DECISION SCIENCES
A JOURNAL OF THE DECISION SCIENCES INSTITUTE

Decision Sciences
Volume 48 Number 6
December 2017

Integrating Social Network Effects in the Share-Of-Choice Problem

Dilek Gunnec
Department of Industrial Engineering, Ozyegin University, Istanbul, Turkey, e-mail: dilek.gunnec@ozyegin.edu.tr

S. Raghavan†
Smith School of Business and Institute for Systems Research, University of Maryland, College Park, MD, e-mail: raghavan@umd.edu

ABSTRACT

Accounting for social network effects in marketing strategies has become an important issue. Taking a step back, we seek to incorporate and analyze social network effects on new product development and then propose a model to engineer product diffusion over a social network. We build upon the share-of-choice (SOC) problem, which is a strategic combinatorial optimization problem used commonly as one of the methods to analyze conjoint analysis data by marketers in order to identify a product with largest market share, and show how to incorporate social network effects in the SOC problem. We construct a genetic algorithm to solve this computationally challenging (NP-Hard) problem and show that ignoring social network effects in the design phase results in a significantly lower market share for a product. In this setting, we introduce the secondary operational problem of determining the least expensive way of influencing individuals and strengthening product diffusion over a social network. This secondary problem is of independent interest, as it addresses contagion models and the issue of intervening in diffusion over a social network, which are of significant interest in marketing and epidemiological settings.[Submitted: September 29, 2015. Revised: June 25, 2016. Accepted: July 6, 2016.]

Subject Areas: Genetic Algorithm, Marketing, Optimization, Peer Influence, Product Design, and Social Networks.

INTRODUCTION

This article is motivated by the desire to account for social interactions among consumers—who influence each other while making purchasing decisions—in the context of product design. Such interactions have become easier to observe with the prevalence of online social networks and online tracking technologies. We focus on the effect of these interactions in the context of the introduction of a new product. In this setting, we consider both the strategic (through product design) and

†Corresponding author.

operational (through product diffusion) aspects by taking into account the spread of the adoption behavior over a social network with on-site and well-advised marketing interventions.

We consider the setting where peer influence plays a significant role in a consumer's product choice or there is a tangible benefit from using the same product as the rest of one's social network, such as with communication products. Family plans encourage using the same phone network by offering large discounts to customers for in-network calls. When a member of a social group has to choose between products, it is natural to take into account the number of people who already own the product in addition to the product's attributes. In the extreme case, if the communication is carried over the Internet, such as voice or video chat, it is compulsory for users to install the same application (e.g., Skype). Modern business models have started harvesting social network information among their consumers into their business. American Express has created an application that links their customers' Twitter accounts with their American Express accounts to provide them with perks while implicitly letting them share their purchases with their friends (https://sync.americanexpress.com/twitter/Index). Mobile applications gain access to a user's social network by asking users to log in using their existing social media accounts. Zynga offers a large variety of games, which can be played over different online social media. It is a platform where the users of a product (a game in this case) are explicitly connected with each other over a social network, which makes it easier to collect information on the relationship network among the customers. In this environment, the product design that best fits the customers can increase the market share significantly. The type of environment we are considering could also be imagined as a potential customer visiting an online shopping Web site (e.g., Amazon), where on one side of the page the customer can observe if friends from a social network (e.g., Facebook) have purchased the same product. (Although Amazon does not have this exact feature, it allows users to connect Amazon and Facebook accounts and makes gift recommendations using content from their friends' Facebook accounts.)

Our approach is focused on influence over neighborhood relationships (social ties) among the users of the product separating it from the traditional network externalities models (Katz & Shapiro, 1985). Being able to analyze a social network provides marketers a competitive advantage in terms of forecasting the spread of product influence and intervening at times with promotions or incentives to strengthen this process. Predictions on market share (the number of people who will purchase the product) are critical when a new product enters the market and can create a difference for businesses.

Product design has been of significant interest both in academic studies and industry applications. Market share forecasts and estimated customer utilities for a product profile allow for a better understanding of the market needs and can lead to better product designs for the companies. Calculation of perceived values for product features has drawn attention from market researchers for many years. Conjoint analysis is one of the most popular tools in new product design for identifying customer preferences and utilities for attribute levels of a product called *part-worth utilities* (Green & Rao, 1971). It has been studied widely in the marketing literature and has been used to design many different products in practice. Broadly, there are two main steps in a conjoint analysis. The first is the data collection from consumers and the second is the analysis of these data to obtain part-worth utilities for each customer on each attribute level. After this, part-worth utilities are analyzed to design the product (or set of products) that maximizes market share.[i] Earlier literature on marketing has focused on these two steps, especially in valid data collection and improved statistical estimation. However, the natural next step of optimally using such data via conjoint *optimization* to design a product with maximum market share has been given relatively less attention (Camm, Cochran, Curry, & Kannan, 2006). The first optimization approach using conjoint data was proposed by Zufryden (1977). In this regard, the share-of-choice (SOC) problem (Kohli & Krishnamurti, 1987) is a product design problem to select levels for each of the attributes of a product. The objective is to create the product profile that will return the largest market share. Finding this product profile requires the use of an optimization approach when the number of attributes and levels of the attributes are large (so explicit enumeration is not viable) and most product designs arising from different attribute level combinations are technologically and economically feasible (Nair, Thakur, & Wen, 1995).

In this article, we also look at the combinatorial optimization problem at the design (strategic) phase of a product with the same objective, maximizing market share. However, our main contribution is that while designing the product we also explicitly include social network effects that take place at the product adoption (operational) stage after the product is launched. In this way, the strategic decision of finding a good product design involves future operational moves, creating a non-myopic decision-making process. We also show that incorporating such network effects in the design phase finds a product profile that has a significantly larger market share than if a profile is constructed without taking into account the network effects. This novel far-sighted approach creates a complex optimization problem that is hard to solve by complete enumeration or by an integer programming (IP) model. Our second major contribution in this article is to propose a hybrid solution approach (a matheuristic that combines mathematical programming with metaheuristics) that provides near-optimal solutions to the problem in an efficient manner.

In the next section, we provide a brief literature review on product design and product influence and define the SOC problem in greater detail. After that, we present models that incorporate social network effects into the SOC problem in the strategic product design phase and introduce a consequent model for the operational phase to manage diffusion of the product over the network. Next, we present a genetic algorithm (GA) for the SOC problem with social network effects. Finally, we present our computational studies that demonstrate both the benefit of explicitly incorporating social network effects in the product design phase and validate the use of the GA as a high-quality solution approach (both in terms of computational speed and near-optimality of the solutions).

[i] Although one can consider profits instead of market share, it is more common to look at market share in the product design phase as pricing decisions are typically taken in the operational phase after the product is designed. Further, it is well known in the marketing literature that market share and profitability are strongly correlated (Szymanski, Bharadwaj, & Varadarajan, 1993); thus, it is reasonable to use market share as the objective in the product design phase. Note that product price may be considered as an attribute within the conjoint analysis.

LITERATURE REVIEW

Research on consumer behavior shows that consumers' purchase decisions and product evaluations are influenced by their reference groups (Bourne, 1957; Burnkrant & Cousineau, 1975; Bearden & Etzel, 1982; Childers & Rao, 1992; Iyengar, Van den Bulte, & Valente, 2011). Such influence has been given various terms such as bandwagon effect, peer influence, neighborhood effect, conformity, and contagion (Iyengar, Han, & Gupta, 2009). In a recent study, Narayan, Rao, and Saunders (2011) introduced conjoint estimation models using three behavioral mechanisms of how consumers' product choice decisions may be affected from influence of their neighbors. We explain these mechanisms in the next section when we describe our influence model.

Product design is a well-studied subject in marketing and the implications of network externalities on several aspects of marketing (including customer behavior and market structure, product-related decisions such as preannouncements, timing of product introductions, and product differentiation and market entry, Srinivasan, Lilien, & Rangaswamy, 2004) have been explored. In terms of social network effects, the organizational structure and the management of new product development teams have been studied (Leenders, van Engelen, & Kratzer, 2003; Sosa, Eppinger, & Rowles, 2004) with respect to their relation with the design of a product and the creativity involved in the process. Social network effects are also being considered for better managing the distribution of the product (Godes et al., 2005). From a product design perspective, Aral and Walker (2011) acknowledged the effects of viral marketing and argued that such viral features can be engineered during the launch of the product. The authors differentiated the "viral characteristics" and "viral features" of a product. The first relates to the content of the product whereas the second corresponds to how the product is shared and how the features allow relationships with the other consumers. Concentrating on the viral features, they showed that whereas the personalized referrals are more effective in encouraging adoption, passive-broadcast viral messaging is used more often and therefore causes a larger overall adoption. Dou, Niculescu, and Wu (2013) considered how a monopolistic firm can strategically engineer the strength of network effects at utility level via social media and also focused on viral features to derive the optimal level of social media functionality that increases the value of social interactions to each user. In this article, we focus more on the viral characteristics of a product implicitly by looking at the changes in the utility consumers get from using the same product with their neighbors on the network. We specifically consider the adoption as a passive-broadcast viral message, which increases utilities for the other consumers that are directly connected to the owner.

