MANUEL CHICA and WILLIAM RAND*

Marketers must constantly decide how to implement word-of-mouth (WOM) programs, and a well-developed decision support system (DSS) can provide them valuable assistance in doing so. The authors propose an agent-based framework that aggregates social network–level individual interactions to guide the construction of a successful DSS for WOM. The framework presents a set of guidelines and recommendations to (1) involve stakeholders, (2) follow a data-driven iterative modeling approach, (3) increase validity through automated calibration, and (4) understand the DSS behavior. This framework is applied to build a DSS for a freemium app in which premium users discuss the product with their social network and promote its viral adoption. After its validation, the agent-based DSS forecasts the aggregate number of premium sales over time and the most likely users to become premium in the near future. The experiments show how the DSS can help managers by forecasting premium conversions and increasing the number of premiums through targeting and implementing reward policies.

Keywords: word of mouth, marketing decision support systems, agent-based modeling, targeting and referrals, freemium business model

Online Supplement: http://dx.doi.org/10.1509/jmr.15.0443

Building Agent-Based Decision Support Systems for Word-of-Mouth Programs: A Freemium Application

Imagine that a brand manager comes up with an idea to send out a designer, one-of-a-kind T-shirt to her most valuable customers. She hopes that customers who receive these T-shirts will talk about them with their friends, and those friends will be more likely to become customers themselves. She knows that word of mouth (WOM) can be a powerful force for marketing (Trusov, Bucklin, and Pauwels 2009), and she hopes that she can utilize this force for the benefit of her brand.

However, she runs into a stumbling block when she begins to think about the plan logistics. Being part of a data-driven organization, she needs to answer several questions to justify this marketing policy to her superiors. How does she balance this WOM program with her traditional marketing mix (Libai, Muller, and Peres 2013)? How much should the rewards cost to maximize revenue (Ryu and Feick 2007), and how should she balance these costs with the number of customers who receive the reward (Schlereth et al. 2013; Stonedahl, Rand, and Wilensky 2010)? Which customers should she target (Hinz et al. 2011; Van der Lans et al. 2010)? Should she target influential users on social media (Hinz et al. 2011; Trusov, Bodapati, and Bucklin 2010; Watts and Dodds 2007), or should she reward only the highest-revenue customers (Haenlein and Libai 2013)? Moreover, what will the effect of this promotion be on noncustomers (Schmitt, Skiera, and Van den Bulte 2011)?

This scenario is not fictional but one faced by managers on a regular basis. Indeed, Blizzard Entertainment faced a very similar set of questions when it sent statues of orcs from its massively multiplayer online role-playing game, *World of Warcraft*, to all customers who had been playing the game for

*Manuel Chica is Chief Technology Officer, RØD Brand Consultants, and Senior Researcher, Department of Computer Science, Open University of Catalonia (e-mail: mchica@rod.com.es). William Rand is Assistant Professor of Marketing, Business Management Poole College of Management, North Carolina State University (e-mail: wmrand@ncsu.edu). This work is supported by Spanish Ministerio de Economía y Competitividad under the NEWSOCO project (ref. TIN2015-67661-P), including European Regional Development Funds (ERDF). Coeditor: Randy Bucklin; Associate Editor: Jacob Goldenberg.

ISSN: 0022-2437 (print)
1547-7193 (electronic)

Journal of Marketing Research
Vol. LIV (October 2017), 752–767
DOI: 10.1509/jmr.15.0443

more than ten years (Pitcher 2015). Airline and hotel companies often consider these questions when designing special rewards for their loyalty programs (Terblanche 2015; Xie et al. 2015). These questions are also relevant for new business models, such as "freemium" apps, whose area of interest centers on encouraging nonpaying users to adopt paid content (Kumar 2014). Of course, managers could spend time examining the previous theory and research on the various effects of WOM on product adoption and market expansion (Libai, Muller, and Peres 2013; Trusov, Bodapati, and Bucklin 2010), but what they really need is a practical tool that embodies this theory and provides them with direct answers to help them make decisions.

The goal of this article is to provide a general framework for the creation of such a tool that can answer these questions. We also apply this framework to a real product, *Creature Party*, a multiplayer online freemium game for kids created by CrazyGames.[1] These questions about WOM are difficult to answer using traditional methods of analysis because WOM models must take into account (1) the heterogeneity of consumer network positions, (2) interactions between people over time, and (3) complex incentive and adoption rules (Goldenberg, Libai, and Muller 2001). As a result, answering these questions requires an individual-level model that gives the analyst the ability to represent the behavior of every customer and the interactions between customers, which is difficult to do using many traditional forms of modeling.

A powerful solution to this problem is to use agent-based modeling (ABM) (Epstein 2006; Macal and North 2005), a computational approach in which every individual can be represented separately and the entire context of one's decision, including one's social network and adoption preferences, can be taken into consideration. The advantage of ABM is that researchers create the model at the individual level, which does not require knowledge of higher-level assumptions. Rand and Rust (2011) describe a set of indicators to consider when deciding whether ABMs are more appropriate than other tools such as analytical or statistical modeling. These indicators show that WOM-related marketing problems are effectively examined through the lens of ABMs. Agent-based modeling is more appropriate than other quantitative tools when a complex and dynamic environment, such as a social network, is involved and when the marketing measure of interest is an emergent result of consumer interactions, such as new conversions or revenue (Rand and Rust 2011). Agent-based modeling also works well when the marketing research questions emphasize the heterogeneity of customers and when the decision processes of those customers can be affected by different individual characteristics, seasonal behavior, media consumption, and the number and type of friends with whom they discuss the brand or the product.

Rand and Rust (2011) note that the patterns of growth in the market that result from consumer interactions are more complex than any one person's adoption decision. Because of this complexity, marketing studies are increasingly using ABM when analyzing new product growth (Delre et al. 2010; Garcia 2005), marketing adoption policies (Libai, Muller, and Peres 2013; Trusov, Rand, and Joshi 2013), and targeting strategies (Haenlein and Libai 2013). Traditionally, many ABM models in the marketing literature have been used to advance marketing theory (Goldenberg, Libai, and Muller 2001, 2010; Watts and Dodds 2007), but ABMs can be calibrated using real data and then can be used to develop insight into real-world applications (Stonedahl and Rand 2014). In this sense, a decision support system (DSS) can be built using ABM to help managers make real tactical and strategic decisions about marketing programs. The realism of ABM facilitates the understanding of the model and can make the DSS more comprehensible to stakeholders, because the model creates an ontology that is very close to the real world.

In this article, we explore a framework for creating an agent-based DSS to provide marketers and researchers with a new and powerful tool to help make WOM decisions and to better understand WOM phenomena. Using this framework, we can situate customers (users) within a social network and give them their own individual states and actions (Wilensky and Rand 2015), and we can use this agent-based DSS to answer many of the questions of our brand manager from the beginning of this article. Specifically, an agent-based DSS can assist marketers and managers in (1) understanding adoption dynamics and customer engagement (the way other people are affected by the engaged customers, directly or indirectly), (2) leveraging customer-to-customer interactions to improve business performance, and (3) testing and evaluating the effects of WOM and social value on revenue and product adoption in hypothetical market scenarios. The agent-based DSS can create market-level outcomes by allowing the incorporation of individual behavioral rules (Libai et al. 2010) through a computational social network, which is representative of customers' real social network (Newman, Barabási, and Watts 2006). These rules describe the typical activity of a customer and how (s)he decides to adopt products or services by using diffusion information models (Rogers 2003) such as those based on cascade models (Goldenberg, Libai, and Muller 2001) and personal thresholds (Granovetter 1978; Watts and Dodds 2007).

Since the time of Little (1970), there has been a robust set of marketing models and DSSs for marketing (Lilien, Roberts, and Shankar 2013; Wierenga, Van Bruggen, and Althuizen 2008). Some DSSs were specifically built for modeling frameworks and decision-making processes in WOM (De Bruyn and Lilien 2008; Lovett, Peres, and Shachar 2013), electronic online WOM activities (Cheung and Thadani 2012; Dellarocas 2006), and viral marketing (Van der Lans et al. 2010). This previous research has shown that DSSs provide managerial benefits such as improving marketers' decision making, enabling the exploration of more decisions, and updating decision makers' mental models (Lilien 2011). However, many successful academic marketing models have a low level of practical use (Lilien 2011). Although models can produce significant benefits, many managers are reluctant to use them based solely on their objective quality in academic publications. In 2004, John Little already noted that "most failures come from trying to deploy sophisticated, black-box optimization models in DSS environments because managers were unwilling to implement recommendations they did not understand" (Little 2004, pp. 1857–58). According to Lilien (2011), researchers must reduce the gap between the users' mental models and implement decision models, which means helping DSS users understand and internalize the factors driving the model results and its managerial recommendations.

[1]The names of the company and the game have been anonymized, but this article is based on real experiences interacting with a real company and game.

Building on this research, our framework emphasizes the creation of an agent-based DSS that encourages the participation of all stakeholders in the model-creation process and constructs the model iteratively to allow for feedback from the stakeholders along the way (Voinov and Bousquet 2010). Moreover, our framework favors computational methods that facilitate the understanding of the models and data-driven validation that allows both modelers and marketers to gain confidence in the DSS recommendations (Oliva 2003; Sargent 2005). Specifically, our study presents a set of methodological guidelines, steps, and decisions to generate the models in the context of large data sets (Leeflang et al. 2015), with a clear focus on the managerial adoption of the results. A decisive step for ensuring this managerial adoption is its validation and testing because decision makers are often concerned with whether each model and its results are correct (Oliva 2003; Sargent 2005; Stonedahl and Rand 2014). Given the growth and availability of new data forms, we encourage modelers to follow a data-driven automated calibration process as the main validation tool and to use metaheuristics for automated model calibration (Chica et al. 2017; Miller 1998). We present the reasons why we recommend metaheuristics (Talbi 2009) here and the criteria and steps to design the most appropriate metaheuristic calibration method for each specific setting.

