Comput Math Organ Theory (2013) 19:42–77
DOI 10.1007/s10588-012-9137-7

MANUSCRIPT

Innovation and learning performance implications of free revealing and knowledge brokering in competing communities: insights from the Netflix Prize challenge

J. Andrei Villarroel · John E. Taylor · Christopher L. Tucci

Published online: 7 November 2012

Abstract Firms increasingly use open competitions to extend their innovation process and access new diverse knowledge. The Netflix Prize case we study in this paper is a multi-stage repeat-submission open competition involving the creation of new knowledge from across knowledge domains, a process which benefits from knowledge sharing across competing communities. The extant literature says little about the effects of different types and levels of knowledge sharing behavior on the learning and innovation outcomes of such a competitive system, or what the performance boundaries may be for the system as a result of such differences. Our research explores those boundaries unveiling important tradeoffs involving *free revealing* behavior—defined as voluntarily giving away codified knowledge and making it into a 'public good'—and *knowledge brokering* behavior—defined as using knowledge from one domain to innovate in another—on the learning performance of *competing communities*. The results, analyzing the system-level average and volatility of learning outcomes, lead to three conclusions: (i) greater knowledge sharing, as portrayed by greater free revealing and knowledge brokering, helps achieve better average learning for the system as a whole, however, (ii) achieving the best overall outcome possible from the system actually requires controlling the amount of knowledge brokering activity in the system. The results further suggest that (iii) it should not be possible to simultaneously achieve both the best overall outcome from the system and the best average learning

(iii) surprises me on first sight

J.A. Villarroel (✉)
Católica-Lisbon School of Business an[illegible] Cima, 1649-023 Lisbon, Portugal
e-mail: andreiv@mit.edu

J.E. Taylor
Department of Civil & Environmental Engineering, Virginia Tech, 113B Patton Hall, Blacksburg, VA 24061, USA

C.L. Tucci
Ecole Polytechnique Fédérale de Lausanne (EPFL), Chair of Corporate Strategy and Innovation, College of Management of Technology, Odyssea 1.04, Station 5, 1015 Lausanne, Switzerland

for the system. The tradeoffs that ensue from these findings have important implications for innovation policy and management. This research contributes to practice by showing how it is possible to achieve different learning performance outcomes by managing the types and levels of knowledge sharing in open competitive systems.

Keywords Managing online communities · Free revealing · Knowledge brokering · Organizational learning · Crowdsourcing · Computer simulation

1 Introduction

Over the last decade, firms have increasingly sought to leverage open sources of innovation outside of their own internal resources, some with a good deal of success. Early on, these were best exemplified by public collective endeavors such as the Linux operating system and the Wikipedia online encyclopedia (Lee and Cole 2003; Giles 2005). Since then, *communities* of individuals beyond the formal boundaries of the firm have been growing in importance as a source of innovation (Jeppesen and Lakhani 2010; Terwiesch and Xu 2008; Schwen and Hara 2003). Simultaneously, open source processes, involving the *free revealing* and *accumulation* of codified knowledge assets on the Internet, have gained momentum as an effective organizational model for innovation (Murray and O'Mahony 2007; Stam 2009; von Hippel and von Krogh 2003). More recently, *competing communities* have been successfully harnessed as an effective source of innovation in private online competitive endeavors actively encouraging the *free revealing* of codified information, as the Netflix Prize case illustrates (Bennett and Lanning 2007). Multi-stage repeat-submission contests like the Netflix Prize enable *knowledge brokering* (Hargadon 1997), an interactive search process by which knowledge from different domains is recombined to enact

what are "different domains" and "same context" in Netflix case?