Part-worth utilities are used as inputs for the SOC problem to optimize the selection of levels for each attribute. In this problem, one buys (or adopts[ii]) a product only when the utility the person gets by using the product is greater than or equal to their "hurdle." The "hurdle" is the utility value at which one would be indifferent between making a purchase or not making a purchase. We model the

[ii] We use the terms adopt and buy interchangeably throughout the article.

social network influence effects on product design by making adjustments to the well-studied SOC problem. The SOC problem can be mathematically modeled as follows. Let $V = \{1, 2, \dots, n\}$ denote the set of people in the market, h_s denote the hurdle utility of person $s \in V$, K be the number of attributes, L_k be the number of levels for attribute $k = 1, 2, \dots, K$, and u^s_{kl} denote the part-worth utility for person s if level l is chosen for attribute k. Data on part-worth utilities are obtained via conjoint analysis studies (Green & Rao, 1971). There are two types of binary decision variables; x_{kl} equals 1, if level l has been chosen for attribute k and is 0, otherwise; and y_s equals 1, if person s decides to buy the product and is 0, otherwise. The SOC problem is then formulated as an integer program identically to Camm et al. (2006), and as shown below.

$$\textbf{SOC}: \qquad \text{Maximize} \sum_{s=1}^{n} y_s, \tag{1}$$

$$\text{subject to } \sum_{k=1}^{K} \sum_{l=1}^{L_k} u^s_{kl} x_{kl} \geq h_s y_s \quad s = 1, 2, \dots, n, \tag{2}$$

$$\sum_{l=1}^{L_k} x_{kl} = 1 \quad k = 1, 2, \dots, K, \tag{3}$$

$$y_s \in \{0, 1\} \quad s = 1, 2, \dots, n, \tag{4}$$

$$x_{kl} \in \{0, 1\} \quad k = 1, 2, \dots, K, \quad l = 1, 2, \dots, L_k. \tag{5}$$

In this model, the objective is to maximize the market share. (If the objective is to maximize the total utility of the customers or the firm's marginal return, the problem is called the "buyer's welfare" or "seller's welfare" problem.) Constraint (2) guarantees that people buy the product only if their utilities from the product exceed their hurdles. Constraint (3) ensures that each attribute is assigned only to a single level. Note that no network effects are taken into account in this model.

The SOC problem has been studied in the marketing literature and shown to be an NP-hard problem (Kohli & Krishnamurti, 1989). Several heuristics have been used as solution approaches including a divide-and-conquer heuristic (Green & Krieger, 1989), greedy search and dynamic programming–based heuristics (Kohli & Krishnamurti, 1987, 1989), a GA (Balakrishnan & Jacob, 1996), and a nested partitioning algorithm (Shi, Ólafsson, & Chen, 2001). An exact branch-and-bound algorithm (Camm et al., 2006) has been applied more recently for the SOC problem.[iii] For the product line design problem (designing multiple products of the same product line), Belloni, Freund, Selove, and Simester (2008) provided a Lagrangian relaxation method and Wang, Camm, and Curry (2009) developed a branch-and-price approach. Belloni et al. (2008) also provided a review of methods for product

[iii] We should note that some commercial packages for conjoint analysis provide optimization capabilities using some of the heuristics mentioned (Sawtooth Software, 2003).

line optimization in general. None of the previous studies consider integrating the network effects, and to our knowledge, this is the *first study* to explicitly incorporate social network (or peer influence) effects among potential customers in the SOC problem and product design. While our article covers a single product, our model and solution methods can be extended to the product line design problem. However, because the influence structure can be augmented by the additional influence among users of different products in the same product line, the extended model is somewhat more complicated than that of a single product design and is described in Appendix C.

Although our focus is on product design, the incorporation of social network effects in different operational or strategic settings is an area of significant interest. For example, incorporation of social network effects in pricing is an area of recent research. In this regard, Candogan, Bimpikis, and Ozdaglar (2012) and Cohen and Harsha (2016) study optimal pricing problems when there are interaction effects between users on a network. While the high level motivations are similar (account for social network effects), because the problem domains are quite different, the models and techniques in these papers are unrelated to the work within this article.

INCORPORATING SOCIAL NETWORK INFORMATION

In this section, we discuss our model to incorporate social network effects in the SOC problem for the strategic product design problem. A secondary problem, the least cost influence problem (LCIP), is introduced for making the operational decision related to product diffusion, as a complementary model using a subset of the network representation used for the SOC problem with network effects.

Share-of-Choice Model Incorporating Social Network Effects (SOCSNE)

Consistent with social contagion research (Bell & Song, 2007; Manchanda, Xie, & Youn, 2008), we follow a multiattribute linear utility-maximization approach. We focus on cases where an individual can only observe the outcome of a consumer's purchase decision, but not the relative preferences among each attribute level for three reasons. First, it is less complicated to collect data on purchase decisions among large social network groups using sales data than information about consumers' relative preferences among attributes (in fact due to privacy concerns it may not be feasible to do so). Second, potential consumers are exposed to peers' decisions about a product profile over online and offline social networks more frequently because such information sharing requires less proximity or intimacy among consumers. Third, this approach allows for a tractable (and solvable) model for the product design problem while still providing a well-established model to include social network effects. Our mechanism to model social network effects processes the influence from neighbors as an additional attribute of the product. Under this mechanism, the influence from other consumers adds to (or detracts from) product utility of a consumer in a linear additive manner.

In their recent study, Narayan et al. (2011) considered three behavioral mechanisms of how consumers' product choice decisions may be affected from influence of their neighbors. The first is a Bayesian mechanism where a consumer's updated preference for an attribute of the product is a weighted average of their initial (prior) preference and the preferences of their neighbors for the same attribute. The second mechanism is a more generalized Bayesian mechanism and allows for a more flexible process of preference revision. Finally, the third mechanism, which is based on the literature on social contagion and identical to our approach, abstains from updating the relative attribute preferences. Although they suggested that their first model fits their particular data set (one study of MBA students in the same class) best, Narayan et al. (2011) also found that the mean extent of influence of neighbors' choices on consumer utility is positive and significant. Thus, the choice of a product profile by an influencer leads to an increase in the utility of that profile for the influenced consumer. Narayan et al. (2011) agreed that their study might be less representative of the cases where influence among individuals exclude choice-related information sharing and when the number of peers is large. Most importantly, data burdens in the first and the second methods are significant and it is not clear if the required data would even be available in a social network setting. Consequently, we propose a linear influence model where the influence between people is dependent only on the person being influenced and the number of neighbors of this person (which is similar to the third model of Narayan et al., 2011). We carry out the computational studies in this article using this linear model. In addition, we also propose two different models where the influence is dependent on the influencer as well as the influenced, and the structure of the influence is nonlinear in the number of influencers.

Communication among friends strengthens the inclination toward buying the same product, which others have also purchased. The counterpart of this behavior in our model is the decrease in one's hurdle (this could alternatively be viewed as an increase in utility). The amount of decrease is limited within a hurdle span, which we define as the interval between a high and a low hurdle. Hurdle in the traditional product design problem (i.e., where no social network effects are considered) corresponds to a high hurdle. A low hurdle is the smallest value of utility a hurdle can have (i.e., when all friends buy the product). For this article, motivated by privacy concerns[iv] prevalent on social networks, we assume a decrease structure that does not depend on the identity of the neighbor for the influence effect in our model. In particular, we use a linearly decreasing influence effect.

Let h_s^H denote the "high" (H) and h_s^L denote the "low" (L) hurdle for person s. The amount of decrease in high hurdle depends on the number of neighbors who purchase the product. (In the graph representation of the social network, each person is represented as a node and the connection to each neighbor on the social network by an edge. The number of neighbors of a node is referred to as the degree of a node.) We calculate the unit decrease in hurdle for person s, Δ_s (≥ 0), as the ratio of the hurdle span and the degree of each node ($deg(s)$): $\Delta_s = \frac{h_s^H - h_s^L}{deg(s)}, \quad \forall s \in V$. Using this definition for Δ_s, the SOC model incorporating

[iv] In 2007, due to protests from users about privacy concerns, Facebook retreated on a tracking program called Beacon, which sent messages to users' friends about what they are buying on Web sites like Travelocity.com (Story & Stone, 2007). To address this, our model assumes that the anonymity of the users' friends will be preserved and only information on the number of friends purchasing the product is provided.

social network effects can be formulated identically to the previous model, SOC, except that constraint (2) is replaced with the following new constraint:

$$\sum_{k=1}^{K}\sum_{l=1}^{L_k} u_{kl}^{s} x_{kl} \geq h_s^H y_s - \Delta_s \sum_{j \in V} a_{js} y_j \quad s = 1, 2, \ldots, n. \tag{6}$$

In constraint (6), the right-hand side represents the *current* hurdle for person $s \in V$. Each entry of the social network adjacency matrix is shown by a_{js}, and it is 1 if there is an edge (j, s) between nodes j and s. We refer to the formulation with objective (1), constraints (3), (4), (5), and (6) as the SOC model incorporating social network effects (SOCSNE).[v] In an environment where the influence effects are negative and have a linear increase structure on the hurdles, Δ_s would simply be negative in constraint (6).

Our model (SOCSNE) can easily be adapted for *any other influence structures* (nonlinear, increasing, etc.). We now describe two additional influence models (that we do not computationally study in this article) to show how this can be achieved.