In addition, we demonstrate the application of the agent-based DSS framework to a real hedonic freemium app, *Creature Party*. We follow the general guidelines and recommendations to construct the DSS, and we show how to generate ABMs using the app data. The DSS forecasts the number of new daily premium adoptions (macro-level simulation) and the specific users who are going to convert in a given time horizon (micro-level forecast). Within this application, we examine different diffusion mechanisms including an agent-based version of the Bass model (Bass-ABM) (Bass 1969; Rand and Rust 2011) and the complex contagion model of Centola and Macy (2007). To our knowledge, this is the first work in marketing to use the complex contagion model. In the complex contagion model, adoption is contingent primarily on the absolute number of people you know that have adopted. We also use a metaheuristic automated calibration tool to tune the parameters of the models with respect to historical data. We then use the validated model to explore targeting and referral marketing policies (Haenlein and Libai 2013; Schmitt, Skiera, and Van den Bulte 2011) and evaluate their impact on premium market expansion through WOM and customer engagement.

FRAMEWORK AND STEPS TO BUILD A DSS

Our framework presents guidelines, design steps, and specific recommendations for creating an agent-based DSS for WOM market scenarios. The basic foundation for the DSS framework that we propose is an individual-level model that captures the social interaction dynamics of the customers, as embedded in a social network. Figure 1 shows the main four guidelines and three steps to consider when building the agent-based DSS. The four guidelines are (G1) to follow an iterative and participatory modeling process with marketers and stakeholders, (G2) to analyze and use all the available data to build the DSS, (G3) to employ data-driven calibration to increase users' confidence, and (G4) to minimize the complexity and the number of parameters in the model to increase ease of understanding. These guidelines are not necessarily steps to be followed in order but, rather, guiding principles to keep in mind during the construction of the DSS that will maximize the probability of its adoption by marketers. We describe the specific steps for creating the agent-based DSS next.

The first guideline G1 encourages the modeler to follow an implementable iterative model-building process (Leeflang et al. 2015) with a participatory element. This process must involve marketing managers and stakeholders, which is a key ingredient in facilitating better decisions, with less conflict and more success (Voinov and Bousquet 2010). G1 is also related to the trialability aspect of innovation adoption (Rogers 2003), because if marketers and stakeholders are able to try out the DSS, they are more likely to adopt it consistently in practice.

Given the growing importance of digital data for companies, guideline G2 discusses how all available data should be used when constructing the DSS. These data serve as an input for the creation of the DSS and may include information about the real social network of potential customers, seasonal information about customer use, empirical data on product or service adoption, and WOM volume or sentiment.

Guideline G3 states that data-driven model calibration is the cornerstone of the validation of the agent-based DSS. This validation is vital to increase the confidence of stakeholders in the DSS recommendations (Sargent 2005). The modeling process of the DSS is aimed at enhancing marketers' knowledge and understanding of the WOM dynamics by identifying the impact of the solutions and supporting marketing decisions for the WOM program. It is also important for the model to be understandable to encourage stakeholder use of the DSS. Therefore, guideline G4 declares that modelers should use the minimum number of parameters and mechanisms that enable satisfactory and valid results (Terano 2008) within the DSS. By creating minimal models, the researcher is more likely to facilitate the understanding of the models by stakeholders and, at the same time, is more likely to create a valid model. As Axelrod (1997) advised, "keep it simple, stupid."

Taking into account these four guidelines, we describe a framework for the creation of the DSS drawing on three main steps (summarized in Figure 1). These steps are to (S1) specify the marketing objective and the basic components of the DSS, (S2) create the model of the WOM dynamics through a social network, and (S3) use metaheuristics to perform a data-driven calibration of the model. In the following subsections, we explain each of the steps in turn.

S1: Specify the DSS Objective and Design the Basic Components of the ABM

To understand WOM dynamics, we need to specify a framework on which to build the agent-based DSS and to embed the adoption dynamics. We recommend the use of ABM (Epstein 2006; Macal and North 2005) because it can effectively model the aggregate consequences of WOM on the basis of local interactions among individual members of a population (Goldenberg, Libai, and Muller 2001; Libai, Muller, and Peres 2013). Web Appendix A goes into considerable detail about why ABM is appropriate as the basis of the DSS, but in brief, ABM provides the ability to model many heterogeneous individuals interacting across a complex social network in which the agents take their own actions that affect their decisions about how to spread WOM. During this construction step S1, five substeps should be taken to define (S1A) the objectives of the system, (S1B) the model architecture, (S1C) the updating

Figure 1

FOUR GUIDELINES AND THREE STEPS TO BUILD AN AGENT-BASED DSS FOR WOM PROGRAMS

FOUR MAIN GUIDELINES TO BUILD AGENT-BASED DSS

G1. Involve marketers and stakeholders in an iterative and participatory modeling process

G2. Analyze and use all the available marketing data for constructing the DSS

G3. Apply data-driven calibration and computational methods for validating the models

G4. Minimize number of parameters and mechanisms to enable the DSS comprehension

Step S1. Specify DSS Objective and Design Basic Components

A. **Establish a clear objective:** Focus the DSS construction to the WOM program setting and policies to explore (targeting, macro-level forecast, etc.)
B. **KPIs definition and initial adoptions:** Study needed agents' properties, initial adopters, and KPIs of interest (DSS output).
C. **Individuals updating rule:** Decide whether asynchronous or synchronous update for the individual's state.
D. **Granularity:** Choose granularity of the individuals according to the trade-off between needed details and computational efforts (e.g., mapping, temporal scale).
E. **Seasonal features:** Study the seasonality of the real customers to include this behavior within the model.

Step S2. WOM Dynamics in a Social Network

A. **Social network generation:** Employ artificial social networks when no information is available (e.g., scale free, ER). Otherwise, replicate real features of the service/app (e.g., degree distribution).
B. **Social influence:** Use weighted SNs for bidirectional diffusion when different social influences exist.
C. **Information diffusion model:** Analyze data to see number of friends when adopting. If necessary, add externalities.

Step S3. Data-Driven Model Calibration by Metaheuristics

A. **KPI selection for calibration:** Decide KPIs to calibrate the model in a single or multi-objective way by also including marketers' preferences.
B. **Deviation measure:** Select the deviation measure according to data set and KPIs. Choose single-point or behavior-pattern calculation.
C. **Holdout approach:** Divide data into training (to find the best model parameters) and test (to validate them).
D. **Search method selection:** If the modeler's knowledge is high, use local-based and single-solution metaheuristics (intensification). If it is low, use population-based and stochastic metaheuristics (high diversity). Select iterative metaheuristics when there are different ranges and types of parameters; select greedy metaheuristics when having hard constraints and dependencies between parameters.
E. **Automated sensitivity analysis:** Select high-diversification metaheuristics to find design relations (e.g., stochastic population-based) and analyze archives of solutions.

Notes: ER = Erdös-Renyi; SNs = social networks.

behavior of the agents, (S1D) the granularity of the agents, and (S1E) the seasonality of user behavior. Although designing a model architecture is important for any DSS, we focus here on the decisions points that should be considered by someone interested in developing a DSS for a WOM program.

S1A: Establish a clear objective. The first step when building the agent-based DSS is to keep in mind the main marketing objectives and the potential WOM programs that the stakeholders would like to explore. This is based on the design principle of building the model toward the question that the model is meant to answer in an incremental fashion (Wilensky and Rand 2015). In addition, the intended use of the models should be defined as precisely as possible (Leeflang et al. 2015), and every decision should be made with this objective in mind throughout all of the building steps of the DSS. In many cases, managers start with a default question, such as: Which users should I incentivize to maximize the adoption of my product?

S1B: Key performance indicators definition and initial adoptions. Agent-based DSSs are discrete-time simulations that end after several time steps. When a time step ends, the simulation collects the key performance indicators (KPIs) of interest at each time step and returns them as the output of the DSS (e.g., the time series of product-service adopters). Typically, in many WOM campaigns, the basic assumption is that some users have already adopted the product or the service (De Bruyn and Lilien 2008). This is usually modeled through a binary state (i.e., an agent is labeled as a nonadopter or adopter). Then, it is necessary to define how these initial adoptions are chosen. It could be through random choice or potentially influenced by some empirical data. A default option is to start by defining a single KPI (e.g., number of purchases, number of total adoptions) and then running the model to examine the adoption of the product in a baseline condition without any WOM program. Afterward, one can see how incentivizing users affects the adoption rate.

S1C: Individual updating rule. The individuals of an agent-based DSS can act asynchronously or synchronously within the simulation. In addition, the individual updates can be affected by competing contagions and other aggregate-level marketing efforts (Manchanda, Xie, and Youn 2008). Synchronous updates occur when no individuals reveal their new state until all have had a chance to change their state. This is usually set to occur during a system-level event that represents the time step of the model. Asynchronous updates occur when agents act and immediately reveal their state (Wilensky and Rand 2015). Synchronous updates are useful when interactions between individuals involved in the adoption process are not constant but, rather, some time lag exists between when they adopt and when the information about adoption can be passed on to others. This could be the case if there are no signs of conspicuous consumption, for instance, and the adoption decision only becomes obvious when users discuss their adoptions. Asynchronous update rules make the behavior of the simulation similar to traditional continuous diffusion models such as the original formulation of the Bass (1969) model; this is because an asynchronous update is closer to assuming that the number of adopters at time t depends not only on the number of adopters at time $t - 1$ but also on the instantaneous number of adopters at time t (Rand and Rust 2011). As a result, the default guideline is to use asynchronous updates for most agent-based DSSs.

S1D: Granularity. An important question is to decide the granularity and mapping of the individuals within the model. This requires specifying the temporal scale of the market context and the number of real customers represented by an agent within the model. At one extreme, this could be modeling every real customer with exactly one agent, and at the other extreme it could mean representing thousands of customers with one agent. This decision is made on the basis of the needed granularity of the DSS to make decisions. For instance, is it necessary to make decisions about individual behavior, or is segment- or population-level behavior a good-enough representation? In addition, the computational resources needed to run the model should be taken into consideration (Wilensky and Rand 2015). The default rule is to find a good trade-off between these two factors by always representing the fundamental level of information necessary to answer the WOM marketer's questions. For instance, if the question is about which agents to incentivize, a one-to-one mapping is often needed.

S1E: Seasonal features. The heterogeneity and flexibility of agent-based DSSs permit an easy inclusion of seasonal patterns of behavior. Because seasonality affects product adoptions (Peers, Fok, and Franses 2012, Guidolin and Guseo 2014), it is important to model seasonal effects when constructing an agent-based DSS. The modeler can define seasonality effects in production acquisition, service usage, or digital access at a given time step. In addition, the modeler can define a time-varying parameter that controls the probability of a particular event occurring at a particular time. At each time step in the simulation, the DSS can first consider whether a customer takes an action, such as accessing the service or using the product, by drawing a random number from a uniform distribution and comparing it to these seasonality parameters. The default suggestion for this step is to analyze the data and base the seasonality processes directly on this data.

