum of edge-weights of C

4 SIMULATIONS

In the simulation environment, we are interested in comparing the performance of the SFINFOCOVER model to the CLIQUEINFOCOVER model. We evaluate the CLIQUE-MAXSUM (abbreviated by CLIQUE) and GREEDY algorithms as well as an LP-relaxation (LP) for the SFINFOCOVER IP for randomly generated instances. LP acts as an upper bound on the optimal solution and is therefore a good scalable benchmark for comparison. The following parameters are varied in the simulations: the number of reporters, number of elements n in the information universe I, the number of photos per reporter, and the probability p that an element is in a photo.

In Simulation 1, we set the number of reporters to 3 with each reporter have 3 photos. The number of items is varied from 1000 to 1050 and for each n 10 runs are performed. In a run, for each photo, an item appears with probability 0.5. GREEDY outperforms CLIQUE on average by about 5 items.

In Simulation 2, the number of reporters is 4, and each reporter has 2 photos from which at most one can be selected. The number of items varies from 2000 to 2100 with 10 random runs being performed for each n. In a run, for each photo, an item appears with probability 0.5. GREEDY outperforms CLIQUE with GREEDY covering on average about 96% of items.

In Simulation 3 the number of reporters is varied from 1 to 5. There are 20 items and 2 photos per reporter. With both the GREEDY and CLIQUE algorithms, the number of items obtained increases as the number of reporters increases. GREEDY outperforms CLIQUE in the number of items selected with both algorithms converging to covering all 20 items when $n = 5$. For a run, an item of a photo has a probability of .5 of appearing.

In Simulation 4, the total number of items is 40, and there are two photos per reporter. The number of reporters varies from 1 to 6. GREEDY outperforms CLIQUE in the number of items selected, but the outperformance is minor. We see that when there is 1 reporter both algorithms cover on average 20 items. This is expected as an item for a photo has a probability of .5 of appearing.

In Simulation 5, the number of items is varied from 20 to 40. There are 3 reporters and 2 photos per reporter. The linear programming relaxation of the IP is also simulated along with GREEDY and CLIQUE. GREEDY and CLIQUE perform quite well to the LP which is an upper bound on the actual IP solution. GREEDY yet again outperforms CLIQUE, but the outperformance is quite minor. There were 15 runs for each n, and for a run, an item of a photo has a probability of .5 of appearing.

In Simulation 6, we set the number of reporters to 3 with 2 photos per reporter. The number of items is varied from 100 to 140. A photo matrix is randomly generate with the probability of an item appearing in a photo of .5. All three algorithms are run twenty times for a given n. GREEDY once again outperforms CLIQUE, with GREEDY being on average about 4 items from the LP upperbound on the optimal.

In Simulation 7, we vary the number of photos per reporter from 1 to 10. We hold constant the number of reporters at 3 and the number of items at 20. The number of runs for each p for each algorithm is 30. We see that the performance of CLIQUE falls off drastically from the performance of the LP and GREEDY. The performance of CLIQUE maintains a mean of about 15 over all p, where both LP and GREEDY increase as p increases with GREEDY covering on average about 18 items and LP covering about 20 with $p = 10$.

In Simulation 8 we vary n from 100 to 140 setting the probability of an item being in a photo to be .2. For each n, there is 100 runs for each of the 3 algorithms. The number of reporters is 3 and the number of photos per reporter is 2. GREEDY and CLIQUE perform comparably with CLIQUE for some n barely outperforming GREEDY on average. Both algorithms select on average about 2 items less than the LP.

In Simulation 9, we vary the probability of an item being included in a photo from 0 to 1 by 0.05. We hold fixed the number of items at 4, the number of reporters at 4, and the number of photos per reporter at 3. We see that both LP and GREEDY converge to covering all 4 items at a probability of .4. CLIQUE takes a longer time to converge to 4. It converges at a probability of .8.

In Simulation 10, the probability that an item is included in a photo is varied from 0 to 1 by 0.05. The number of items is 10, the number of reporters

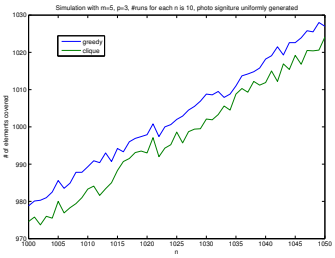

Figure 1: Simulation 1.

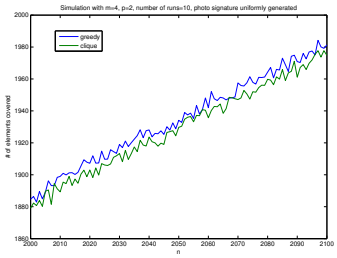

Figure 2: Simulation 2.