(i) Let f_{si}, $i, s \in V$, be the amount of decrease in hurdle of person s for the ith additional neighbor who has purchased or adopted the product, and let g_{si} be a binary decision variable which equals 1 if there are at least i neighbors of node s who buy the product, and 0 otherwise. Then, constraint (6) would be replaced by constraints (7), (8), and (9). Constraint (7) is similar to constraint (6) in terms of calculating the current hurdle for node s. Constraint (8) sets the number of g_{si} variables that are 1 equal to the number of neighbors who adopt the product. Constraint (9) establishes an ordering on the g_{si} variables to correctly compute the change in hurdle. When compared to the previous model, Δ_s and f_{si} represent the decrease in the hurdle when a neighbor buys the product. However, f_{si} is used when the decrease is represented with a nonlinear function; therefore, the network effect is dependent on the number of previous buyers, i. The adjacency matrix, a_{js}, is the same for both models.

$$\sum_{k=1}^{K}\sum_{l=1}^{L_k} u_{kl}^{s} x_{kl} \geq h_s^H y_s - \sum_{i=1}^{deg(s)} f_{si} g_{si} \quad s = 1, 2, \ldots, n. \tag{7}$$

$$\sum_{j \in V} a_{js} y_j = \sum_{i=1}^{deg(s)} g_{si} \qquad s = 1, 2, \ldots, n. \tag{8}$$

$$g_{s1} \geq g_{s2} \geq \ldots \geq g_{s(deg(s))} \qquad s = 1, 2, \ldots, n. \tag{9}$$

(ii) In a setting where neighbors of a node have different influences (this may be the case when the privacy concerns discussed earlier are moot),

[v] In Appendix A we provide a worst case bound on the lost market share when network effects are *not* incorporated in the design process.

we suggest modifying the previous model as follows. Let Δ_{js} represent the influence of neighbor node j on node s (i.e., the amount by which node j's product adoption reduces node s's hurdle). To calculate Δ_{js}, we propose a weighted directed graph model, where (j, s) represents the directed link from node j to node s and $a_{js} = 1$ if there is a directed link from node j to node s with $A(s) = \{j|a_{js} = 1\}$ being the set of neighbors of node s. Further, the relative influence of node j on node s is denoted by the weight w_{js} (analogous to Narayan et al., 2011). For each link (j, s), Δ_{js} can now be calculated using the formula

$$\Delta_{js} = \left(h_s^H - h_s^L\right) \frac{w_{js}}{\sum\limits_{j \in A(s)} w_{js}}. \tag{10}$$

After Δ_{js} is calculated (by Equation (10)), constraint (6) in the SOCSNE model is replaced with the inequality (11) so that the influence depends on the neighbor as well as the node itself. (If there is no directed link from node j to node s, then Δ_{js} is set to 0.)

$$\sum_{k=1}^{K} \sum_{l=1}^{L_k} u_{kl}^s x_{kl} \geq h_s^H y_s - \sum_{j \in V} \Delta_{js} a_{js} y_j \quad s = 1, 2, \ldots, n. \tag{11}$$

LCIP over the Buyer Network

The SOCSNE model implicitly allows for simultaneous buying; if one knew that the others are buying, all of their hurdles would decrease at the same time and they would all buy (which is similar to group buying settings such as Groupon). In an operational setting, we would need at least one person to purchase the product first and lead neighbors to adopt (as their utilities will be increased through their neighborhood relationship). Therefore, to obtain the maximum market share solution from the product design of the SOCSNE model, some people may need to be provided with incentives to make a purchase. In practice, such intervention is not costless, and for a business they can be contemplated as advertising or marketing costs of distributing free samples or discount coupons to a select group of individuals. We consider an incentive as an additional utility provided from the company selling the product to the potential customer. Once a person makes a purchase, their neighbors' hurdles are updated and checked to see if they also adopt the product. Every time a new person adopts the product, this update and comparison is repeated. If at some point there is no new buyer and the market share has not yet reached the amount dictated by the SOCSNE model, a new person needs to be given incentives to purchase the product and to restart the diffusion. Although providing incentives results in a larger market share (equal to the amount dictated by the SOCSNE model), it could be expensive, therefore the set of people to give incentives to needs to be selected with care. This trade-off in the operational setting is the subject matter of the second problem in this article.

The objective of the LCIP is to accomplish the market share of the SOCSNE model while *minimizing the total amount of incentives given*. To choose the set of critical people to give incentives to, we analyze the ordering of buyers by introducing time dimension, $t = 0, 1, \ldots, T$ (where T is the number of time periods). The product profile, the market share, and the individuals who will buy the product are inputs for the LCIP model because this problem is solved *after* the solution to the SOCSNE model is obtained. The social network in this problem, G', is a subset of the network in the SOCSNE model and includes only the nodes, V', that adopt the product as a result of the product profile chosen after solving the SOCSNE model (i.e., it includes only those nodes for which $y_s = 1$ in the SOCSNE model solution) and the edges connecting them, E'. There are two types of decision variables; $z_s, s \in V'$, represents the amount of incentive given to person s and $y_{st}, s \in V', t = 0, 1, \ldots, T$, is a binary variable, which is 1, if person s buys in period t and is 0, otherwise. Because the attribute levels have been chosen by the SOCSNE model, the utility one gets by using the product can simply be represented as one parameter, U_s, for each person $s \in V'$ (it is the summation of utilities from each level selected in each attribute). The mathematical formulation is as follows:

$$\textbf{LCIP}: \quad \text{Minimize} \quad \sum_{s \in V'} z_s, \tag{12}$$

$$\text{subject to } U_s \geq h_s^H y_{s0} \qquad \forall s \in V', \tag{13}$$

$$U_s \geq h_s^H y_{st} - z_s - \Delta_s \sum_{j \in V'} a'_{js} y_{j,t-1} \quad \forall s \in V', \forall t \geq 1, \tag{14}$$

$$y_{st} \geq y_{s,t-1} \qquad \forall s \in V', \forall t \geq 1, \tag{15}$$

$$y_{sT} = 1 \qquad \forall s \in V', \tag{16}$$

$$y_{st} \in \{0, 1\} \qquad \forall s \in V', \forall t \geq 0, \tag{17}$$

$$z_s \geq 0 \qquad \forall s \in V'. \tag{18}$$

The objective in this model is to minimize the total amount of incentives given to all the people in the buyer network. Constraint (13) identifies the people who buy the product in period 0 without requiring any incentives or influence from neighbors. Constraint (14) is similar to constraint (6) in the SOCSNE model; the right-hand side captures the current hurdle. The current hurdle of a person in any time period is calculated as the remainder after the amount of incentives *and* the social network effects are subtracted from the high hurdle. Note that in the SOCSNE model, the hurdle of a person decreases only with social network effects. In the LCIP model, in addition to the decrease from social network effects, hurdles are also decreased by the incentives people receive. The second term in the right-hand side of the constraint (14), z_s, represents this incentive amount for person s.

Observation 1: The amount of incentive one receives equals the difference between one's current hurdle and the utility received from the product. If the incentive is less than this difference, the person will not buy the product. If it is greater than this difference, it will hurt our objective function of minimizing the total amount of incentives. This consequently suggests a weak upper bound for the objective function as $\sum_{s\in V'}(h_s^H - U_s)$, which is the summation of all the differences between hurdles and utilities for each node.

Observation 2: The number of periods is less than or equal to the number of people: $T \leq |V'|$. Note that once $y_{st} = 1$ for $t = k$, $y_{st} = 1$ for $t \geq k$ by constraint (15) and constraint (16) guarantees that every person on the network (i.e., the buyer network) buys the product by forcing the last period decision to be 1. By this definition of a time period, there may be more than one buyer in a period.

Our model for the LCIP effectively follows the well-known linear threshold model (Granovetter, 1978) that is extensively used to model contagion or diffusion over a network. The linear threshold model starts with an initial set of nodes that have already adopted a product. A node is "influenced" by its neighbors who have adopted the product. When the sum total of influences coming to a node from its neighbors that have adopted the product exceeds a given threshold (which is unique to each node), the node adopts the product. Our LCIP model follows the linear threshold model because a node adopts the product if and only if the sum of the influences from its neighbors and any incentives from outside the social network is greater than the threshold, $h_s^H - U_s$.

The problem of finding nodes to maximize the spread or influence over a social network has been studied previously, and is of increasing interest in a variety of disciplines including marketing, computer science, and information systems. The most prominent research in this area is the "influence maximization problem" (Kempe, Kleinberg, & Tardos, 2003). Here, the objective is, for a given parameter k, to find a set of k nodes to maximize influence over the network. The k individuals are seeded (i.e., given the product for free) at the beginning of the diffusion process and the diffusion process takes place (with these k nodes exerting an influence on their neighbors). Our LCIP deviates from this approach; the incentives in our model are not seed products and involve partial inducements tailored for a set of individuals.[vi] We introduce incentive-discrimination, which in a marketing setting could correspond to 10%, 50%, etc. discount coupons rather than free samples of a product. In this approach, an incentive is used to resolve a knot in the diffusion process that will lead to a larger spread with more influenced nodes at the end. Such catalysation addresses the trade-off of minimizing the amount of incentives given and reaching each individual in the network.

The LCIP is a computationally challenging problem, which can take a significant amount of time to solve the IP model even for a relatively small problem. This is because the model has a time index, and thus for a model where $T = |V'|$ has $O(|V'|^2)$ variables and $O(|V'|^2)$ constraints. Consequently, it is not practical to solve this LCIP formulation. Later in this article, we provide an iterative approach to solve the LCIP optimally and in much shorter time. The LCIP is a problem of

[vi] Note that opinion leaders in this model are not determined by network measures such as centrality or highest connectivity. Rather the model explicitly takes into account the amount of incentives required for an individual to adopt a product as well as the network structure and how the product adoption would cascade in determining the opinion leaders or individuals to target.

independent interest with applications in epidemiology. This connection is discussed in Appendix B.

A GA FOR THE SOCSNE MODEL

The SOCSNE problem is a computationally challenging problem, hence we use a GA to solve this problem. A GA (Holland, 1975) is an evolutionary search algorithm that imitates natural selection of species in order to reach near-optimal solutions. We use a GA approach for the SOCSNE problem for three reasons. First, the time required to solve the presented IP model exactly increases rapidly with the size of the problem (making it a nonviable computational option). Second, due to influence effects, even for the small integer programs state-of-the-art solvers like CPLEX have numerical instability and one is unable to find optimal solutions (specifically CPLEX finds incorrect optimal solutions!) to the problem. Third, specialized approaches to solve the original SOC problem cannot be extended easily to the SOCSNE problem (as we explain further below). Our GA generates high-quality solutions and is robust in terms of computational time.