S2: WOM Dynamics in a Social Network

Social networks play a fundamental role in the way information reaches consumers, channel members, and suppliers (Goldenberg et al. 2009, Van den Bulte and Joshi 2007). This is because the individual adoption decisions of customers normally depend on two factors: (1) external influence (by salespeople, advertising, promotions, and other marketing efforts) and (2) internal influence (affected by WOM or by observing conspicuous consumption of someone in their social network) (Libai et al. 2010). A social network is generally defined by a set of actors and the relationships (ties) among them. An individual's social network properties can affect the success of marketing actions, such as pricing or promotion strategies (Godes and Mayzlin 2009). Within this step S2 of our framework, we define three substeps: (S2A) generate the social network structure, (S2B) define the social influence between agents, and (S2C) model how information dynamics occur in the social network.

S2A: Social network generation. The social network defines the relationship between different consumers or users. Although it is a common approach to approximate a real social network with a synthetically generated preferential attachment network (Barabási and Albert 1999), many studies have provided evidence that most of the real-world social networks have distinct structural properties from synthetically generated social networks (Newman 2003; Stonedahl, Rand, and Wilensky 2010). Given the growth of online data and social media platforms, at least partial information and data about the social networks used by customers frequently exists about the customer base. Whenever possible, the DSS should be designed to take into account all the existing information about the real social network of the marketing context (G2). Therefore, the default rule of this step is that the actual social network should be used if it is known. Otherwise, the modeler is encouraged to use all the available information to guide the network generation by replicating the properties of the real-world social network. There are methods that can take partial social network information into account to generate more realistic social networks. For instance, generalized random networks allow for specific degree distribution (Milo et al. 2004; Viger and Latapy 2005) and/or the clustering coefficient of the social network (Newman 2009; Serrano and Boguná 2005). If no social network properties are known, the marketer can use one of the synthetic networks that does not take any information into account, but (s)he should consider whether it is worth creating a DSS for WOM if no social network information is available, as the social network is a critical component of understanding WOM programs.

S2B: Social influence. Many standard social network models assume that all consumers exert the same influence on each other, but in reality we know that is not the case. It is useful to consider the role of social influence between the WOM actors. One typical approach is to enrich the WOM process by modeling heterogeneous social influence by a weighted social network. To do this, the links of the social network are weighted on the basis of the social influence between customers. If all the links have the same weights, then the model collapses to the traditional model with homogeneous influence. The default rule is always to include these social influence differences, which can be inferred from data analysis after the implementation of marketing social-based activities by the company. For instance, it may be possible to observe when two users are talking with each other and use that information as a model of social influence. When this information is not available, modelers can first equally set these weights for all the customers and later run a sensitivity analysis to evaluate the effect of changing the weights.

S2C: Information diffusion model. Apart from designing the network structure, it is important to model the WOM dynamics that occur through the medium of the network. The probability that an individual chooses between one product alternative or another is increased according to the relative number of others choosing the same alternative and the influence of those others on the focal individual (Goldenberg, Libai, and Muller 2001; Trusov, Rand, and Joshi 2013; Van den Bulte and Joshi 2007; Watts and Dodds 2007). Threshold (Granovetter 1978; Watts and Dodds 2007) and cascade (Goldenberg, Libai, and Muller 2001, 2010) models are common individual-level diffusion mechanisms used in marketing to model an individual's decision. Stochastic cascade models hypothesize that when each agent adopts a product, it a small probability of influencing any of its social neighbors to adopt the product. In the threshold model, each agent observes the fraction of neighbors that have adopted and then adopts if this fraction exceeds a certain threshold. In both cases, it is also possible to add an external influence parameter, which can encourage adoption of the product independent of social influence. Unless data analysis suggests a more appropriate approach, it makes sense to start with something similar to an agent-based version of the Bass model (Bass-ABM; Rand and Rust 2011), which assumes independence of the internal (customer-to-customer interactions through WOM) and external (Libai et al. 2010) effects. This model is a form of a cascade model and is already well accepted in the marketing literature.

S3: Data-Driven Model Calibration by Metaheuristics

Automated calibration is a data-rich and computationally intensive process that uses an error measure to compare real-world data with model data and then tunes the parameters of the model to identify a set of parameters that best match the data (Oliva 2003; Sargent 2005; Stonedahl and Rand 2014). Automated calibration attempts to discover the best parameters of the model that fit the model to the data. This evaluation of the model fitting is done by running the computational model and comparing its outputs with the data. This means that automated calibration requires an error measure and an optimization method to modify the parameters in a systematic way to minimize the error measure. Our framework presents the building steps to calibrate the agent-based DSS that was created in the previous two steps, using metaheuristic methods (Talbi 2009).

Metaheuristics are a family of approximate nonlinear optimization techniques that provide acceptable solutions in a reasonable time even when problems are difficult and complex (Talbi 2009). When calibrating a complex system such as an agent-based DSS, metaheuristics are preferred over gradient-based methods or mathematical programming for two main reasons: (1) we can make only minimal assumptions about the nonanalytical simulation model (i.e., the relationship between all parameters and all outputs in the ABM framework is unknown) and (2) the objective function (i.e., a function that formulates the goal to achieve when optimizing) is time consuming and needs to be run many times to accurately compare the simulated model with real marketing data. This makes it difficult to create a closed-form solution and computationally too expensive to conduct a full search of the parameter space.

One well-known type of metaheuristic is the genetic algorithm (GA; Goldberg and Holland 1988), which are powerful search methods that have already been applied to marketing problems (Luo 2011; Venkatesan, Krishnan, and Kumar 2004). Metaheuristics can be classified according to various characteristics (Talbi 2009): nature-inspired versus not nature-inspired, deterministic versus stochastic, population-based versus single-solution based search, and iterative versus greedy. Yet the relevant issue when building an agent-based DSS is to find the most suitable metaheuristic for the particular WOM setting, which can be difficult because there are two contradictory goals that must be taken into account when choosing and designing the metaheuristic. On the one hand, there is a goal of maximizing the exploration of the parameter space (diversification) and, on the other hand, there is a goal of exploiting the best solutions discovered so far (intensification). In the next five steps, we discuss these and other criteria to select the most appropriate metaheuristic calibration method.

S3A: KPI selection for calibration. The first step is to identify one or more KPIs to compare the output of the DSS with real data. These KPIs can be the number of adoptions, WOM volume and/or sentiment, or the sales of a brand. Depending on the number of KPIs in conflict, modelers must choose between single-objective (i.e., only one KPI) and multiobjective (i.e., when multiple KPIs that are potentially in conflict need to be optimized simultaneously) metaheuristics (Chica et al. 2010; Talbi 2009). An additional option when more than one KPI exists is to include stakeholder knowledge to weight and value the KPIs within the calibration process by using preference relations or units of importance between the defined KPIs (Chica et al. 2011). This enables the creation of a single KPI by taking a weighted sum of multiple KPIs. However, in the spirit of simplicity, the best place to start is to calibrate the DSS with a single KPI and then move to a more advanced approach if needed.

S3B: Deviation measure. The metaheuristic deviation measure evaluates the quality of a set of parameters by comparing the model results for that set of parameters with historical data. Modelers can use different error or deviation measures, and this choice can significantly affect the calibration performance (Stonedahl and Rand 2014). A traditional approach is to use single point-based measures (e.g., root mean square error [RMSE], Euclidean distance, mean absolute percentage error). The selection of the specific measure depends on the problem and data characteristics (e.g., trends in historical data, number of KPI data sets). The default rule of this step is to use RMSE or Euclidean distance to calibrate a simple set of KPIs for a match of a typical series of historical data points. If the goal is to favor general trends over specific data matches, then we suggest using mean absolute percentage error because it decreases the effect of large, isolated data point errors within the model calibration (Chai and Draxler 2014).

S3C: Holdout approach. Independent of the choice of metaheuristic, modelers are encouraged to use a holdout approach when calibrating the model. As explained in Stonedahl and Rand (2014), the modeler divides the whole historical data set of KPIs into two data sets: (1) training and (2) test, with their corresponding environmental variables. The environmental variables are those that will not change from run to run of the models, because they do not belong to the set of parameters to be calibrated. The calibration of the model is then accomplished by identifying, using the training data set, the model parameters that minimize the chosen deviation measure. Later, the model's validity is examined by using the same set of parameters but for the test data set. More advanced approaches such as k-fold cross-validation can be also used (Witten and Frank 2005). By default, the rule is always to use a basic holdout approach to ensure that the calibrated model is generalizable to data that were not used to train the model.

S3D: Search method selection. The knowledge of the modeler about possible good parameters' values and the features of the parameter space can be used to decide which metaheuristic to use. The modeler's knowledge should be used to help constrain the search space and create a metaheuristic that can explore the restricted parameter space in-depth, a process called intensification (Talbi 2009). When knowledge about the best configuration parameters is scarce, population-based metaheuristics allow for a wider search for the best set of parameters' values. In addition, iterative metaheuristics (e.g., GAs that start with a complete solution or population and transform it at each iteration using search operators) are more flexible if different parameter ranges and types exist when calibrating (e.g., integer, real, binary). This is because the solution is built at the start and the process does not need to be customized for each type of parameter (Talbi 2009). Greedy and constructive metaheuristics (e.g., greedy randomized adaptive search procedure; Feo and Resende 1995) or iterated local search (Lourenço, Martin, and Thomas 2003) are suggested when there is a high number of dependencies between parameters and hard optimization problem constraints (Chica et al. 2010). This is because a greedy metaheuristic starts from an empty solution and constructively assigns at each step a parameter value for the calibration problem until a complete solution is obtained. As a default recommendation, when modeler knowledge is limited, it makes sense to use as a default one of the population-based metaheuristics, such as the aforementioned GAs (Holland 1975), ant colony optimization (Dorigo, Maniezzo, and Colorni 1996), or particle swarm optimization (Kennedy 2010). In general, we recommend iterative metaheuristics given the unusual presence of hard parameter constraints in this kind of DSS.