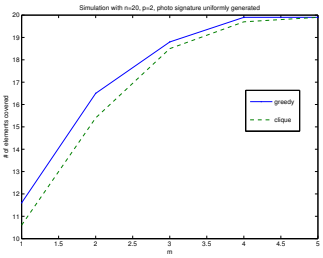

Figure 3: Simulation 3.

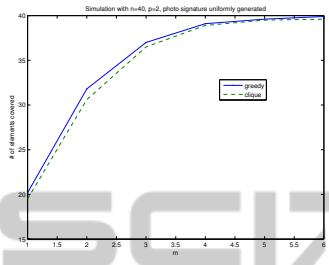

Figure 4: Simulation 4.

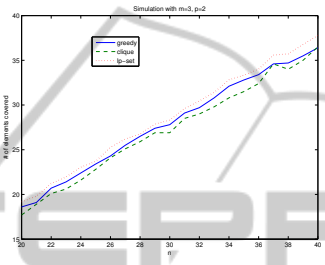

Figure 5: Simulation 5.

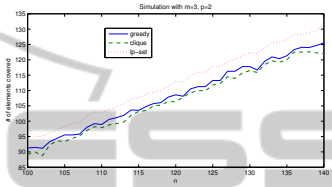

Figure 6: Simulation 6.

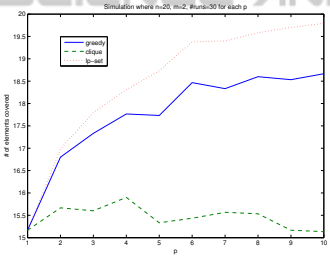

Figure 7: Simulation 7.

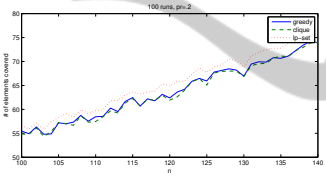

Figure 8: Simulation 8.

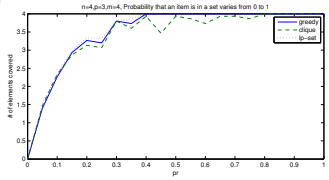

Figure 9: Simulation 9.

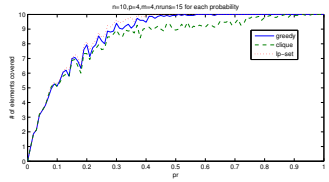

Figure 10: Simulation 10.

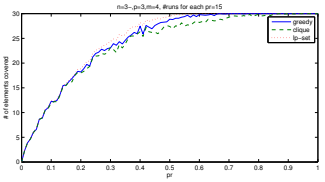

Figure 11: Simulation 11.

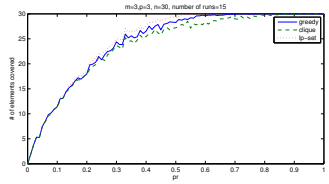

Figure 12: Simulation 12.

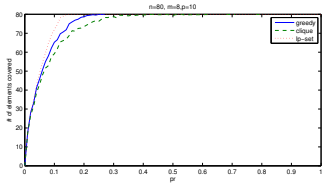

Figure 13: Simulation 13.

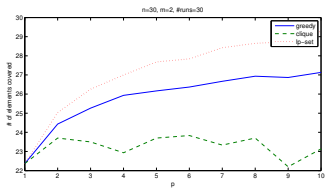

Figure 14: Simulation 14.

is 4, and there are 4 photos per reporter. The number of runs for each probability is 15. We see that the performance of GREEDY is quite close to the performance of LP. LP converges to covering all items at a probability of about 0.4. GREEDY converges to covering all items at a probability of about .5. CLIQUE converges to covering all items at a probability of .9. CLIQUES performance matches LP and GREEDY for probabilities of 0 to .3, but then its performance falls off slightly.

In Simulation 11, we set the number of items to be covered to 30. The number of reporters is 4 and each reporter has 3 photos from which he needs to choose one from. The probability of a item being included in a photo varies from 0 to 1 by .05. The number of runs for each probability is 15. LP converges to covering all items at a probabiliy of about .5. GREEDY converges to covering all items at a probability of .6. CLIQUE converges to covering all items at a probability of 0.85. While its performance is on par with LP and GREEDY when the probability is between 0 and .4, it underperforms both from .4 to .85.