For the original SOC problem, Camm et al. (2006) use Lagrangian relaxation with the branch-and-bound method to solve the problem exactly. In this method, a search tree is developed, where each level of the search tree corresponds to an attribute of the product. So a node down a path in the tree would have some levels of attributes fixed already. The search tree is pruned using logic-based rules. With these rules, a node is fathomed if the path starting at this node cannot produce a feasible solution superior to one that is already known. This is evaluated by checking whether people's hurdles fall in the range of the minimum and maximum utilities a person may have if that path is followed for the product design. The network relationship among prospective customers in this article prevents such logical inferences because comparison of utilities from the product and the hurdles include social network effects, which depend on the buying status of one's neighbors. The calculation of the objective value at each node would still require the solution of the IP model proposed in this article, which would significantly slow down the methodology and actually make it computationally intractable. For the product line design case, efficient methods are presented and compared by Belloni et al. (2008). Their findings are consistent with the literature in terms of supporting the superiority of the GA approach over other methods such as dynamic programming, beam search, or a greedy heuristic. While simulated annealing is as successful, they report that it has a running time that is one or two orders of magnitude larger than other methods. Balakrishnan and Jacob (1996) demonstrated the use and advantages of GAs for solving product design problems. Their study provides a starting point for the GA in this article. We use a similar approach to obtain the product profile with the highest market share. However, our GA varies in several regards from Balakrishnan and Jacob (1996) including the fitness evaluation. The outline and the details of the GA are given below. In our description, we assume the reader has some familiarity with GAs. A good introduction to GAs is the text by Goldberg (1989).

Outline of the GA for the SOCSNE Problem

Input: *Parameters*: population size, mutation rate, number of generations.

Data: number of attributes, number of levels for each attribute, social network of people, high and low hurdles for each person, utilities for each person.
Output: Recommended product design, market share of the chosen product design.

Step 1 [GENERATE] Generate an initial population of q product profiles. Set $t = 0$.
Step 2 [EVALUATION] Calculate fitness of each product profile and let BEST = the profile with the largest fitness.
Step 3 [CROSSOVER] Perform single-point crossover operation to generate q offsprings.
Step 4 [MUTATION] Perform mutation.
Step 5 Calculate fitness of each product profile. If the largest fitness > BEST, update BEST.
Step 6 [REDUCTION] Reduce the population to half by choosing the ones with greatest fitness. $t = t + 1$.
Step 7 If $t <$ number of generations, then go to Step 3.
Else, STOP.

GENERATE: Each individual in the population is a product profile and is represented by a binary string, size $\sum_{k=1}^{K} L_k$, where L_k is the number of levels for attribute k. An *initial population* is generated randomly by assigning one level for each attribute. For example, if the product has 2 attributes, color and size with 2 and 3 levels as (black, white) and (small, medium, large), respectively, then the product with color white and size small would be represented as (01 100).

EVALUATION: After the population is generated, each product profile in the population is evaluated for fitness. *Evaluation* of a product profile corresponds to calculating the market share if that product profile is launched in the market. The exact value of market share is easily determined with the given x_{kl} values for that profile (e.g., by solving the integer program SOCSNE with the x_{kl} values fixed). Methods used to calculate the fitness should be chosen carefully. In an earlier approach, we used an approximate fitness function (where the number of people adopting the product was calculated simply by comparing hurdles with the utilities while hurdles are updated after each purchase) for which the results of the GA were significantly worse. Using the exact evaluation via an integer program (e.g., with the solver CPLEX) corresponds to a hybrid approach called MATHEURISTICS (Maniezzo, Stützle, & Voß, 2009) marrying mathematical programming with metaheuristic approaches.

CROSSOVER: In this step, two offspring product profiles are produced by two parent product profiles. Parents are chosen from a given population with respect to their fitnesses using the roulette-wheel mechanism (Michalewicz, 1996). This allows individuals with higher fitnesses to be more likely to get selected as parents. The offspring carry properties of both parents. We use a *single-point crossover* to determine how the heritage is carried to the new generation of individuals. A point is chosen randomly from the points where binary representation of each attribute ends. One of the offspring gets the entries before that point from the first parent and the entries after that from the second parent. The other offspring gets the properties of the attributes after that point from the first parent and the properties of attributes before that point from the second parent. This step is carried out until the number of offspring created is equal to the population size, so the size of the population is doubled at the end of this stage.

MUTATION: *Mutation* in product profiles are used to incorporate a different direction in the search process. It corresponds to making a change in the product profile and creating an individual whose properties are not all inherited from the parents. Each individual in the population undergoes this step, however mutation occurs with a predefined mutation rate or probability. In this algorithm, mutation is done by changing the level of one of the attributes to another level. If a profile is subject to mutation, each attribute has an equal chance of being changed. Similarly, all other levels are equally likely to be selected to be the new level. When an individual is mutated, only the mutated version stays in the population. So, at the end of this phase, the size of the population stays the same.

REDUCTION: The population size is halved by eliminating the individuals with the least market share. The other half of the profiles with higher market share are carried to the next generation.

STOPPING CONDITION: The process is repeated until either there is no significant improvement over multiple generations or a predefined number of iterations is reached.

COMPUTATIONAL RESULTS

In this section, we discuss our computational experience with the GA on a large set of simulated problem instances on simulated and real social network data. Our computational results demonstrate the performance of the GA in terms of near-optimality and computational speed, and the benefits of the SOCSNE model in terms of market share. We also provide an iterative optimal solution method for the LCIP model.

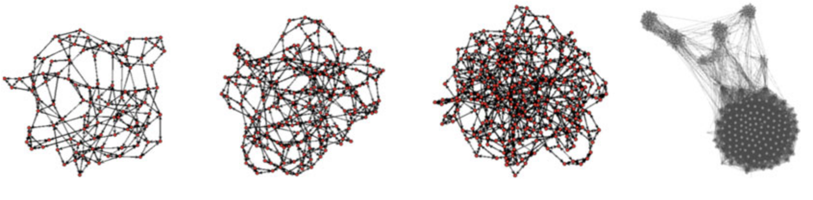
Data Generation

Similar to earlier work on the SOC problem (Kohli & Krishnamurti, 1987; Balakrishnan & Jacob, 1996; Shi et al., 2001; Camm et al., 2006; Wang et al., 2009), we created simulated data sets. In these data sets, part-worth utilities are generated from a uniform distribution that ranges from 0 to 1 and normalized within individuals (Kohli & Krishnamurti, 1987; Nair et al., 1995). In contrast to simulated data sets for the SOC problem that require a single hurdle value for each individual, our modeling approach requires both a high and a low hurdle for each individual in the data set. To represent a heterogeneous preference behavior among the customers, hurdle spans are created as follows: 1,000 product profiles are randomly generated and for every individual these product profiles are ranked in descending order with respect to their total utilities. High hurdle, which could correspond to total utilities of a status-quo product, is randomly selected from the first 500 product profiles of this ordered set and low hurdle is randomly selected from the second half of this ordered set.

Determining the appropriate sample size to represent the whole population in conjoint analysis (which excludes network relations) is a challenge (see Chapter 7, Orme, 2014). In our model, there is the additional complexity in representing a social network with a sample. Unbiased sampling from a social network that will both represent the whole network and carry the same social network properties as the original social network is an active research topic (Gjoka, Kurant, Butts, & Markopoulou, 2010, 2011). However, because the SOCSNE problem would be solved for given conjoint analysis results, we view sampling to occur in practice as a step before running our proposed model.

To obtain the network relations (i.e., graph) between the customers in the SOC problem, we generate three simulated networks that have social network properties and use one real social network from Facebook. The real social network is crawled from Facebook using Gephi 8.2. The network shows the relationship between 404 individuals with 8,743 edges, that is, there is an edge between two nodes if they are connected over Facebook. The creation of simulated social networks is much harder than typical random graph generation. The reason for this is the connection topology for social networks lies somewhere between the two extremes of completely regular and completely random graphs (Watts & Strogatz, 1998). In our test data, we use the small-world network generation model in R programming (R Development Core Team, 2010). In the small-world model every node ends up being only a few (about six) connections away from each other (a property that many real social networks exhibit). To be able to generate such a graph, Watts and Strogatz (1998) proposed the following rewiring method. In this method, the network starts with a ring of n nodes, each connected to its k nearest neighbors by undirected edges. A node and the edge that connects it to its nearest neighbor in a clockwise sense are chosen. With probability p, the edge is reconnected to a node chosen uniformly at random over the entire ring, with duplicate edges forbidden; otherwise the edge is left in place. This is repeated until each edge in the original graph has been considered once. For $p = 0$, the original graph is unchanged; as p increases, the graph becomes increasingly disordered until for $p = 1$, all edges are rewired randomly. Watts and Strogatz (1998) showed that values of p in the range of .1 to .3 correspond more closely to small-world networks; consequently we used $p = .1$, .2, and .3 as the rewiring probabilities when generating the three

Figure 1: Simulated social networks of 100, 200, and 300 people for $p = .2$, and the Facebook network of 404 people, respectively.



social networks with 100, 200, and 300 people. Four of the 10 networks used in the article are illustrated in Figure 1, respectively.

For the linear influence structure used in our computations, for an individual s, each additional buying neighbor decreases the hurdle an equal amount, Δ_s. When the social network is generated and the neighbors are determined for each person, Δ_s is calculated as the ratio of the hurdle span and the number of neighbors. By this definition, if all neighbors of an individual buy the product, the current hurdle for that individual reaches the low hurdle.

Performance Evaluation of the GA

To evaluate the performance of the proposed GA for the SOCSNE model, we compare it with the optimal (exact) solution of the IP model SOCSNE (described earlier) obtained using CPLEX 12.6 on a 3.40-GHz Intel Core i7-3770 Processor with 32.0 GB of RAM. After considerable experimentation on small data sets, the GA parameters for the population size and the mutation rate are set to 100 and 0.3, respectively, and the stopping condition is set as 10 generations. The market shares for the product profiles selected by the GA and IP methods for 90 SOCSNE problems[vii] are shown in Tables 1 and 2.