S3E: Automated sensitivity analysis. Sensitivity analysis reveals those parameters to which the model behavior is highly sensitive (Saltelli et al. 2008). Together with calibration, it is a key ingredient for model testing and verification (Oliva 2003). Miller (1998) initially highlighted GAs as an appropriate tool for sensitivity analysis because of their capability to explore a wider range of parameter settings with a higher resolution and to also consider potentially complex or nonlinear interactions between them. Specifically, population-based metaheuristics offer not only a final calibration solution but an *archive* of evaluated model calibration solutions for the model. Automated sensitivity analysis can be performed on these solutions to discover hidden properties related to the model design (Chica et al. 2017). The default rule is to use metaheuristics that emphasize diversity, often in the form of stochastic population-based metaheuristics, to assist in the sensitivity analysis of the model.

APPLICATION TO A FREEMIUM BUSINESS MODEL

Freemium business models offer a service or a product free of charge, but a premium is charged for advanced features, functionality, or related products and services (Anderson 2009). Over the past decade, freemium has become an important business model for digital-based products and services including software, games, and websites (Teece 2010). The 2015 freemium app monetization report by App Annie (an app analytics firm) and International Data Corporation states that freemium app revenues grew by over 70%, whereas paid app revenues declined by 19% from 2013. Word of mouth often plays a large role in freemium app adoption because many of these apps are game-related or have a built-in social network component (Bapna and Umyarov 2015). In this section, we examine the application of our agent-based DSS framework for WOM to a real-world, hedonic freemium app.

The importance of WOM in freemium models is not new. Cheng and Tang (2010) show that the optimal price of commercial software increases with the network intensity of the software, which can be enhanced by free trials. Oestreicher-Singer and Zalmanson (2013) study the premium services of Last.fm, an online music website, and find that the willingness to pay for premium services is strongly associated with the user's level of community participation (i.e., social behavior). Bapna and Umyarov (2015) show that influence from an adopting friend caused more than a 50% increase in the probability of adopting premium services. They also found that users with a smaller number of friends experienced stronger relative increase in the adoption likelihood due to influence from their peers as compared with users with a larger number of friends. Other studies presented relevant conclusions on the importance of WOM programs for managing freemium business models. For instance, Lee, Kumar, and Gupta (2013) examine consumer referral behavior and show that referral invites recruit consumers who later convert to premium consumers. Therefore, our article makes a vital contribution to this literature by illustrating how a general DSS for WOM can help managers gain insight into many of these questions for their particular data sets that have previously been explored by marketers.

App Data and the Main Marketing Questions

In this subsection, we explore the specific application of our framework to a freemium app called *Creature Party*. It is a multiplatform social online game for kids in which users interact online with other users. Basic users can freely access and play the game and interact with other users (both premium and basic), but premium users receive additional benefits such as weekly in-game currency allowances (called "diamonds" and "gems"), the ability to adopt virtual pets, access to all the avatars, and premium-only adventures. CrazyGames, the developer of *Creature Party*, was interested in making better marketing decisions for expanding its premium market. It wanted to determine whether WOM played a large role in the adoption of premium services by freemium users. This knowledge could then potentially be used for several different purposes, including designing a reward-based marketing campaign to maximize the spread of positive WOM about premium membership, which might, in turn, increase overall adoption rates and revenue.

In general, referral rewards are designed to motivate consumers to spread positive WOM about products and services and to turn customers into an element of the sales force (Biyalogorsky, Gerstner, and Libai 2001). Lee, Kumar, and Gupta (2013) examine rewarding through referrals in the context of a freemium software service. They explore the question of the right referral bonus incentive to offer to free users. Contrary to the belief that more is better, they found, by maximizing the average consumer referral rate and changing the referral incentives, that there exists an optimal incentive point for referrals that is not simply as much as possible. Therefore, the decisions related to implementing an optimal referral program are not straightforward. In our case, and to help *Creature Party* managers with their marketing goals, we constructed a DSS using the guidelines and steps described previously to (1) replicate and forecast the conversion rate from freemium to premium members and (2) evaluate incentivization-based marketing campaigns and measure the additional customer acquisition created by amplified WOM.

The company provided us with daily conversion and subscription data of its users in 2012. These data included 1.4 million game users and the company's social network, and they are valuable because they give us the opportunity to properly design the DSS and to validate the models with respect to historical trends. For the quantitative models' validation, we restricted the whole data of 1.4 million users to only those users who were active during the months of the study (i.e., from 40,000–50,000 basic and premium users).

In the following sections, we explain how we built a DSS for *Creature Party* utilizing our framework to help answer the main questions of interest to managers. Note that in addition to the detail provided in this section, we present the documentation, verification, and validation details of the DSS for *Creature Party* in Web Appendix B. Furthermore, the agent-based DSS source code can be freely accessed at https://bitbucket.org/mchserrano/socialdynamicsfreemiumapps. The graphical user interface of the DSS with its main functionalities is also available at a major ABM repository (https://www.openabm.org/model/5191). These tools provide researchers with the ability to see the actual code as well as run the model to explore their own scenarios of interest. We provide some screenshots of the DSS in Web Appendix E.

S1: Definition of the DSS Objective and Designing Basic Components

During this step, we paid close attention to guideline G2 by using the data provided by the company and to guideline G4 by simplifying the DSS design.

S1A: Establish a clear objective. The main goal in building the *Creature Party* DSS is to forecast the adoption of the premium services by users of the app. Because the incentive programs of managers' interest often focus on individual-level policies (e.g., targeting specific users), we should consider a DSS based on a very granular model. The DSS should be able to forecast both the total number of new daily premium adoptions of the app (macro-level forecast) and whether a specific user is likely to become a premium user in a given time horizon (micro-level forecast).

S1B: KPIs definition and initial adoptions. There is one central KPI: the number of premium adoptions of the basic users of the app. This KPI must be forecasted daily or weekly as a conversion rate for the total number of users of the app. As noted by guideline G1, we followed an iterative process with stakeholder participation to find the correct KPI necessary to answer the WOM marketer's questions. During this process, we discovered that there was an interest in forecasting not only the aggregate conversion rate but also particular users who were likely to convert in the near future. To begin to calibrate our model, we defined an initial adoption rate (α) that reflects the number of initial premium adopters of the app. The simulation of the DSS will then begin with a ratio $\alpha = .0406$, which corresponds to the 4.06% of premium users of the real app at the starting simulation date.

S1C: Individual updating rule. We used an asynchronous update rule in the model to better replicate the WOM dynamics of the app, following the default rule of this framework step. Within the same time step (day), the users can make decisions and update their states asynchronously. This more closely mirrors the real app, in which a user can influence his or her online neighbors as soon as (s)he becomes a premium member.

S1D: Granularity. After a preliminary exploration of granularity, we defined one individual of the ABM framework to correspond to two real app users active during the months of the forecast. Therefore, our simulation consists of 20,000 individuals (with 812 of them being initial premium adopters) that map to 40,000 active real users of the app during the months of the study. We made this assumption to decrease the computational cost of the model without sacrificing too much resolution.

S1E: Seasonal features. We found a strong seasonality in the app dynamics after analyzing the time series of premium conversions in the historical data (guideline G2). The app seasonality shows that people tend to be more active on weekends than on weekdays. The observation seems obvious because the app is hedonic, aimed at kids, and typically played by users when they have more free time. Please note that the seasonality of the *Creature Party* DSS is crucial because app users cannot adopt premium content and/or talk with their direct friends on the app if they do not access the app on a particular day. We define seasonality within the model by grouping the probability of using the app into two seasonality parameters: the probability of using the app on a weekday (γ_0) and the probability of using the app on a weekend (γ_1). This simplification is in line with guideline G4 to keep the model as simple as possible without losing accuracy (Terano 2008; Wilensky and Rand 2015).

S2: Replicating the WOM Diffusion Process Through a Social Network

The analysis of the freemium app data shows there is a clear difference in friendship patterns between basic and premium users, observed in the degree density graphs (guideline G2). The fact that premium members show such a different pattern of social behavior lends credence to the related literature on freemium business models showing that users' social activity is related to their conversion (Bapna and Umyarov 2015; Lee, Kumar, and Gupta 2013). In further support of the importance of social effects is the observation that having more premium members as friends increases the likelihood that one will adopt. From analyzing the app data, we find that (1) less than 1% of users without a premium friend subscribe, (2) the chance of conversion triples (from .8% to 2.3%) if a user goes from having zero premium friends to just one, and (3) the incremental impact on adoption of having premium friends is higher the fewer friends one has. Using these insights and the S2 step of the framework, we constructed a WOM diffusion model for the DSS, which we explore next.

S2A: Social network generation. By analyzing the real social network, we obtained a distribution degree that represents the social relations of the app users. One interesting fact about this distribution degree is that the app limits a user's number of friends to 100; as a result, the degree distribution is heavily bimodal. As a result, there is a group of users with very few friends and another group clustered around the upper limit of 100 friends, which is enforced by the *Creature Party* app; there are fewer users in between these extremes. Using this information, we employed the generalized random networks algorithm of Viger and Latapy (2005) to generate a social network of individuals with similar distribution degree to the real social network of users in the real app. Although we could have used the model's actual network information, by using a synthetic network, we are able to scale the size of the network to the size of the population being simulated. Web Appendix B provides more information about the degree distribution of the generated social network.

S2B: Social influence. Following S2, for each link between two individuals i and j of the social network, there are two different weights, $\tau_{i,j}$ and $\tau_{j,i}$. These two weights generate a bidirectional influence, where the social influence of i when (s)he talks with j can be different than when j talks to i. When $\tau_{i,j} > k$ (or $\tau_{i,j} < k$) the influence of i on j is increased (or decreased) according to the weight. Initially for the model, and because we do not have specific information about the social influence between the users, we set the influence τ_{ij} to 1 for all the links. We did use communication records from the real game to model social influence, but in this case, it did not appear to improve the results.