While manage

vealing (von Hippel and von

1998, 2002) in different contexts, little attention has been given to exploring and untangling their effects when they occur in the same context. More generally, the growing number of initiatives adopting open source processes (Pénin 2011), for profit and not, brings communities into competition for unique and scarce knowledge resources scattered across a finite number of prospective individual members. Little research has been done to understand the learning dynamics across *competing communities* in this finite context, particularly regarding the effects of *free revealing* and *knowledge brokering* behaviors (cf. Sect. 2: Background). An important reason for this gap is the relative novelty of combining free revealing, knowledge brokering, and competing communities as an interactive search strategy conducive to better learning performance. In this sense, we use the Netflix Prize challenge a a relevant case study where these concepts are unveiled through participant su y data, solution submission data, and qualitative research. (cf. Sect. 3: Evidence tilizing these data and insights hinted at by our study of the Netflix Prize case propose a learning model that quantitatively brings together these concer d allows us to further investigate—through computer simulation- rning performance (cf. Sect. 4: Model).

Still not sure I understand knowledge "brokering" because I don't understand "different domains" "overall system"

Our research finds that in a competitive system combining different levels of free revealing and knowledge brokering, different patterns of knowledge diffusion arise and with them different learning rates for the overall system. At the outset, these differences result in differences in innovation outcomes for the system as a whole (cf. Sect. 5: Results). Our model's simulation results replicate the empirical evidence from the Netflix Prize, analyzed earlier, contributing to explain how learning performance across competing communities is influenced by different levels of free revealing and knowledge brokering behaviors in individuals (cf. Sect. 6: Analysis). Finally, we discuss the implications of our findings for innovation theory and practice (cf. Sect. 7: Conclusions).

coarse but
interesting observations

2 Background

The interest on *communities* as an open source of competitive advantage that the firm can manage has increased in recent years, and with it their prominence in the management literature (see Villarroel 2008 for an early review and discussion of this phenomenon). As of July 2011, over 138,000 articles in Google Scholar referred to *communities* in the context of "open source", while 4,650 articles held these terms in their title. However, little of the literature had focused on formally defining the term itself, particularly when it comes to the firm using such a mechanism, otherwise referred to as 'crowdsourcing' (Howe 2009). Only about 22 of the articles seek to define the term *community* in any way (Google Scholar 2011), and when they do, the definitions differ in substantial ways. Regarding this lack of consensus, West and Lakhani suggest that to prevent confusion, authors should explicitly articulate what they mean (West and Lakhani 2008, p. 224). We therefore begin by defining what we mean by the term *community*, a key construct in our discussion.

Definition of community A community is a population of individuals[1] that grows in membership by virtue of having individuals successfully execute project opportunities together, developing and sharing unique knowledge on how to effectively work well with one another. The greater the membership, the greater the knowledge shared amongst its members, and the more influential the community. In the absence of knowledge brokering and free revealing behaviors in individual members, once the community is formed, it is assumed to be independent from other communities (cf. Sect. 4.4, Sect. 5 Table 2 baseline scenario).

A central aspect of our paper discusses how individual members of disjoint 'competing communities' hold scarce knowledge resources that *could*: (1) be codified and shared with everyone (free revealing), typically using digital media over the Internet, or (2) be carried along by certain individuals *across* knowledge domains (knowledge brokering), through their interactions with others, to effectively address innovative new project opportunities. In the next subsections we review the relevant literature to inform our modeling of these two behaviors in the context of competing communities (cf. Sect. 4), keeping in mind that our core contribution is to unveil the effects of their interactions on learning performance outcomes (cf. Sect. 5).

[1] Individuals are also referred to as *contributors* to an initiative, or as *agents* in the model introduced later.

2.1 Free revealing

In the last two decades we have seen the emergence of free, "libre", and open source software projects (FLOSS), where the principal actors of value creation are individuals with no formal organizational affiliation who freely reveal software code they produce. Prime examples are the Linux operating system, the Apache web server, and the MySQL database system now widely used as a software system suite in the enterprise (Krishnamurthy 2003). Several online software code repositories, or "software commons", have been created to host such projects. The largest of these is sourceforge.net, with over 230,000 projects and 2 million registered users (van der Waal 2010), a number that has grown to 324,000 projects and 3.4 million developers as of 2012.[2] In his investigation of these projects, Krishnamurthy (2002) found that despite the high numbers of projects and developers, the median number of contributors *per* project is only four. We later use this value to inform our model.

how has this changed?