For simulated networks, information on the rewiring probabilities used to generate the social networks and the product profile are given in the first three columns of Table 1. The remaining columns provide a comparison of market share and running times of the GA and IP approach to solve the SOCSNE model. We should note that other than one instance ($p = .3$, 300 people, 10 attributes with five levels) all other instances are solved to optimality with the IP approach. This one instance was terminated after 13 hours of computational time and the reported solution is the best integer solution found by the IP approach. We use an "*" to denote that the solution found by the IP is not provably optimal. From Table 1 it should be evident that, while the run time of the GA is very robust, the run time of the IP approach rapidly increases. Thus, it should be clear that the IP is not a scalable approach for solving the SOCSNE model. (The run time of the GA is very stable because it evaluates the same number of product profiles in the algorithm regardless of problem size, while the run time of the IP is largely determined by the number of possible product profiles in the problem.) Comparing the market shares for the GA and the IP methods to solve SOCSNE, on the average, the GA obtains 98.44% of the optimal market share while consistently taking no more than about 90 seconds. When we look at the results for the Facebook network in Table 2, a similar behavior is observed. Here the GA always takes less than 2 minutes of running time, while the run time of the IP is as large as 17 hours. (The number of possible product profiles grows faster with the number of levels than the number of attributes. Because the run time of the IP is largely determined by the number of possible product profiles, the effect of the number of levels is more marked than the number of attributes in Table 2.) At the same time, the GA is able to obtain 98.45% of the optimal market share for the Facebook network.

[vii] The complete combination of product profiles of attributes (8,9,10), levels (3,4,5), rewiring probabilities ($p = .1, .2, .3$), and number of people (100,200,300) in the market constitute the 81 data sets. Nine additional data sets with the same number of attributes and levels are tested on the Facebook network. The largest data set (10 attributes and five levels) we have tested has 9,765,625 possible product profiles.

Table 1: Market share comparison of the GA and the exact IP approach for the SOCSNE problem on simulated social networks.

			100 People				200 People				300 People			
			Market Share		Run Time (s)		Market Share		Run Time (s)		Market Share		Run Time (s)	
p	# of Attributes	# of Levels	GA	IP	GA	IP	GA	IP	GA	IP	GA	IP	GA	IP
.1	8	3	82	82	89	1	145	145	85	6	211	214	105	4
	8	4	81	85	58	7	159	159	84	32	219	220	88	64
	8	5	86	86	73	48	155	158	78	159	219	227	89	570
	9	3	78	85	88	2	152	152	75	4	218	221	93	21
	9	4	86	89	78	22	155	157	86	86	229	233	92	202
	9	5	83	87	79	384	166	166	97	445	228	230	88	1,782
	10	3	84	84	87	5	152	152	90	34	234	234	87	46
	10	4	85	86	91	153	155	155	92	274	227	232	95	878
	10	5	84	89	81	2,202	157	159	91	12,517	236	240	87	46,982
.2	8	3	82	82	62	1	154	154	75	2	223	223	91	3
	8	4	85	87	77	7	161	161	84	36	226	226	90	59
	8	5	84	85	57	56	152	154	82	142	238	238	81	227
	9	3	85	85	74	2	153	153	69	7	214	214	79	15
	9	4	85	86	68	28	160	160	81	80	228	232	86	163
	9	5	89	89	66	216	155	161	77	2,210	233	238	87	1,073
	10	3	80	84	68	7	143	143	82	52	219	223	88	54
	10	4	80	83	63	164	156	156	74	260	224	227	89	2,143
	10	5	90	92	63	627	163	164	88	2,455	235	238	95	41,413
.3	8	3	80	81	65	1	158	158	90	2	213	213	88	5
	8	4	83	83	69	8	161	165	80	31	218	222	92	60
	8	5	82	84	61	64	154	156	76	121	222	223	91	283
	9	3	81	82	68	2	143	146	83	7	222	225	93	12
	9	4	89	91	64	24	151	155	84	94	221	223	85	181
	9	5	79	84	61	249	153	156	86	743	234	234	92	1,443
	10	3	80	81	69	8	161	161	89	27	214	218	78	55
	10	4	83	85	63	116	159	162	77	569	222	223	74	747
	10	5	84	88	84	9,604	162	171	80	6,322	213	226*	78	48,093

Table 2: Market share comparison of the GA and the exact IP approach for the SOCSNE problem on the Facebook network.

		Market Share		Run Time (s)	
# of Attributes	# of Levels	GA	IP	GA	IP
8	3	280	280	107	9
8	4	285	292	107	184
8	5	306	306	103	1,677
9	3	274	285	113	45
9	4	294	294	128	521
9	5	297	302	108	10,086
10	3	292	295	109	178
10	4	285	288	119	2186
10	5	291	303	104	62,304

Table 3: Market share comparison of the GA and the exact IP approach for the SOCSNE problem on larger size data sets (for $p = .2$).

		300 People			
		Market Share		Run Time (s)	
# of Attributes	# of Levels	GA	IP	GA	IP
13	5	241	238*	87	56,141
13	6	229	242*	87	48,385
14	5	228	234*	95	35,986
14	6	221	234*	87	41,057
15	5	235	237*	90	37,346
15	6	233	241*	87	95,334

We now turn our attention to a larger data set.[viii] Table 3 provides a comparison of the GA and IP on these larger data sets. The running time of the GA stays around 90 seconds whereas the IP is no longer viable as an exact approach. It can take the IP around 15 hours to find a bound (i.e., none of the solutions found by the IP are provably optimal solutions). Again, we use an * to denote that the solution found by the IP is not provably optimal. Although the GA only takes about 90 seconds, on average, the GA is able to get 97.27% of the best found solution.

Taken altogether, the results in Tables 1–3 indicate that the GA obtains very high-quality solutions (in fact near-optimal solutions in cases where the optimal solution is known) for the SOCSNE model; and is very robust in terms of its running time, scalability, and ability to adapt to additional constraints/variations of the model.

[viii] Product profiles of size (13,14,15) attributes with (5,6) levels for $p = .2$, 300 people.

Table 4: Analyzing network effects and the value of the SOCSNE model for the Facebook network.

# of	# of	IP + NA	GA+NE			GA+IN		
Attributes	Levels	SOC	SOC	SOCSNE	% Inc.	SOC	SOCSNE	% Inc.
8	3	144	234	270	15	243	280	15
8	4	139	235	244	4	245	285	16
8	5	146	230	282	23	255	306	20
9	3	139	223	250	12	225	274	22
9	4	134	222	277	25	208	294	41
9	5	141	246	284	15	246	297	21
10	3	152	237	281	19	242	292	21
10	4	148	255	244	−4	267	285	7
10	5	158	257	259	1	266	291	9
Average		144.56	237.67	265.67	12	244.11	289.33	19

Network Effects on Market Share and the Benefits of the SOCSNE Model

We now provide two types of comparisons to evaluate the network effects on the market share and more importantly to evaluate the benefits of the SOCSNE model (as opposed to using the SOC model to select a product).

We first consider the data sets on the Facebook network (Table 4). For each problem, we consider product profiles chosen with the SOC and SOCSNE model under various scenarios. Either the SOC model or the SOCSNE model may be used to obtain a product profile. Further, the solution method to solve the model could either be the GA or the IP (solved using CPLEX). Next, we consider the market share for a product profile under three scenarios: (i) allow for no network effects "NA," (ii) allow for network effects but no payment of incentives "NE," or (iii) allow for network effects and payments of incentives "IN." For example, "GA+IN" for SOCSNE represents that the SOCSNE model was solved using a GA and the product profile obtained is evaluated allowing for network effects and payments of incentives. Rather than computing market shares under the 12 possible combinations, we focus on a few interesting ones.

Our primary comparison is the market share under the "GA+IN" columns for the product profile chosen under the SOC and SOCSNE models. (We use GA instead of IP because the GA is scalable and the method of choice as the data sets get larger.) The difference in market share here represents the benefit (i.e., increase in market share) by using the SOCSNE model. This comparison shows a 19% increase in market share on average using the SOCSNE model compared to the SOC model to select a product profile. As a matter of curiosity, we provide a similar comparison under the "GA+NE" columns. This comparison shows a significant advantage for the SOCSNE model (a 12% increase in market share) even if one allowed for network effects but no payment of incentives. This suggests that even in the event of a limited budget for marketing incentives, the product profile obtained with the SOCSNE model generally provides a significant increase in market share than the product profile under the SOC model. Finally, the column "IP+NA" provides an estimate of market share if there were no network effects in the marketplace. While this is certainly not the case (i.e., there are always some network effects) in practice, it suggests that marketing products in an environment where users are actively able to share their experiences with friends is likely to generate a significantly larger market share than marketing products in an environment where it may not be easy for users to share their experiences (with the caveat here that our experiments have been in situations where the peer influence effects have always been positive; i.e., lowered the threshold). In our experiments, there is typically a doubling of market share in the environment where there are network effects ("GA+IN") compared to the environment where there are no network effects ("IP + NA").

We now turn our attention to the three simulated data sets (Table 5), comparing the market share under the "GA+IN" columns for the product profile chosen under the SOC and SOCSNE models. The difference in market share here represents the benefit (i.e., increase in market share) by using the SOCSNE model. This comparison shows a 19% increase in market share on average using the SOCSNE model compared to the SOC model to select a product profile. These experiments demonstrate the value of the SOCSNE model in that it enables significantly better identification of products (compared to the SOC model) that yield a larger market share, in an environment where social network effects play a role in one's purchasing behavior.

Solving the LCIP Model: Identifying Individuals to Pay Out Incentives to

The solution to the LCIP identifies the set of people who receive some incentives to buy the product and further influence their neighbors. Even for very small problem sizes, the LCIP model described earlier is computationally intractable to solve. To overcome this problem, we first preprocess the model and then use a tractable, iterative, and much faster approach that preserves optimality (i.e., ensures the LCIP is solved to optimality).

In the LCIP model, the influence spreads through the network over a finite number of time periods (Observation 2) and eventually reaches every node. At the beginning of each period, an individual's hurdle is updated with respect to neighbors' buying decisions. Decisions made in time t affect neighbors' hurdles in period $(t + 1)$.

Preprocessing: As a preprocessing step before solving the LCIP, we identify by simple comparison the nodes whose utilities are already greater than or equal to their hurdles and the cascade of nodes that purchase the product after being influenced from previous buyers *without* requiring any incentives (these nodes constitute the market share under "NE"). Once these nodes are eliminated, the remaining network of nodes (i.e., people that constitute the market share difference between columns GA+IN and GA+NE) are the ones that need an incentive to start a new cascade of buyers.