S2C: Information diffusion model. Our data analysis and the existing work on freemium apps (Bapna and Umyarov 2015) highlights the marginal impact of additional premium friends when a user already has many premium friends. This suggests the use of a complex contagion model (Centola and Macy 2007) as the diffusion model for the agent-based DSS because this model mimics a similar pattern of adoption. The model is called a complex contagion model because successful transmission depends on interaction with multiple carriers. The complex contagion model is similar to the threshold model, but the main difference is that in the threshold model, the threshold ϕ describes the exposure amount necessary to convert relative to the number of friends an individual has, whereas in the complex contagion model the threshold ϕ is the absolute (not relative) number of exposures necessary to convert. We also implemented an agent-based Bass model (Bass-ABM; Rand and Rust 2011). The Bass-ABM translates the hazard rates of the Bass model to probabilities for a single consumer and embeds the consumer in a social network in which the decision to adopt also depends on the fraction of neighbors who have converted. Finally, we considered an extended complex contagion model. This new diffusion model adds an external influence probability, which, similar to the Bass-ABM, explicitly includes an influence outside the network. Web Appendix B provides more details about these three diffusion models: Bass-ABM, traditional complex contagion model, and extended complex contagion model with an external influence probability.

S3: Designing the Data-Driven Calibration Method

We describe how to design the automated metaheuristic calibration for the DSS of the *Creature Party* (step S3 of the framework). This step is used to validate the model (guideline G3) (Sargent 2005). The parameters of the agent-based DSS must be calibrated to adequately fit the historical data of premium adoptions. The size of the parameters set P* to be calibrated is either three or four, depending on the diffusion model: two probability values for seasonality ($\gamma_0, \gamma_1 \in [0, 1]$), and the external and internal influence coefficients ($\hat{p}, \hat{q} \in [0, 1]$) in the Bass-like model, or the minimum threshold of premium friends for the complex contagion ($\phi \in [0, m]$, where m is the maximum number of friends). Table 1 shows the list of parameters to be calibrated for all the considered diffusion models. In the following subsections, we discuss the decisions necessary to design the most appropriate metaheuristic calibration method (step S3).

S3A: KPI selection for calibration. We use a single KPI of interest, the aggregated number of new premiums (adoptions) per day in the app. We could instead use individual conversions as the KPI, but it turns out that calibrating using the aggregate patterns also creates an accurate tool for micro-level targeting, as we explore in the validation section of the experimentation. The quality of the calibration solutions generated by the metaheuristic is evaluated by calculating the difference between the historical new premium users and the model output in a daily setting. This model output is obtained by running a Monte Carlo model simulation of 15 runs for each parameters setting, which was sufficient to account for the model's variability.

S3B: Deviation measure. We follow a single-point approach to calculate the deviation between the real and simulated new premium values. We use the Euclidean distance as the objective function of the metaheuristic after running a preliminary experiment to check the goodness-of-fit to the seasonal trend using this measure.

S3C: Holdout approach. A holdout approach over the whole data set is carried out as recommended by the default rule of the framework. From a data set of three months of daily conversions, two data sets are generated: 60 days of historical premium daily conversions as a training set and 31 days for the test data set. Therefore, the metaheuristic calibrates the models'

Table 1
SET OF MODEL PARAMETERS P* TO BE CALIBRATED BY THE GA

		Seasonality Parameters		*Bass-ABM Coefficients*		*Complex Contagion Thresholds*
Diffusion Models of the DSS	*Number of Parameters*	$\gamma_0 \in [0, 1]$	$\gamma_1 \in [0, 1]$	$\hat{p} \in [0, 1]$	$\hat{q} \in [0, 1]$	$\phi \in [0, m]$
Bass-ABM	4	✓	✓	✓	✓	—
Complex contagion	3	✓	✓	—	—	✓
Complex contagion with external influence	4	✓	✓	—	✓	✓

Notes: m = the maximum number of friends a user can have in the app.

parameters using the training data to fit historical data, and then this calibrated model is used to predict premium conversions on the test data set to validate the model generalization.

S3D: Search method selection. We did not have any particular model knowledge for the set of parameters we were exploring. In addition, there is no hard constraint on the relationship between the parameters and each of the three diffusion models has a set of parameters of different types. For instance, the threshold ϕ of the complex contagion model is an integer parameter while the seasonality parameters γ_0 and γ_1 are real. Therefore, and following the default rule of this framework step, we used a high-diversity, iterative, and population-based metaheuristic (e.g., a GA; Goldberg and Holland 1988; Holland 1975).

S3E: Automated sensitivity analysis. We are also interested in understanding the ranges of each parameter that seem to provide the highest validity and the actual forecasts for the freemium app. For instance, the values for the threshold parameter of the complex contagion model can provide notable insights about the how many freemium friends a focal friend needs to convert. The use of a population-based metaheuristic has the advantage of performing an automatic sensitivity analysis because it generates a large set of solutions to the problem (Chica et al. 2017). This enables the exploration of the already evaluated set of parameters and allows for the examination of nonlinear interactions between the parameters.

The recommended metaheuristic characteristics given by steps S3D and S3E suggest that a GA can work well when calibrating the DSS for the *Creature Party* app. In a nutshell, a GA evolves a population of solutions (chromosomes), each of which represents a model's parameter set. These solutions are evolved until the best possible design (parameters of the model) for a given modeling goal (i.e., correctly matching historical data) is achieved. We detail the remaining GA components and the GA parameters in Web Appendix C. Finally, it is important to note that the designed GA calibration method is independently run to calibrate each of the three diffusion models: Bass-ABM, original complex contagion, and complex contagion with external influence coefficient. We can then use the accuracy of the macro-level forecasting of the premium adoption of the app to adjudicate which model does the best job of describing the real WOM behavior.

VALIDITY AND PERFORMANCE OF THE MODELS FOR THE FREEMIUM APP

Modelers and managers need to know how well the DSS models replicate the real-world data after the calibration. The agent-based DSS for the app case provides two different forecast levels: macro- and micro-level forecasts. We first analyze the agent-based DSS to examine its ability to replicate macro-level historical data (i.e., the number of daily new adoptions provided by the calibrated models). Then, we examine the models' performance when forecasting the most likely users to adopt premium in the near future (micro-level forecast).

Model Calibration Results for Macro-Level Forecast

The GA run for calibrating the DSS for *Creature Party* ends when 20,000 different parameter settings (solutions) have been tested. This is the stopping criteria of the GA and is sufficient for finding a good-quality solution. We also run the overall GA calibration method 15 times per model because the GA itself is nondeterministic. At the end of all the runs, the metaheuristic calibration method returns the mean and standard deviation values of the Euclidean distance between the real and simulated data.

The Bass-ABM presents the best-fitting within the set of three diffusion models of the DSS. The GA calibration of the Bass-ABM, ends with a Euclidean mean value of 358.72 and a standard deviation of 2.01. By following the holdout approach, we examine the calibrated Bass-ABM model on the test data set by obtaining a Euclidean mean value of 339.01 and 9.09 for the standard deviation. The DSS with a complex contagion model obtains mean values of 447.16 and 420.54 for the training and test data sets, respectively. The standard deviation is again higher for the test data set: 20.37 for test while obtaining 4.75 for training data set. Finally, the complex contagion variant with the external influence coefficient obtains similar results to the original complex contagion model. Its Euclidean mean values are 441.53 (training) and 416.76 (test) while its standard deviation values are 5.46 (training) and 19.49 (test). The low standard deviation values indicate that the 15 GA runs were sufficient to obtain quality results.

By using the population-based results, we can directly perform a sensitivity analysis and observe the parameters' range to better understand the strength of the social influence when adopting premium services. Web Appendix D presents boxplots of the parameters' distributions for the three diffusion models. From this sensitivity analysis, we can see that the threshold parameter of both complex contagion models (ϕ) is mostly set to three friends by the calibration process. In the case of the complex contagion model with external influence probability, the range of good values for the external influence coefficient ($\hat{p}$) is much wider than in the Bass-ABM. This means that the influence of the external influence coefficient for the complex contagion model is less critical than for the Bass-ABM and also reinforces the aforementioned conclusion of low differences between both complex contagion models. We also observe that the GA-based calibration method focuses on very specific seasonality and internal influence coefficient values to provide a good model fit for the complex contagion model. However, the Bass-ABM results are more robust with respect to a wider range of values for the internal influence coefficient and seasonality parameters ($\hat{q}$, γ_0, and γ_1).

Web Appendix D provides additional analysis of the temporal evolution of premium adoptions. It shows in more detail that the Bass-ABM model's predictions better match the total number of adopters at the end of the simulation period than the other models. Thus, the Bass-ABM outperforms both of the complex contagion models in the training and test data sets. As a result, the Bass-ABM is the diffusion model of preference for macro-level forecasting in the *Creature Party* case, though performance differences between the three models are not large. Finally, these premium adoption results for the *Creature Party* setting suggest that the introduction of an external influence coefficient $\hat{p}$ does not improve the performance results of the traditional complex contagion model, which indicates that the internal diffusion forces for that model capture most of the behavioral patterns.

Micro-Level Forecast: Models Comparison

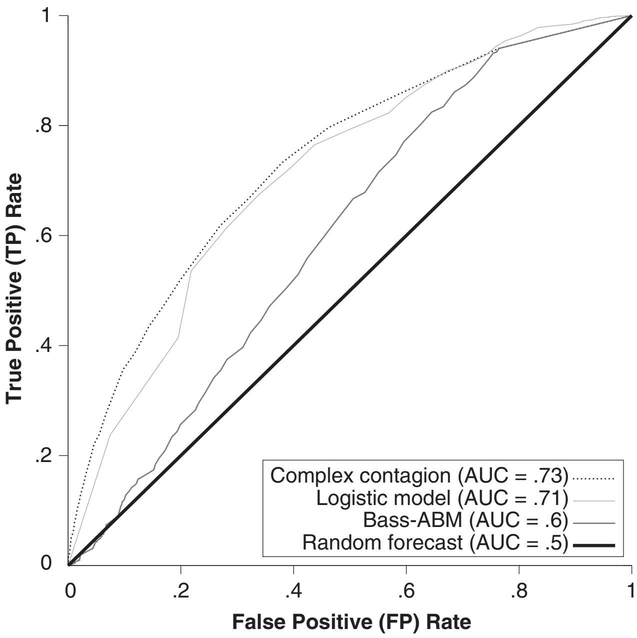
One of the main interests of the managers was to better understand how to target and incentivize basic users. The objective was to identify those users who could easily be convinced to become premium users. To do so, we created a model that could make micro-level predictions. We applied the Bass-ABM and traditional complex contagion models to identify basic users who are more likely to adopt the premium membership after one month. In this comparison, we did not include the complex contagion with external influence coefficient, because there were no significant performance changes with respect to the traditional complex contagion model. However, we did build a simple logistic model with a ridge estimator (Le Cessie and Van Houwelingen 1992) using the Weka toolkit implementation (Witten and Frank 2005) and a random classifier as a baseline to help understand the performance of the agent-based DSS. The logistic model was designed to use one independent variable (number of premium friends of a user) and one dependent variable (if the user is adopting or not after one month). Again, a holdout approach is followed by using 75% of the data for training and 25% for testing.