Over this same period, management scholars have set the foundation for the study of *free revealing* by focusing on open source software (von Hippel and von Krogh 2003; Lakhani and von Hippel 2003). Users have been found to benefit from the *free revealing* associated with FLOSS (Harhoff et al. 2003), while a mixed-model of deliberate "selective revealing" (partly sharing, partly protecting) has been found to be advantageous to the firm (Henkel 2006). Indeed, Henkel (2006, p. 960) found empirically that individual contributors to open source software projects from commercial organizations reveal only about half of the code they produced.

an interesting group — open source and commercial allegiances

Nonetheless, the free revealing phenomenon has been widely studied in the context of industries other than software (e.g. Franke and Shah 2003), referred to recently as "open source beyond software" (e.g. Balka et al. 2009). In this literature, free revealing is defined as voluntarily giving up intellectual property rights on once proprietary information and granting free access to that information to all interested parties equally—namely making that information a public good, characterized by non-excludability and non-rivalry (Fauchart and Von Hippel 2008, p. 191; Harhoff et al. 2003, p. 1753; von Hippel and von Krogh 2006, p. 295).

For the remainder of this paper, unless specifically noted otherwise, we will refer to *free revealing* to mean the open disclosure of codified knowledge by an individual contributor to all others, the further consumption of which does not require direct interaction between individuals.

2.2 Competition across communities

Large corporations such as IBM, Novell, and Sun Microsystems, among others have been described in Austin (2004), Bagley and Lane (2006), MacCormack (2002), and O'Mahony et al. (2005) as historically very involved in open source software initiatives. In turn, firm-sponsored communities standing behind those firm's open source corporate initiatives, such as Sun Microsystems's OpenSolaris (a free open source operating system), faced competition from several other open source software communities. On one side, the OpenSolaris community was competing for resources with

[2] See: www.sourceforge.net/about (Last accessed July 24, 2012).

self-organized communities standing behind public open source Linux-based projects (e.g., Debian, Slackware, and Ubuntu, which are also free open source operating systems). On the other side, they were competing with other firm-spons
nities standing behind private interests (e.g., Novell's openSUSE, R
and Linspire's Freespire).

interesting paragraphs

These various operating systems are technically similar (UNIX-like) and appeal to similar potential contributors as well. As a result, the different communities associated with each of the initiatives compete for the same (limited) pool of innovative talent, namely individuals with unique and scarce knowledge, to join and contribute to their community. This creates a conflicting situation for potential contributors who likely have to choose a community to which to devote their own limited resources. Once individual contributors develop experience with a particular community, a relationship-specific investment develops among the members of that community who learn how to effectively work with one another over time. To the extent that these individual contributors develop experience together in joint problem solving and artifact creation on behalf of the initiative, they develop a sense of community (Knorr Cetina 1999). This can make collaboration amongst members of different communities more difficult.

According to Wenger et al. (2002, p. 141), the "very qualities that make a community an ideal structure for learning—a shared perspective on a domain, trust, a communal identity, longstanding relationships, an established practice—are the same qualities that hold it hostage to its history and its achievements". When specialists with prior disjoint experiences as members of different communities are presented with an opportunity to work together, they face an impasse situation (Handley et al. 2006). While they may be able to take advantage of the opportunity to collaborate, each specialist may also have a vested interest in the relationship developed through the experience with his or her own community. Contributors from different communities may thus refrain from working together due to concerns that they may give other communities increased knowledge bargaining power (Yanow 2004), lose trust from their respective communities (Lazaric and Lorenz 1998), or they may simply do it out of habit working solely with their peers (Mutch and Roberts 2006).

so "domain" might not mean "movies vs fine art", but literally a place or community?