It is then easier to recast the LCIP as follows. For convenience, we will repeat notation and let V' denote the nodes in the problem after preprocessing, b_s denote the difference between the current hurdle and product utility for $s \in V'$ (observe

Table 5: Analyzing the value of the SOCSNE model over the SOC model by market share comparison of product profiles on simulated social networks.

			100 People GA + IN			200 People GA + IN			300 People GA + IN		
	# of Attributes	# of Levels	SOC	SOCSNE	% Inc.	SOC	SOCSNE	% Inc.	SOC	SOCSNE	% Inc.
$p = .1$	8	3	62	82	32	118	145	23	173	211	22
	8	4	71	81	14	133	159	20	199	219	10
	8	5	61	86	41	142	155	9	191	219	15
	9	3	67	78	16	133	152	14	175	218	25
	9	4	67	86	28	128	155	21	224	229	2
	9	5	66	83	26	122	166	36	201	228	13
	10	3	61	84	38	127	152	20	169	234	38
	10	4	55	85	55	131	155	18	184	227	23
	10	5	63	84	33	129	157	22	197	236	20
	Average		63.67	83.22	31	129.22	155.11	20	190.33	224.56	18
$p = .2$	8	3	74	82	11	126	154	22	190	223	17
	8	4	60	85	42	144	161	12	188	226	20
	8	5	67	84	25	126	152	21	208	238	14
	9	3	73	85	16	127	153	20	193	214	11
	9	4	62	85	37	142	160	13	205	228	11
	9	5	75	89	19	118	155	31	218	233	7
	10	3	69	80	16	122	143	17	188	219	16
	10	4	69	80	16	137	156	14	189	224	19
	10	5	81	90	11	140	163	16	210	235	12
	Average		70.00	84.44	21	131.33	155.22	18	198.78	226.67	14
$p = .3$	8	3	66	80	21	137	158	15	189	213	13
	8	4	62	83	34	141	161	14	184	218	18
	8	5	67	82	22	132	154	17	187	222	19
	9	3	76	81	7	109	143	31	209	222	6
	9	4	72	89	24	132	151	14	194	221	14
	9	5	62	79	27	135	153	13	208	234	13
	10	3	71	80	13	142	161	13	188	214	14
	10	4	69	83	20	145	159	10	208	222	7
	10	5	70	84	20	137	162	18	204	213	4
	Average		68.33	82.33	20	134.44	155.78	16	196.78	219.89	12

$b_s > 0$ for $s \in V'$), and $d_{js} = a'_{js}\Delta_s$ denote the influence of node j on node s if node j adopts the product. The LCIP can then be rewritten as follows:

$$\textbf{LCIP}: \quad \text{Minimize} \sum_{s \in V'} z_s, \tag{19}$$

$$\text{subject to } z_s + \sum_{j \in V'} d_{js} y_{j(t-1)} \geq b_s y_{st} \quad \forall s \in V', \forall t \geq 1, \tag{20}$$

$$y_{s0} = 0 \quad \forall s \in V', \tag{21}$$

$$\text{Constraints} \quad (13), (14), (15), \text{ and } (16).$$

Iterative approach: Our iterative approach further reduces the size of the LCIP by limiting the number of time periods to obtain an initial solution, and then incrementing the number of periods by one at each iteration. The rationale behind adding a single time period at each iteration is to identify the potential network effects at that period. When the problem is solved in a single period, network effects cannot take place and every individual is fully paid incentives (i.e., they are paid the difference between the utility of the product and the current hurdle) to adopt the product. In the next iteration, the LCIP model is re-solved with two time periods. Here the network effects are present but limited to only the "first"- and "second"-generation buyers. In every successive iteration, the LCIP is solved from the beginning. Note that the solutions of the previous iteration for the incentives are not used in the current iteration, that is, each problem instance is solved to optimality without assuming partial solutions from the previous solution. The iterations are continued until the total amount of incentives given (i.e., the objective function) stays the same as the cost of the previous iteration. This process leads us to the following three observations.

Observation 3: The cost of total incentives can never be worse than costs in the previous iteration.

As time periods are added, the network effects cascade further into the network and compensate some of the incentives, that is, the amount of incentives required to persuade individuals to buy the product decreases and therefore the optimal solution decreases.

Observation 4: The costs can decrease only when there is at least one new buyer in a period.

The decrease in the amount of incentives can only be covered by the influence of new buyers in the previous period.

Observation 5: The iteration where the objective stays the same as the previous iteration is the **optimal solution**.

No improvement in the objective value shows that there is no influence taking place at that iteration, that is, it is not necessary to increase the number of time periods anymore.

Table 6: LCIP results for the Facebook network.

# of Attributes	# of Levels	# of New Buyers	# of People with Incentives	# of Iterations	Return
8	3	10	6	6	1.67
8	4	41	18	6	2.28
8	5	24	10	8	2.40
9	3	24	15	12	1.60
9	4	17	10	8	1.70
9	5	13	6	8	2.17
10	3	11	5	6	2.20
10	4	41	8	13	5.13
10	5	32	16	13	2.00

Computational results for the LCIP model

We analyze the LCIP Model in a couple of different ways. First, we consider the ratio between the number of people who decide to buy as a result of incentives being provided in the model (i.e., the difference between the "NE" and "IN" scenarios) and the number who receive incentives, and represent it as "Return." The ratio represents the average number of additional buyers obtained for each person that has been provided incentives. Second, to evaluate the efficacy of the proposed solution method for the LCIP, we consider the number of iterations required to solve the LCIP problem.

Table 6 provides results for the Facebook network and Table 7 provides results for the simulated graphs for $p = .2$. Note that the number of new buyers in column 3 of Table 6 is equal to the difference between columns "GA+NE" and "GA+IN" for SOCSNE in Table 4. This number is the size of the set of buyers who purchase the product after a subset of the network has been given incentives. Overall, the average return for the Facebook network is 2.35 (i.e., for each individual provided incentives there are 2.35 new buyers), and the LCIP is solved on average in 8.89 iterations. For the 27 instances for the simulated network in Table 7, the average return is 2.48 and the LCIP is solved on average in 6.15 iterations.

Social Welfare Comparison

With numerical studies we have shown that the market share increases when products are designed in a way that allows incentives to be provided. However, incentives and the increase in the market share are not measured in the same units, perhaps making it somewhat hard to analyze the trade-off. To ease the comparison, we use a welfare measure similar to the utilitarian function (selecting a product on the basis of the sum of utilities) as in Gupta and Kohli (1990). They refer to the problem of finding the product profile that results in the largest overall consumer utility as the "buyer's welfare problem." Social welfare is the summation of all individual buyer utilities from the product, where each buyer utility is calculated as the difference between utility from the product and the current hurdle of the individual.

Table 7: LCIP results for simulated network with $p = .2$.

		100 People				200 People				300 People			
# of Attributes	# of Levels	# of New Buyers	# of People with Incentives	# of Iterations	Return	# of New Buyers	# of People with Incentives	# of Iterations	Return	# of New Buyers	# of People with Incentives	# of Iterations	Return
8	3	31	11	5	2.82	16	5	5	3.20	36	14	6	2.57
8	4	6	2	4	3.00	28	13	5	2.15	68	21	9	3.24
8	5	14	5	4	2.80	37	19	7	1.95	33	10	7	3.30
9	3	11	5	5	2.20	26	12	5	2.17	38	17	7	2.24
9	4	29	10	9	2.90	11	3	4	3.67	44	24	7	1.83
9	5	14	7	5	2.00	39	19	10	2.05	49	18	9	2.72
10	3	12	5	3	2.40	35	15	7	2.33	45	19	8	2.37
10	4	31	14	5	2.21	36	16	8	2.25	26	12	3	2.17
10	5	18	11	6	1.64	31	15	6	2.07	52	20	7	2.60

Table 8: Effects of incentives on social welfare, for simulated networks.

		100 People				200 People				300 People			
		Social Welfare				Social Welfare				Social Welfare			
# of Attributes	# of Levels	GA + NE	GA + IN	GA + IN Incentives	Return	GA + NE	GA + IN	GA + IN Incentives	Return	GA + NE	GA + IN	GA + IN Incentives	Return
8	3	42.51	77.59	1.60	21.90	118.58	131.73	0.45	29.23	150.63	187.13	1.85	19.74
8	4	77.05	84.29	0.10	76.00	121.91	149.31	1.01	27.04	128.89	201.66	2.35	31.02
8	5	61.63	76.55	0.48	30.80	101.87	138.79	2.36	15.66	203.03	236.17	1.04	31.98
9	3	79.45	89.71	0.70	14.64	106.86	127.51	1.51	13.68	153.25	191.20	1.35	28.19
9	4	52.82	77.84	0.83	30.17	144.85	158.52	0.11	128.60	178.01	226.01	2.65	18.13
9	5	71.21	85.35	0.45	31.45	112.71	153.14	2.73	14.83	176.60	232.83	2.51	22.39
10	3	59.36	74.16	0.40	36.83	95.32	130.19	1.58	22.12	158.35	201.22	1.64	26.10
10	4	52.18	84.67	1.48	21.95	112.44	154.57	1.61	26.19	205.61	235.24	1.52	19.52
10	5	95.24	114.47	1.68	11.44	140.18	175.50	1.72	20.49	194.47	264.87	1.60	43.88

Table 9: Effects of incentives on social welfare for the Facebook network.

# of Attributes	# of Levels	Social Welfare GA + NE	Social Welfare GA + IN	GA + IN Incentives	Return
8	3	209.94	221.74	0.16	75.42
8	4	188.52	216.24	1.28	21.66
8	5	223.00	242.21	0.62	30.82
9	3	179.53	198.38	0.75	25.23
9	4	209.46	225.19	0.24	65.55
9	5	243.01	257.33	0.29	50.22
10	3	246.32	258.59	0.20	62.74
10	4	214.40	248.52	0.67	50.87
10	5	242.41	276.08	1.37	24.65

Table 8 and Table 9 describe the impact of incentives on social welfare for the simulated data sets with $p = .2$ and the Facebook network, respectively. The impact is shown by stating the social welfare for the "GA+NE" and "GA+IN" solutions. Their difference provides the increase in social welfare. This difference divided by the amount of incentives provided to obtain the "GA+IN" solution provides the "Return" on the incentives provided. For the simulated data sets in Table 8, the average "return on incentives" is 30.15, while for the Facebook network in Table 9, the average "return on incentives" is 45.24.