To validate the micro-level forecast performance of the agent-based DSS, we defined a set of app users in the initial time period of one month and label which of those users will eventually adopt. This data set has 10,798 app users (basic users at the beginning of the month). We label those users who will adopt the premium membership after one month (718 from the total number of 10,798 users). The set of 718 premium adopters is called the true positives (TPs). The true negatives are those who never convert in this month (the remaining 10,080 app users). We also calculated the false negative and false positive (FP) rates of the models. An FP occurs when a user is incorrectly predicted as a premium adopter when (s)he actually never converts to premium. A false negative is the opposite case, in which the user is incorrectly predicted to be a nonadopter when (s)he does adopt. A good way to summarize this analysis is by using the receiver operating characteristic (ROC) curve and the area under the curve (AUC) (Witten and Frank 2005).

The ROC curve provides a way to represent the trade-off between FPs and TPs for different values of the rejection threshold by showing the relation between the sensitivity and specificity of the forecast. The AUC summarizes the area under the ROC in the entire range $[0, 1]$ of the FP rate. The higher the AUC value, the lower the FP rate for a given TP rate (i.e., the model performs better because it identifies TPs more frequently with fewer FPs). Figure 2 shows the ROC curves for Bass-ABM, complex, logistic, and random models. In addition, its legend shows the AUC values. Complex contagion achieves a higher forecast performance than the Bass-ABM. In fact, the complex contagion model achieves better results than the logistic model. The AUC values are also in line with the curves: complex contagion has the highest AUC value (.73), which surpasses both the Bass-ABM (.60) and the logistic model (.71).

Figure 2

ROC ANALYSIS OF THE MODELS FOR THE MICRO-LEVEL FORECAST



TARGETING AND INCENTIVE POLICIES USING THE DSS

Using the DSS to Increase Social Influence Through Rewards

Before examining specific targeting policies, we use the DSS to shed light on how reward campaigns can increase the amplified WOM in the *Creature Party* setting. This enables us to better understand how conversions lead to other conversions. This step constitutes a test of the agent-based DSS behavior and provides first insights for the app managers by exploring different social influences within a set of app users. To accomplish this goal, we use the calibrated Bass-ABM (i.e., the diffusion model that fits the macro-level data the best).

One area of interest to managers involves understanding how rewarding a user who converts to premium could affect the conversion of other users through positive WOM. We simulate the provision of rewards to a target set of users postconversion and then use the agent-based DSS to explore the effect of social influence of the rewarded users on their friends. The idea being that if managers reward users when adopting, the rewarded users are more likely to talk positively about the benefits of a premium membership with their friends, increasing their influence over the baseline rate. The revenue of these policies is measured by counting the successful referrals after a given period. It is also important to note that, to examine different types of rewards (i.e., app bonus, extra software features, or gifts), we explore different amounts of social influence by varying the social influence value per dollar value of the reward (i.e., we explore a variety of results on the basis of how much a reward causes a user to talk positively about conversion).

In addition, we made use of a weighted social influence (per dollar of investment) between the users to adjust the influence with their friends (parameter τ_{ij} defined in step S2B). These weights are included in the individual adoption probability rule of the Bass-ABM. This is done, for each basic agent, by multiplying the fraction of her or his premium contacts by the social influence weight of each contact. By using this method, it is possible to amplify the influence of each converted premium user with their friends during the simulation time. At the beginning of the simulation, all the social weights τ_{ij} are set to 1, which means that, initially, all users affect each other uniformly. However, when a basic user adopts premium content and is rewarded, her or his social influence with other users is increased (value greater than 1).

Creature Party, like most companies, does not empirically know the real social influence between two friends when rewarding one of them because it had never implemented a similar policy in the past. Thus, to shed light on the potential range of effects, we ran a sensitivity analysis to understand the market expansion results when considering different values for the social influence weights. Marketers can look at these results, which include both pessimistic and optimistic scenarios, and make their decisions accordingly. If the app company implements this kind of reward policy in the near future, it will be possible to analyze the app conversion data and estimate a value for the social influence weight per dollar spent on reward.

The plot of Figure 3 shows the number of additional premium users during 31 days in the test period by considering different increases in the social influence (x-axis of the plot) with respect to offering no rewards (i.e., point [0, 0]). The main finding after analyzing this sensitivity analysis and scenario is that increasing social influence between users by rewarding users who adopt premium content has a positive nonlinear impact on increasing the number of premium adopters of the app (i.e., referrals can be quite successful). We can also see how the lift in additional premium members does not demonstrate linear behavior when the social influence is increased by rewarding users at the time of adoption.

Targeting the Most Likely Users to Adopt Premium

Once the agent-based DSS is validated and the effect of the initial postconversion reward polices has been explored, it is time to explore more complex WOM strategies and scenarios of interest. In this subsection, we try to answer app managers' main marketing questions: How can we target and incentivize specific basic users to expand the premium market of the app? Furthermore, how many additional premium adopters can the amplified WOM of the incentivization policies encourage to convert?

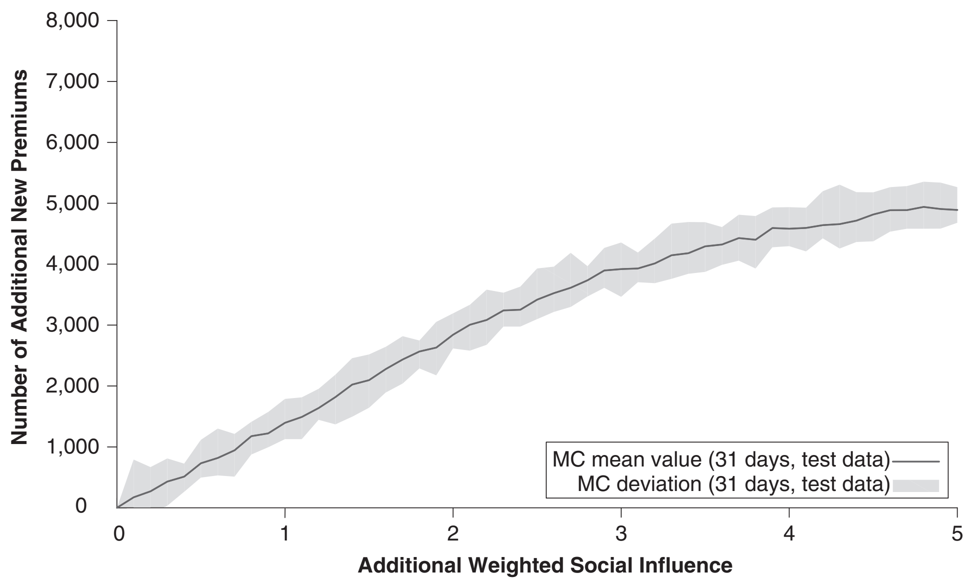
For instance, a policy that targets basic users (vs. rewarding premium conversions) could be a profitable strategy. Instead of rewarding users after they convert, as we did in the previous subsection, we evaluate a marketing strategy to incentivize different groups of nonconverted users (basic users) to stimulate them to adopt premium and to increase the amplified WOM of the app. We follow a similar targeting approach to the one of Haenlein and Libai (2013) by studying the strategy to reward the most likely users to adopt—that is, those expected to generate more value (revenue leaders).

We jointly use the micro and macro levels of the agent-based DSS to select the best basic users to target by running simulations to evaluate the implications of different targeting policies for market expansion. To identify which users to target, we examine how likely users are to convert to a premium user account on their own using the micro-level forecasting method of the ABM; then, we examine what the effect of rewarding those users is on the overall premium conversion rate using the macro-level forecasting method. We use the complex contagion model for the micro-level forecast to help determine which users to seed, and we use the Bass-ABM for the macro-level forecast to investigate the long-term impact of that seeding policy. We use these two models because they outperformed the rest of the diffusion models during the validation phase of the DSS. The DSS generates a group of the 2,000 most likely basic users to adopt. This user selection is done using the micro-level DSS forecast.

After targeting the selected group of basic users with incentives, we ran different simulations to forecast the additional premium conversions the company can obtain with respect to the case of not applying any incentive policy. A sensitivity

Figure 3

AN AGENT-BASED DSS SIMULATION OUTPUT OF ADDITIONAL PREMIUM ADOPTIONS WHEN REWARDING ADOPTERS VERSUS THE SOCIAL INFLUENCE PER DOLLAR



Notes: MC = Monte Carlo.

analysis is run for the parameter that affects the weighted social influence (τ_{ij}). The exact effect of providing an incentive on the conversion rate is not known. To solve this problem, we ran an additional sensitivity analysis on the lift in the external influence coefficient $\hat{p}$ when rewarding a target user.

Web Appendix D contains three heat maps of the results after applying the targeting and incentive policies and running the DSS simulations. Each map is associated with a different way of selecting the group of target users. The different groups are selected on the basis of the complex contagion model parameter ϕ, which discriminates whether a user is forecast to adopt or not depending on the minimum number of adopters within her or his friends. Concretely, we examine 1, 3, and 6 for that threshold ϕ (minimum premium friends). Under some conditions (high social influence and high increase of the external influence coefficient), the incentivization policy forecasts that the company would obtain more than 1,000 additional premium conversions in one month.

For instance, when considering a scenario of a .5 increase in the external influence coefficient ($\hat{p}$) and 1.8 for the social influence weight (τ_{ij}), we observe a considerable market expansion. In this case, the runs of the macro-level version of the DSS (created with the Bass-ABM diffusion model) result in an average of 686.8 additional premium conversions when targeting basic users with a minimum of one premium friend, in 713.33 when targeting basic users with a minimum of three premium friends, and 819.86 when targeting basic users with a minimum of six premium friends (see Web Appendix D). In general, the best expansion results are those when creating a target group of users with six or more premium friends, but the three scenarios present similar results.