In Simulation 12, we set the number of items to be covered to 30. The number of reporters is 3 and each reporter has 3 photos from which he needs to choose one from. The probability of a item being included in a photo varies from 0 to 1 by .05. The number of runs for each probability is 15. LP converges to covering all items at a probabiliy of about .5. GREEDY converges to covering all items at a probability of .6. CLIQUE converges to covering all items at a probability of 0.85. While its performance is on par with LP and GREEDY when the probability is between 0 and .4, it underperforms both from .4 to .85.

In Simulation 13, we once again vary probability that an item is included in a photo from 0 to 1 by .05. This time there are 80 total items, the number of reporters is 8 and the number of photos per reporter is 10. All three algorithms converge to covering all items rather quickly, with LP converging at a probability of about .15, GREEDY converging at a probability of about .25 and CLIQUE converging at a probability of about .4. A large gap of about 7 items on average can be seen in this simulation between GREEDY and CLIQUE at a probability of about 0.2.

In Simulation 14, we vary the number of photos per reporter. We fix the number of reporters to 2 and the number of items to be covered to 30. We set the probability of an item appearing in a photo to be .6. We run each of the three algorithms 30 times for a given number of photos per reporter. We see the performance of CLIQUE falls off significantly in comparison to the other two algorithms as p increases, selecting on average 23 items over all p, and remaining relatively flat on average.

From these simulations, we see that GREEDY outperforms CLIQUE with GREEDY performing very well compared to an LP-relaxation upper bound on the optimal solution.

5 CONCLUSIONS AND FUTURE WORK

We propose two novel models, a maximum coverage based model and a clique based model for modeling the problem of maximizing the amount of relevant diversity of social swarming data received by a commander. We provide well-performing algorithms/heuristics for both models with the maximum coverage algorithm outperforming the clique heuristic. In future work, we look to experimentally analyze the multiple-format models and develop a physical system based on our models.

ACKNOWLEDGEMENTS

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

Alidaee, B., Glover, F., Kochenberger, G., and Wang, H. (2007). Solving the maximum edge weight clique problem via unconstrained quadratic programming. *European journal of operational research*, 181(2):592–597.

Chekuri, C. and Kumar, A. (2004). Maximum coverage problem with group budget constraints and applications. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 72–83.

Chen, B., HONG, J., and WANG, Y. (1997). The problem of finding optimal subset of features. *CHINESE JOURNAL OF COMPUTERS-CHINESE EDITION-*, 20:133–138.

Dijkhuizen, G. and Faigle, U. (1993). A cutting-plane approach to the edge-weighted maximal clique problem. *European Journal of Operational Research*, 69(1):121–130.

Hunting, M., Faigle, U., and Kern, W. (2001). A lagrangian relaxation approach to the edge-weighted clique problem. *European Journal of Operational Research*, 131(1):119–131.

Jiang, Y., Xu, X., Terlecky, P., Abdelzaher, T. F., Bar-Noy, A., and Govindan, R. (2013). Mediascope: selective

on-demand media retrieval from mobile devices. In *IPSN*, pages 289–300.

Lappas, T., Crovella, M., and Terzi, E. (2012). Selecting a characteristic set of reviews. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 832–840. ACM.

Liu, B., Terlecky, P., Bar-Noy, A., Govindan, R., Neely, M. J., and Rawitz, D. (2012a). Optimizing information credibility in social swarming applications. *IEEE Transactions on Parallel and Distributed Systems*, 23(6):1147–1158.

Liu, B., Terlecky, P., Xu, X., Bar-Noy, A., Govindan, R., and Rawitz, D. (2012b). Timely report delivery in social swarming applications. In *DCOSS*, pages 75–82.

Macambira, E. and De Souza, C. (2000). The edge-weighted clique problem: valid inequalities, facets and polyhedral computations. *European Journal of Operational Research*, 123(2):346–371.

Margules, C., Nicholls, A., and Pressey, R. (1988). Selecting networks of reserves to maximise biological diversity. *Biological conservation*, 43(1):63–76.

Park, K., Lee, K., and Park, S. (1996). An extended formulation approach to the edge-weighted maximal clique problem. *European Journal of Operational Research*, 95(3):671–682.

Tsaparas, P., Ntoulas, A., and Terzi, E. (2011). Selecting a comprehensive set of reviews. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–176. ACM.

Yu, W., Zhang, R., He, X., and Sha, C. (2013). Selecting a diversified set of reviews. In *Web Technologies and Applications*, volume 7808 of *Lecture Notes in Computer Science*, pages 721–733.

Zhuang, L., Jing, F., and Zhu, X.-Y. (2006). Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 43–50.