SUMMARY AND CONCLUDING REMARKS

We proposed a novel model to include peer influence effects in product design within the framework of the SOC problem. Although the SOC problem has been studied in the marketing literature, to our knowledge we are the first to explicitly consider peer influence effects in the SOC model. By taking into account peer influence effects, one is able to design products with far larger market shares than obtained by the original SOC model. While the effects of peer influence on consumer choice are well documented, previous analysis of conjoint data typically assumed that a consumer's attribute preferences and product choices are independent of choices of others. Narayan et al. (2011) took a significant first step in the development of conjoint estimation models that incorporate peer influence. The SOCSNE model introduced in this article allows one to successfully use the results of such conjoint estimation models toward a logical next step—the design of a product with largest market share.

The model we constructed remains a computationally challenging NP-Hard problem. We developed a solution method that integrates an exact mathematical model within a GA to solve the product design problem. The model with network effects is complicated due to the dynamic relationship of the nodes and hence we use a matheuristic approach (Maniezzo et al., 2009) that integrates an optimal fitness evaluation (via an integer program) within the GA. After extensive computational studies, we show that the GA is robust, finds high-quality solutions for the simulated data sets, and preserves a running time around less than 2 minutes independent of the size of the problem. As one of the characteristics, it is flexible with extensions requiring only minor modifications to the algorithm.[ix]

As the next step, we focused on the operational problem of achieving the desired market share predicted by the SOCSNE model. This requires intervention in the diffusion/product adoption process by providing tailored incentives to a group of individuals in the network. We work on the product design and product diffusion as two separate problems for two reasons. First, product design is typically done using utility information from a *sample*[x] via conjoint analysis. In contrast, the LCIP modeling approach can be applied to the entire network of the targeted market (i.e., it need not be restricted to the sample). Second, the product design problem is a strategic decision, whereas the LCIP is an operational problem. In other words, the profile design decision is usually a rare (if not a one-time) decision, but the LCIP can be applied repeatedly as part of the ongoing operational decisions. Engineering the diffusion of a product has already been used by businesses in the form of free samples or coupons. Because the LCIP model is computationally very hard to solve, we develop a preprocessing procedure and a simple iterative strategy to solve it. As noted earlier, the LCIP is applicable in a larger number of settings. Further, social network structures are dynamic and may change over time. Thus, with a product (or product line) in place a marketer could benefit from the LCIP model to analyze the trade-offs and incentives required to reach a desired fraction of the population operationally during the marketing phase of the product. Although the LCIP model considers incentives in terms of customer utilities, one should note that following Miller, Hofstetter, Krohmer, and Zhang (2011), it is possible to infer dollar values from customer utilities.

There are several natural directions for future research. Broadly, they all encompass the expansion of product or product line design problems to settings with social network or peer influence effects. They include (i) pricing of the levels of attributes for a product as part of the design process and maximizing revenue as an alternate objective, (ii) consideration of alternate assumptions of product selection amongst consumers (i.e., instead of the highest choice pattern where each consumer deterministically self-selects the product from the line that provides the highest surplus one could consider the multinomial choice logit model (Chen & Hausman, 2000)), and (iii) extension of the problem to stochastic network settings (i.e., one in which the social network changes over time).

[ix] For example, if the product design decisions are made in the context of budgetary constraints, especially if the costs are in terms of man-power or time, or the focus is a prestated design objective, we add the constraint $\sum_{k=1}^{K}\sum_{l=1}^{L_k} c_{kl}x_{kl} \leq B$ to the SOCSNE model. The design cost associated with level l of attribute k is given by c_{kl} and the total budget is B. In this way, the total cost of the selected levels for the attributes will not exceed a given design budget. In modifying GA, the fitness of a product would be evaluated using the SOCSNE model, however now the model would include this budget constraint.

[x] This raises interesting questions as to how to sample from a social network so that the structure of the sample network resembles the original one. This is an open area of recent research in computer science (Leskovec & Faloutsos, 2006).

REFERENCES

Aral, S., & Walker, D. (2011). Creating social contagion through viral product design: A randomized trial of peer influence in networks. *Management Science*, *57*(9), 1623–1639.

Balakrishnan, P. V., & Jacob, V. S. (1996). Genetic algorithms for product design. *Management Science*, *42*(8), 1105–1117.

Bearden, W. O., & Etzel, M. J. (1982). Reference group influence on product and brand purchase decisions. *Journal of Consumer Research*, *9*(2), 183–194.

Bell, D. R., & Song, S. (2007). Neighborhood effects and trial on the internet: Evidence from online grocery retailing. *Quantitative Marketing and Economics*, *5*(4), 361–400.

Belloni, A., Freund, R., Selove, M., & Simester, D. (2008). Optimizing product line designs: Efficient methods and comparisons. *Management Science*, *54*(9), 1544–1552.

Bourne, F. S. (1957). Group influence in marketing and public relations. In R. Likert & S. P. Hayes (Eds.), *Some applications of behavioral research*. Switzerland: UNESCO, 207–225.

Burnkrant, R. E., & Cousineau, A. (1975). Informational and normative social influence in buyer behavior. *Journal of Consumer Research*, *2*(3), 206–215.

Camm, J.D., Cochran, J.J., Curry, D.J., & Kannan, S. (2006). Conjoint optimization: An exact branch-and-bound algorithm for the share-of-choice problem. *Management Science*, *52*(3), 435–447.

Candogan, O., Bimpikis, K., & Ozdaglar, A. (2012). Optimal pricing in networks with externalities. *Operations Research*, *60*(4), 883–905.

Chen, K. D., & Hausman, W. H. (2000). Mathematical properties of the optimal product line selection problem using choice-based conjoint analysis. *Management Science*, *46*(2), 327–332.

Childers, T. L., & Rao, A. R. (1992). The influence of familial and peer-based reference groups on consumer decisions. *Journal of Consumer Research*, *19*(2), 198–211.

Cohen, M., & Harsha, P. (2016). Designing price incentives in a network with social interactions, available at SSRN: http://ssrn.com/abstract=2376668.

Dou, Y., Niculescu, M.F., & Wu, D.J. (2013). Engineering optimal network effects via social media features and seeding in markets for digital goods and services. *Information Systems Research*, *24*(1), 164–185.

Gjoka, M., Kurant, M., Butts, C. T., & Markopoulou, A. (2010). Walking in facebook: A case study of unbiased sampling of OSNs. *INFOCOM, 2010 Proceedings IEEE*, San Diego, CA: 1–9.

Gjoka, M., Kurant, M., Butts, C. T., & Markopoulou, A. (2011). Practical recommendations on crawling online social networks. *IEEE Journal on Selected Areas in Communications*, *29*(9), 1872–1892.

Godes, D., Mayzlin, D., Chen, Y., Das, S., Dellarocas, C., Pfeiffer, B., Libai, B., Sen, S., Shi, M., & Verlegh, P. (2005). The firm's management of social interactions. *Marketing Letters*, *16*(3-4), 415–428.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Granovetter, M. (1978). Threshold models of collective behavior. *American Journal of Sociology*, *83*(6) 1420–1443.

Green, P. E., & Krieger, A. M. (1989). Recent contributions to optimal product positioning and buyer segmentation. *European Journal of Operational Research*, *41*(2), 127–141.

Green, P. E., & Rao, V. R. (1971). Conjoint measurement for quantifying judgmental data. *Journal of Marketing Research*, *8*(3), 355–363.

Gunnec, D. (2012). Integrating social network effects in product design and diffusion. PhD thesis, University of Maryland. Available at http://hdl.handle.net/1903/13103.

Gupta, S., & Kohli, R. (1990). Designing products and services for consumer welfare: Theoretical and empirical issues. *Marketing Science*, *9*(3), 230–246.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI: The University of Michigan Press.

Iyengar, R., Han, S., & Gupta, S. (2009). Do friends influence purchases in a social network? Working Paper 09-123, Harvard Business School, Boston, MA.

Iyengar, R., Van den Bulte, C., & Valente, T. W. (2011). Opinion leadership and social contagion in new product diffusion. *Marketing Science*, *30*(2), 195–212.

Katz, M. L., & Shapiro, C. (1985). Network externalities, competition, and compatibility. *The American Economic Review*, *75*(3), 424–440.

Kempe, D., Kleinberg, J., & Tardos, É. (2003). Maximizing the spread of influence through a social network. *Proceedings of the 9 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC: 137–146.

Kohli, R., & Krishnamurti, R. (1987). A heuristic approach to product design. *Management Science*, *33*(12), 1523–1533.

Kohli, R., & Krishnamurti, R. (1989). Optimal product design using conjoint analysis: Computational complexity and algorithms. *European Journal of Operational Research*, *40*(2), 186–195.

Leenders, R. T. A. J., van Engelen, J. M. L., & Kratzer, J. (2003). Virtuality, communication, and new product team creativity: A social network perspective. *Journal of Engineering and Technology Management*, *20*(1–2), 69–92.

Leskovec, J., & Faloutsos, C. (2006). Sampling from large graphs. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Philadelphia, PA: ACM, 631–636.

Manchanda, P., Xie, Y., Youn, N. (2008). The role of targeted communication and contagion in product adoption. *Marketing Science*, *27*(6), 961–976.

Maniezzo, V., Stützle, T., & Voß, S. (2009). *Matheuristics: Hybridizing metaheuristics and mathematical programming*. New York: Springer Verlag.

Michalewicz, Z. (1996). *Genetic algorithms+ data structures= evolution programs*. UK: Springer-Verlag London.