Managerial Implications of the DSS

We created this agent-based DSS directly in response to questions that the app managers had about the best way to target and incentivize their basic users. Specifically, we addressed the following questions:

- *What is the importance of WOM and its role when free app users consider adopting premium content?* By analyzing the model's performance and sensitivity analysis on the diffusion parameters, the DSS showed that WOM influence has a significant effect on premium market expansion. For instance, we found, as a result of the complex contagion model, that the average user is likely to convert when three of his or her friends are premium.
- *What is the effect of rewarding conversions and how does this affect market expansion?* The exploration of different amounts of social influence, generated by rewards for conversion, showed that there is also a great potential to amplify market expansion when implementing such a strategy. However, we discovered that this fact exhibits a nonlinear relationship, and that the amount of expansion slows down as the value of influence per dollar goes up.
- *What are the benefits of running incentivization campaigns that target basic users?* The agent-based DSS was able to evaluate the effect of different incentivization policies on basic users. An important insight for marketers was the need to target users with a high number of premium friends to generate more value through new premium adoptions (e.g., having six or more premium friends was the best tested policy).

FINAL DISCUSSION

Previous research has studied WOM programs and viral marketing (De Bruyn and Lilien 2008; Dellarocas 2006; Van der Lans et al. 2010), and it has become clear that there are important concerns modelers must take into account when building a successful WOM-related DSS and to encourage a significant use by practitioners (Lilien 2011). Our methodological framework proposes an agent-based DSS to model the WOM dynamics and considers several guidelines that may allow the successful managerial adoption of their results. The cornerstone method of the framework, ABM, has already shown that it can capture both the social network structure and complex phenomena of customer interactions in previous research on WOM programs (Haenlein and Libai 2013; Libai, Muller, and Peres 2013; Schlereth et al. 2013), but in this article we develop a generalizable way to build an agent-based DSS to assist WOM decisions and programs.

Building and Using the DSS for a Freemium App

We presented the application of the agent-based DSS framework to a real hedonic app, *Creature Party*. The DSS application illustrated how to use the modeling guidelines and model construction steps to create a DSS for this particular app context, taking into account the data provided by the company. We used the Bass-ABM and two variants of the complex contagion to model the premium adoption of the app. The use of the complex contagion model (Centola and Macy 2007) came from the app data analysis because the adoption of premium content by a basic user appears to require a minimum number of premium contacts. We are not aware of any other study on marketing in which adoption or WOM effects have been modeled using complex contagion, but it works well in this context, and we encourage future researchers to explore the use of complex contagion models in other marketing research.

The use of a GA-based calibration method was helpful in validating the models of the DSS with respect to the empirical app data and in understanding parameter variations and appropriate values of the models. The validation step showed that complex contagion achieves better results than the Bass-ABM and simple logistic models when forecasting if a user is going to adopt premium content in the near future (micro level). Complex contagion supports the freemium model hypothesis of Bapna and Umyarov (2015), in which the authors showed that the effect of peer influence is moderated by the user's number of friends, but users with a smaller number of friends are more susceptible. However, complex contagion performs slightly worse for macro-level forecasting (fitting the historical premium evolution). Therefore, and although the differences in the macro-level forecasts are not huge, the best DSS configuration for the *Creature Party* setting is to use the Bass-ABM to forecast the premium evolution of the market (macro) and the complex contagion model to individually forecast if a user is adopting (micro).

We examined a set of scenarios and policies by using the DSS to understand how to better expand the market by using different reward policies. We ran a sensitivity analysis to show that WOM leads to more premium members and enhances the effects of traditional activities (e.g., promotions or rewards) as also demonstrated by Trusov, Bucklin, and Pauwels (2009) and Wuyts et al. (2004). We confirmed that, for this freemium app setting, the lift in premium adoptions when rewarding users is not linear. This means when the social influence of users' relations overcomes a certain value, increasing the reward does not significantly increase premium expansion.

The joint use of micro- and macro-level forecasts in a validated DSS created a powerful decision-making tool to run marketing policies for a freemium business model. Both micro- and macro-level forecasts are employed to simulate the effects of different targeting policies for the most likely basic users to adopt, a strategy to target revenue leaders (Haenlein and Libai 2013; i.e., those expected to generate high profitability on their own). We compared three different configurations for targeting a group of basic users and observed a notable lift in the number of additional premiums if incentives affect social influence and external influence considerably.

Limitations and Future Studies

Our hope is that this study enhances research and development into agent-based DSSs and constitutes an important step toward the use of DSSs with high managerial success. Indeed, there are many ways researchers can expand the capabilities of the framework, such as by including information about how influential a user is when targeting (Trusov, Bodapati, and Bucklin 2010). Following this research line, practitioners could compare the premium expansion implications of targeting users on the basis of different social network features, similar to the high-degree and high-betweenness seeding strategies suggested by Hinz et al. (2011). We also assume that the adoption model of users is homogeneous across all users. Future studies could explore the role of heterogeneity in adoption processes.

Although we built a DSS for a particular freemium app following the guidelines and proposed steps, the main purpose of this article is to identify a set of guidelines and construction steps for the creation of a DSS for WOM programs. We would like further research to extend our work by considering more freemium scenarios and marketing insights such as the role of functionality and content updates, the customer perceived value of new software features, or the impact of including more social features versus nonsocial functionalities. A great deal of work is still needed to fully understand the best freemium model practices, but we believe that the DSS created here is a first step toward exploring these questions.

REFERENCES

Anderson, Chris (2009), *Free: The Future of a Radical Price*. New York: Hyperson.

Axelrod, Robert (1997), *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton, NJ: Princeton University Press.

Bapna, Ravi, and Akhmed Umyarov (2015), "Do Your Online Friends Make You Pay? A Randomized Field Experiment on Peer Influence in Online Social Networks," *Management Science*, 61 (8), 1902–20.

Barabási, Alberto Laszlo, and Reka Albert (1999), "Emergence of Scaling in Random Networks," *Science*, 286 (5439), 509–12.

Bass, Frank (1969), "A New Product Growth Model for Consumer Durables," *Management Science*, 36 (5), 1057–79.

Biyalogorsky, Eyal, Eitan Gerstner, and Barak Libai (2001), "Customer Referral Management: Optimal Reward Programs," *Marketing Science*, 20 (1), 82–95.

Centola, Damon, and Michael Macy (2007), "Complex Contagions and the Weakness of Long Ties," *American Journal of Sociology*, 113 (3), 702–34.

Chai, Tianfeng, and Roland Draxler (2014), "Root Mean Square Error (RMSE) or Mean Absolute Error (MAE)?—Arguments Against Avoiding RMSE in the Literature," *Geoscientific Model Development*, 7, 1247–50.

Cheng, Hsing Kenneth, and Qian Candy Tang (2010), "Free Trial or No Free Trial: Optimal Software Product Design with Network Effects," *European Journal of Operational Research*, 205 (2), 437–47.

Cheung, Christy, and Dimple Thadani (2012), "The Impact of Electronic Word-of-Mouth Communication: A Literature Analysis and Integrative Model," *Decision Support Systems*, 54 (1), 461–70.

Chica, Manuel, Jose Barranquero, Tomasz Kajdanowicz, Oscar Cordón, and Sergio Damas (2017), "Multimodal Optimization: An Effective Framework for Model Calibration," *Information Sciences*, 79–97.

Chica, Manuel, Oscar Cordón, Sergio Damas, and Joaquin Bautista (2010), "Multiobjective, Constructive Heuristics for the 1/3 Variant of the Time and Space Assembly Line Balancing Problem: ACO and Random Greedy Search," *Information Sciences*, 180 (18), 3465–87.

Chica, Manuel, Oscar Cordón, Sergio Damas, and Joaquin Bautista (2011), "Including Different Kinds of Preferences in a Multi-Objective Ant Algorithm for Time and Space Assembly Line Balancing on Different Nissan Scenarios," *Expert Systems with Applications*, 38 (1), 709–20.

De Bruyn, Arnaud, and Gary Lilien (2008), "A Multi-Stage Model of Word-of-Mouth Influence Through Viral Marketing," *International Journal of Research in Marketing*, 25 (3), 151–63.

Dellarocas, Chrysanthos (2006), "Strategic Manipulation of Internet Opinion Forums: Implications for Consumers and Firms," *Management Science*, 52 (10), 1577–93.

Delre, Sebastiano, Wander Jager, Tammo Bijmolt, and Marco Janssen (2010), "Will It Spread or Not? The Effects of Social Influences and Network Topology on Innovation Diffusion," *Journal of Product Innovation Management*, 27 (2), 267–82.

Dorigo, Marco, Vittorio Maniezzo, and Alberto Colorni (1996), "Ant System: Optimization by a Colony of Cooperating Agents," *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics*, 26 (1), 29–41.

Epstein, Joshua (2006), *Generative Social Science: Studies in Agent-Based Computational Modeling*. Princeton, NJ: Princeton University Press.

Feo, Thomas, and Mauricio Resende (1995), "Greedy Randomized Adaptive Search Procedures," *Journal of Global Optimization*, 6, 109–33.

Garcia, Rosanna (2005), "Uses of Agent-Based Modeling in Innovation/New Product Development Research," *Journal of Product Innovation Management*, 22 (5), 380–98.

Godes, David, and Dina Mayzlin (2009), "Firm-Created Word-of-Mouth Communication: Evidence from a Field Test," *Marketing Science*, 28 (4), 721–39.

Goldberg, David, and John Holland (1988), "Genetic Algorithms and Machine Learning," *Machine Learning*, 3, 95–99.

Goldenberg, Jacob, Sangman Han, Donald Lehmann, and Jae Weon Hong (2009), "The Role of Hubs in the Adoption Process," *Journal of Marketing*, 73 (2), 1–13.

Goldenberg, Jacob, Barak Libai, and Eitan Muller (2001), "Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth," *Marketing Letters*, 12 (3), 211–23.

Goldenberg, Jacob, Barak Libai, and Eitan Muller (2010), "The Chilling Effects of Network Externalities," *International Journal of Research in Marketing*, 27 (1), 4–15.

Granovetter, Mark (1978), "Threshold Models of Collective Behavior," *American Journal of Sociology*, 83 (6), 1420–43.

Guidolin, Mariangela, and Renato Guseo (2014), "Modelling Seasonality in Innovation Diffusion," *Technological Forecasting and Social Change*, 86, 33–40.

Haenlein, Michael, and Barak Libai (2013), "Targeting Revenue Leaders for a New Product," *Journal of Marketing*, 77 (3), 65–80.