2.3 Knowledge brokering, communities and the social boundaries of knowledge

Hargadon (1998, p. 214) first described knowledge brokers as individuals or organizations located between otherwise disconnected groups that profit by "transferring ideas from where they are known to where they represent innovative new possibilities". Later, adopting a microsociological perspective, Hargadon (2002) sought to explain the importance of the social and structural linkage in knowledge brokering relating learning and innovation. On the one hand, the solutions learned by individuals are only useful when they can be applied to the right problems, but those problems may structurally lie away from the individuals. On the other hand, for innovation to occur individuals first need to be exposed to new situations in which to enhance their knowledge and then they need to develop new networks to support it. As such, knowledge brokering behavior entails bridging knowledge from different social domains and linking it to new situations while building new social networks around the innovations created (Hargadon 2002, p. 41).

In this paper, (i) the social boundaries of knowledge domains are the *communities*, (ii) the individual agents are the *knowledge holders*, and (iii) the individual's boundary spanning behavior is referred to as *knowledge brokering*. Building on Hargadon's qualitative concept, Hsu and Lim (2011) show quantitatively—at the firm level—that knowledge brokering behavior can be an important organizational capability and a determinant of innovation performance across competing firms. They describe knowledge brokering as a recombination phenomenon, namely taking knowledge from one (or more) domain(s) and reapplying it to another domain in order to innovate. Translating this argument to the context of our paper, we suggest knowledge brokering can be a desirable behavior in individual contributors, likely leading to better learning performance by recombining knowledge from across competing communities.

social boundaries

The spectrum of knowledge brokering comprises the following two extreme situations. On one end, a single individual knowledge holder from one given community successfully works with individuals all belonging to another community. On the other end of the spectrum, several individuals each belonging to a different community successfully work together, effectively recombining their knowledge to innovate. In both successful circumstances learning occurs that benefits all the individuals involved, while at the same time modifying the social boundaries of knowledge.

don't seem particularly extreme

2.4 Knowledge brokering and learning performance

the exploitation - exploration tradeoff is ubiquitous in machine learning, and presumably human learning as well

collective search for a solution

The literature shows that boundary spanning search behavior of this kind has a positive association with innovation and learning performance, but falls short quantifying the effects of knowledge brokering.

aside: in coopetition, competition encourages each team (path, thread) to work its hardest, and collaboration communicates between teams (search paths)

March (1991), discussing the role of exploration relative to exploitation in learning, was among the first to highlight how local feedback—characteristic of exploitation—produces strong path dependence and therefore suboptimal innovation outcomes; hence the importance of exploration or search behavior. Furthering this view, Ahuja and Lampert (2001) argued against innovation traps arising from search favoring the familiar, mature, and local. Based on their investigation of patent and patent citation activity in the chemical industry, they suggest experimenting with the new and unfamiliar as a means to achieve breakthrough inventions. Rosenkopf and Almeida (2003), studying patent data in the semiconductor industry, suggest bridging geographic and technological distance by developing alliances and encouraging labor mobility, as a means to overcome the limitations of local search. Combining the new and unfamiliar and integrating it in a new context defines the specific kind of interactive search or exploration behavior we here refer to as *knowledge brokering*.

Brown and Eisenhardt (1997) contributed to a complex system theory in organizations by unveiling the importance of semistructures, namely organizational forms that exhibit partial order between the very rigid and the completely chaotic, which are synergistic with boundary spanning behavior. We assimilate such semi structured organizational form to the *community*, and use the microsociological perspective adopted by Hargadon (2002) as a point of departure for studying the dynamic effects of boundary spanning behavior on learning performance. In our model, knowledge brokering implies the crossing of community boundaries by a knowledge holder—an individual learning agent holding knowledge of his own, who finds new use for it in a new situation, where the outcome is socially supported by other learning agents (from other communities).