Miller, K. M., Hofstetter, R., Krohmer, H., & Zhang, Z. J. (2011). How should consumers' willingness to pay be measured? An empirical comparison of state-of-the-art approaches. *Journal of Marketing Research*, *48*(1), 172–184.

Nair, S. K., Thakur, L. S., & Wen, K. W. (1995). Near optimal solutions for product line design and selection: Beam search heuristics. *Management Science*, *41*(5), 767–785.

Narayan, V., Rao, V. R., & Saunders, C. (2011). How peer influence affects attribute preferences: A Bayesian updating mechanism. *Marketing Science*, *30*(2), 368–384.

Orme, B. (2014). *Geting started with conjoint analysis* (3rd edition). Madison, WI: Research Publishers LLC.

R Development Core Team. (2010). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. URL http://www.R-project.org. ISBN 3-900051-07-0.

Sawtooth Software. (2003). Advanced simulation module for product optimization v1.5. Tech. rep., Sawtooth Software, Inc., Sequim, WA.

Shi, L., Ólafsson, S., & Chen, Q. (2001). An optimization framework for product design. *Management Science*, *47*(12), 1681–1692.

Sosa, M. E., Eppinger, S. D., & Rowles, C. M. (2004). The misalignment of product architecture and organizational structure in complex product development. *Management Science*, *50*(12), 1674–1689.

Srinivasan, R., Lilien, G. L., & Rangaswamy, A. (2004). First in, first out? The effects of network externalities on pioneer survival. *Journal of Marketing*, *68*(1), 41–58.

Story, L., & Stone, B. (2007). Facebook retreats on online tracking. http://tinyurl.com/4358cev. *The New York Times*.

Szymanski, D. M., Bharadwaj, S. G., & Varadarajan, P. R. (1993). An analysis of the market share-profitability relationship. *Journal of Marketing*, *57*(3) 1–18.

Wang, X. J., Camm, J. D., & Curry, D. J. (2009). A Branch-and-Price approach to the share-of-choice product line design problem. *Management Science*, *55*(10), 1718–1728.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of small-world networks. *Nature*, *393*(6684), 440–442.

Zufryden, F. S. (1977). A conjoint measurement-based approach for optimal new product design and market segmentation. In A. D. Shocker (Eds.), *Analytical approaches to product and market planning*. Cambridge, MA: Marketing Science Institute, 110–114.

APPENDIX A: WHY DO SOCIAL NETWORK EFFECTS MATTER IN PRODUCT DESIGN?

When social network effects are present, taking them into account in the SOC model can lead to product designs with a significantly higher market share. The following example is constructed to emphasize how large the difference can be among market shares with and without social network effects taken into account in the product design process. Consider a simple social network with three people where all customers are connected to each other, that is, a fully connected network with three nodes. The problem is to design a product with a single attribute and maximize the market share. For simplicity, assume that there are only two possible levels for this attribute, level 1 and level 2. The corresponding data on utility, high hurdle, and Δ values for each person are provided in Table 10.[xi] The utilities are 200, 320, 205 for using level 1; 280, 280, 240 for using level 2, and the high hurdles are 300, 320, 250 for the three people, respectively. So person 1 would prefer level 2 to level 1 but would not buy the product in either case because their hurdle is larger than their utilities from both levels. Again for simplicity, let Δ_s values be the same for each person, 20, that is, for each additional neighbor purchasing the product, a person's hurdle decreases by 20. When the original SOC formulation (without the social network effects) is solved, it is easy to see by simply comparing utilities with hurdles that the optimal solution is to choose level 1; the market share is 1 and only person 2 purchases the product. Although the social network effects have not been taken into account in the design phase, if they are allowed after the product is launched, the purchase by person 2 decreases the hurdles for person 1 and 3. Their current hurdles become 280 and 230 but still are greater than their utilities (insufficient to induce them to buy the product). So the market share does not change. When the problem is solved taking social networks into account in the design phase (SOCSNE), the optimal solution is to choose level 2 and the market share is 3 where all three people buy the product. In this case, each hurdle is lowered by $20 \times 2 = 40$ because two neighbors of each person are purchasing the product. Now the comparison ($300 - 40 = 260$ vs. 280, $320 - 40 = 280$ vs. 280 and $250 - 40 = 210$ vs. 240) shows that all hurdles are less than or equal to the utilities. In this example, neglecting the social network effects in the product design leads to a solution that is three times worse than the solution obtained by

Table 10: Utility, hurdle, and Δ_s values for the example.

Person	Utilities		High Hurdle	Δ_s
	Level 1	Level 2		
1	200	280	300	20
2	320	280	320	20
3	205	240	250	20

[xi] We should note that these data are generated only for this example and does not represent the data range used for the computational studies, which is explained in the article.

the model that includes social network effects. This example can be generalized (simply add customers to the market identical to person 3) to show that in the worst case, the loss of market share when social network effects are ignored can be as large as the size of the market!

APPENDIX B: APPLICATION OF THE LCIP IN EPIDEMIOLOGY AND A SLIGHT GENERALIZATION

Although the LCIP has been framed in a product diffusion setting, it can also be equivalently viewed in an epidemiological setting. Suppose that e_{js} denotes the risk factors or influence of untreated node j on node s (e.g., if δ_{js} denotes the probability of node s getting infected by untreated node j, then $e_{js} = -\log(1 - \delta_{js})$). Let f_{js} denote the reduction of influence of node j on node s if node j is treated so that its risk level is less than or equal to a threshold risk level r_j. In other words, if node j is treated so that its risk level is below r_j, its influence on node s is $e_{js} - f_{js}$. We would like to ensure that the sum of all $e_{js} - f_{js}$'s for node s minus the intervention or treatment strategy z_s reduces the overall risk of node s below the threshold risk level r_s. This may be equivalently cast in the marketing setting with $b_s = -r_s + \sum_{j \in V'} e_{js}$ and $d_{js} = f_{js}$, with a discrete set of intervention or treatment strategy choices at each node (with associated costs). The LCIP in the epidemiological setting is then the problem of finding a least cost treatment plan to ensure that a given population has its risk levels for a particular epidemic reduced to below a target threshold level for each member of the population. In the epidemiological setting, once the risk level of a node is reduced below a threshold risk level (unless the neighbors of the nodes or their influences on it change), it is viewed as being safe, and no repeated intervention is therefore necessary unless the structure of the network or the influences of the neighboring nodes change and increase over time.

This illustrates that the LCIP is an extremely useful model in a social network setting where the behavior of one's immediate neighbors influences one's own, and it is of interest to understand how "information" spreads through a network. In particular, it is of interest to understand the power of nodes in a network in terms of their relative "influence" in helping spread (or stop the spread) the "information" over a network.

APPENDIX C: PRODUCT LINE DESIGN

Product line design is an important problem companies face when they want to create a selection of products to appeal to heterogenous consumer segments in the market. Such products may be manufactured goods as well as a service good such as a cell phone plan. The product-line design problem with social network effects is more complicated than the single product design problem because the peer influence effects among users of the product would have to be modeled differently. We model this influence in two levels as first- and second-order peer influence effects and introduce a second-order effect, Δ_s^2, which includes network effects from buyers of other products in the product-line which is of smaller magnitude than the first-order effect, Δ_s^1 (which is the network effect from buyers of the identical product). So the influence on person s from their neighbors should now be represented with the following, where y_{jq} equals 1 if person j buys the q-th product in the product line. (Δ_s^2 is subtracted from Δ_s^1 to eliminate double counting of the effect from the same consumer.)

$$\left(\Delta_s^1 - \Delta_s^2\right) \sum_{j \in V} a_{js} y_{jq} + \Delta_s^2 \sum_{j \in V} a_{js} \sum_{l \in P} y_{jl}. \tag{C1}$$

The first term represents the network effects from purchasers of the same product, while the second term represents the network effects from purchasers of other products in the same product line. Here P represents the products selected in the product line. This influence should be subtracted (as in Equation (6)) from the high hurdle value of person s.

This further complicates both the product design and product diffusion problems. We elaborate on this briefly. Before a purchase decision is made, the utility one gets from a product are dynamic under peer influence. This fluctuation (change) in the utilities due to peer influence prevents one from making a list of preference ordering among the products before solving the product line design problem, such as the one Belloni et al. (2008) used in their model (consequently their model cannot be used in this setting). Wang et al. (2009) also model the product-line design problem for the SOC problem without requiring an a priori preference ordering among the product profiles for the consumers, however, their model does not require identification of the highest utility product. Determining the highest utility product is important in the setting where peer influences vary across products. Thus, the IP model must incorporate the peer influence effects and ensure the customer picks the product with the highest utility. The model that takes these issues into account is weak (in the sense that the linear programming relaxation provides a bound that is quite far from the optimal integer solution) and is not computationally viable other than for extremely small problems. On the other hand the GA approach easily extends to the product line setting. (The main modification deals with extending the representation to allow for a family of products in the product line.) The LCIP also needs to be dealt with in the product-line setting. However, it turns out with appropriate preprocessing the problem can be cast in a similar fashion to the single product setting, and thus the solution procedure discussed in the article for the single product setting can be applied. Further experience and discussion on the product line problem are in Gunnec (2012).

Dilek Gunnec is an assistant professor of Industrial Engineering at Ozyegin University, Istanbul. She holds a PhD degree in Operations Management/Management Science from the Smith School of Business at the University of Maryland, MSc degree in Industrial Engineering from Koc University, and a BSc degree in Mathematics with a minor degree in Operations Research from Middle East Technical University. Her research interests are mainly in modeling, analysis, and computational studies of network systems optimization problems with applications in

information propagation over social networks, new product development, and humanitarian logistics.

S. Raghavan is a professor of Management Science and Operations Management at the Smith School of Business at the University of Maryland. He also holds a joint appointment at the Institute for Systems Research within the Clark School of Engineering. His research interests and activities cover a broad domain including auction design, computational marketing, data mining, economics, information systems, operations management, logistics, networks, and optimization. He has won numerous awards for his research and teaching.