Hinz, Oliver, Bernd Skiera, Christian Barrot, and Jan Becker (2011), "Seeding Strategies for Viral Marketing: An Empirical Comparison," *Journal of Marketing*, 75 (6), 55–71.

Holland, John (1975), *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press.

Kennedy, James (2010), "Particle Swarm Optimization," in *Encyclopedia of Machine Learning*, Claude Sammut and Geoffrey Webb, eds. New York: Springer, 760–66.

Kumar, Vineet (2014), "Making Freemium Work: Many Start-Ups Fail to Recognize the Challenges of This Popular Business Model," *Harvard Business Review*, 92 (5), 27–29.

Le Cessie, Saskia, and Hans van Houwelingen (1992), "Ridge Estimators in Logistic Regression," *Applied Statistics*, 41 (1), 191–201.

Lee, Clarence, Vineet Kumar, and Sunil Gupta (2013), "Designing Freemium: A Model of Consumer Usage, Upgrade, and Referral Dynamics," dissertation, Harvard Business School, Harvard University.

Leeflang, Peter, Jaap Wieringa, Tammo Bijmolt, and Koen H. Pauwels (2015), "Building Models for Markets," in *Modeling Markets: Analyzing Marketing Phenomena and Improving Marketing Decision Making*. New York: Springer, 1–24.

Libai, Barak, Ruth Bolton, Marnix S. Bügel, Ko De Ruyter, Oliver Götz, Hans Risselada, et al. (2010), "Customer-to-Customer Interactions: Broadening the Scope of Word of Mouth Research," *Journal of Service Research*, 13 (3), 267–82.

Libai, Barak, Eitan Muller, and Rosanna Peres (2013), "Decomposing the Value of Word-of-Mouth Seeding Programs: Acceleration Versus Expansion," *Journal of Marketing Research*, 50 (2), 161–76.

Lilien, Gary (2011), "Bridging the Academic–Practitioner Divide in Marketing Decision Models," *Journal of Marketing*, 75 (4), 196–210.

Lilien, Gary, John Roberts, and Venkatesh Shankar (2013), "Effective Marketing Science Applications: Insights from the ISMS-MSI Practice Prize Finalist Papers and Projects," *Marketing Science*, 32 (2), 229–45.

Little, John (1970), "Models and Managers: The Concept of a Decision Calculus," *Management Science*, 16 (12), B466–85.

Little, John (2004), "Comments on 'Models and Managers: The Concept of a Decision Calculus'," *Management Science*, 50 (12), 1841–61.

Lourenço, Helena, Olivier Martin, and Thomas Stützle (2003), "Iterated Local Search," in *Handbook of Metaheuristics*, Michel Gendreau and Jean-Yves Potvin, eds. New York: Springer, 320–53.

Lovett, Mitchell, Renana Peres, and Ron Shachar (2013), "On Brands and Word of Mouth," *Journal of Marketing Research*, 50 (4), 427–44.

Luo, Lan (2011), "Product Line Design for Consumer Durables: An Integrated Marketing and Engineering Approach," *Journal of Marketing Research*, 48 (1), 128–39.

Macal, Charles, and Michael North (2005), "Tutorial on Agent-Based Modeling and Simulation," in *Proceedings of the 37th Conference on Winter Simulation*. New York: Association for Computing Machinery, 2–15.

Manchanda, Puneet, Ying Xie, and Nara Youn (2008), "The Role of Targeted Communication and Contagion in Product Adoption," *Marketing Science*, 27 (6), 961–76.

Miller, John (1998), "Active Nonlinear Tests (ANTs) of Complex Simulation Models," *Management Science*, 44 (6), 820–30.

Milo, Ron, Nadav Kashtan, Shalev Itzkovitz, Mark Newman, and Uri Alon (2004), "On the Uniform Generation of Random Graphs with Prescribed Degree Sequences," https://arxiv.org/abs/cond-mat/0312028.

Newman, Mark (2003), "The Structure and Function of Complex Networks," *Society for Industrial and Applied Mathematics Review*, 45 (2), 167–256.

Newman, Mark (2009), "Random Graphs with Clustering," *Physical Review Letters*, 103, 058701.

Newman, Mark, and Albert-László Barabási, and Duncan J. Watts (2006), *The Structure and Dynamics of Networks*. Princeton, NJ: Princeton University Press.

Oestreicher-Singer, Gal, and Lior Zalmanson (2013), "Content or Community? A Digital Business Strategy for Content Providers in the Social Age," *Management Information Systems Quarterly*, 37 (2), 591–616.

Oliva, Rogerio (2003), "Model Calibration as a Testing Strategy for System Dynamics Models," *European Journal of Operational Research*, 151 (3), 552–68.

Peers, Yuri, Denis Fok, and Philip Hans Franses (2012), "Modeling Seasonality in New Product Diffusion," *Marketing Science*, 31 (2), 351–64.

Pitcher, Jenna (2015), "Blizzard Gifts WoW Veterans with Their Own Orc Statue," IGN (January 18), www.ign.com/articles/2015/01/19/blizzard-gifts-wow-veterans-with-their-own-orc-statue.

Rand, William, and Roland Rust (2011), "Agent-Based Modeling in Marketing: Guidelines for Rigor," *International Journal of Research in Marketing*, 28 (3), 181–93.

Rogers, Everett M. (2003), *Diffusion of Innovations*, 5th ed. New York: The Free Press.

Ryu, Gangseog, and Lawrence Feick (2007), "A Penny for Your Thoughts: Referral Reward Programs and Referral Likelihood," *Journal of Marketing*, 71 (1), 84–94.

Saltelli, Andrea, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, et al. (2008), *Global Sensitivity Analysis: The Primer*. Chichester, UK: John Wiley & Sons.

Sargent, Robert (2005), "Verification and Validation of Simulation Models," in Proceedings of the 37th Conference on Winter Simulation. New York: Association for Computing Machinery, 130–43.

Schlereth, Christian, Christian Barrot, Bernd Skiera, and Carsten Takac (2013), "Optimal Product-Sampling Strategies in Social Networks: How Many and Whom to Target?" *International Journal of Electronic Commerce*, 18 (1), 45–72.

Schmitt, Philipp, Bernd Skiera, and Christophe Van den Bulte (2011), "Referral Programs and Customer Value," *Journal of Marketing*, 75 (1), 46–59.

Serrano, Angeles, and Marián Boguná (2005), "Tuning Clustering in Random Networks with Arbitrary Degree Distributions," *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, 72 (3), 036133.

Stonedahl, Forrest, and William Rand (2014), "When Does Simulated Data Match Real Data? Comparing Model Calibration Functions Using Genetic Algorithms," in *Advances in Computational Social Science*, Shu-Heng Chen, Takao Terano, Ryuichi Yamamoto, and Chung-Ching Tai, eds. New York: Springer, 297–313.

Stonedahl, Forrest, William Rand, and Uri Wilensky (2010), "Evolving Viral Marketing Strategies," in *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*. New York: Association for Computing Machinery, 1195–202.

Talbi, El-Ghazali (2009), *Metaheuristics: From Design to Implementation*. Hoboken, NJ: John Wiley & Sons.

Teece, David (2010), "Business Models, Business Strategy and Innovation," *Long Range Planning*, 43 (2/3), 172–94.

Terano, Takao (2008), "Beyond the KISS Principle for Agent-Based Social Simulation," *Journal of Socio-Informatics*, 1 (1), 175–87.

Terblanche, Nic S. (2015), "Customers' Perceived Benefits of a Frequent-Flyer Program," *Journal of Travel & Tourism Marketing*, 32 (3), 199–210.

Trusov, Michael, Anand Bodapati, and Randolph Bucklin (2010), "Determining Influential Users in Internet Social Networks," *Journal of Marketing Research*, 47 (4), 643–58.

Trusov, Michael, Randolph Bucklin, and Koen Pauwels (2009), "Effects of Word-of-Mouth Versus Traditional Marketing: Findings from an Internet Social Networking Site," *Journal of Marketing*, 73 (4), 90–102.

Trusov, Michael, William Rand, and Yogesh Joshi (2013), "Improving Prelaunch Diffusion Forecasts: Using Synthetic Networks as Simulated Priors," *Journal of Marketing Research*, 50 (4), 675–90.

Van den Bulte, Christophe, and Yogesh Joshi (2007), "New Product Diffusion with Influentials and Imitators," *Marketing Science*, 26 (3), 400–21.

Van der Lans, Ralf, Gerrit Van Bruggen, Jehoshua Eliashberg, and Berend Wierenga (2010), "A Viral Branching Model for Predicting the Spread of Electronic Word of Mouth," *Marketing Science*, 29 (2), 348–65.

Venkatesan, Rajkumar, Trichy Krishnan, and Vineet Kumar (2004), "Evolutionary Estimation of Macro-Level Diffusion Models Using Genetic Algorithms: An Alternative to Nonlinear Least Squares," *Marketing Science*, 23 (3), 451–64.

Viger, Fabien, and Matthieu Latapy (2005), "Efficient and Simple Generation of Random Simple Connected Graphs with Prescribed Degree Sequence," in *Proceedings of the 11th Annual International Conference on Computing and Combinatorics*. New York: Association for Computing Machinery, 440–49.

Voinov, Alexey, and Francois Bousquet (2010), "Modelling with Stakeholders," *Environmental Modelling & Software*, 25 (11), 1268–81.

Watts, Duncan, and Peter Dodds (2007), "Influentials, Networks, and Public Opinion Formation," *Journal of Consumer Research*, 34 (4), 441–58.

Wierenga, Bernd, Gerrit van Bruggen, and Niek Althuizen (2008), *Advances in Marketing Management Support Systems*. New York: Springer.

Wilensky, Uri, and William Rand (2015), *Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. Cambridge, MA: MIT Press.

Witten, Ian, and Eibe Frank (2005), *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Francisco: Morgan Kaufmann.

Wuyts, Stefan, Stefan Stremersch, Christophe Van den Bulte, and Philip Franses (2004), "Vertical Marketing Systems for Complex Products: A Triadic Perspective," *Journal of Marketing Research*, 41 (4), 479–87.

Xie, Karen, Lina Xiong, Chih-Chien Chen, and Clark Hu (2015), "Understanding Active Loyalty Behavior in Hotel Reward Programs Through Customers Switching Costs and Perceived Program Value," *Journal of Travel & Tourism Marketing*, 32 (3), 308–24.