3 Evidence from the Netflix Prize case

The Netflix Prize challenge offers a valuable case study that makes concrete the concepts studied in this research. On one side, Netflix managed to mobilize external communities with distinct knowledge backgrounds by means of an open competition for which it (i) freely revealed a large amount of data, and (ii) offered an online forum where participants could interact. On the other side, the competing communities formed into teams to work with the freely revealed data using their own algorithms, as well as with the algorithms of others, in order to develop new and improved ones. As such, the Netflix Prize challenge offers insights on how individual participants from different knowledge communities incurred in *free revealing* and *knowledge brokering* behavior in order to find new solutions. We informed our analysis of this case using qualitative evidence, solution submission data over a one-year period, and data from a survey of participants conducted shortly after the challenge was over (November 2009). The insights from this analysis suggest an important association between knowledge sharing behaviors of the participants and system-level patterns of learning performance. Finally, we use these insights to inform a model which investigates this association further (cf. Sect. 4).

3.1 The Netflix Prize challenge

Netflix is an online movie-rental company with $3.2 billion in sales and 26 million subscribers by yearend 2011. Netflix's interactions with its customers occur solely online and one source of competitive advantage lies in its ability to offer its customers with movie recommendations tailored to their individual tastes. Therefore, one of the company's core assets is the information it collects about its customers, particularly their individual movie ratings. The company processes this information to turn it into new movie recommendations using a sophisticated recommendation algorithm.[3]

In 2006, Netflix made a bold attempt to improve their recommendation algorithm by massively engaging with experts from outside of their own firm through an open challenge on the Internet. The Netflix Prize was a contest devised to attract top talent from around the world to help Netflix improve its movie recommendation system, at

[3] A recommendation algorithm automatically suggests targeted products to a user based on his or her available information and that of other people with similar information. Recommendation algorithms use different kinds of input data, such as product and consumer attributes, and transactional data involving consumers and products (e.g., buying, rating, browsing). Algorithms vary in implementation, including techniques such as regression, classification, collaborative filtering, link analysis, among others. Companies such as Amazon.com rely on recommendation systems to increase online sales and improve customer loyalty.

a time when the company rented out movies solely on DVD.[4] In this challenge, talent was drawn from various communities, including academic communities in different fields and industry participants.

On October 2, 2006, Netflix kicked off the contest freely revealing an unprecedented data set of (anonymized) customer data—it opened up 100 million entries of its customer's movie ratings database free for download—, and offered a $1-million reward in exchange for a 10 % improvement to its movie recommendation system. Netflix decided on a performance metric (RMSE, see footnote 5) to benchmark the solutions of others, using the company's own solution's RMSE as the baseline (Netflix Cinematch RMSE score was 0.9525). The contest—meant to last at least 5 years—explicitly stated in its terms that: "to win, . . . you must share your method with Netflix . . . and you must describe to the world how you did it and why it works" (Netflix Prize Rules 2006). Netflix provided a dedicated website and community platform to facilitate communication and interaction among participants (Netflix Prize Community 2008).

One year after the launch of the contest, Netflix took advantage of the ac[illegible] effort of over 20,000 submissions originating from over 160 countries. In [illegible] 2007, the company paid a 'progress prize' of $50,000 (Netflix 2007) for the [illegible] performing solution (less than the 10 % required for the Grand Prize) from the [illegible]bined team KorBell which offered an 8.43 % improvement to its movie recommendation system. The amount of this reward is small given the level of computational and mathematical skill, and compound collective effort displayed by the then over 30,000 Netflix Prize participants. On the one hand, the use of an open challenge to improve the recommendation algorithm was cost-effective. On the other hand, of key interest to our analysis, there were important knowledge spillovers influencing the outcomes achieved.

how important is the statement of the goal to the behavior that emerged?

3.2 Free revealing and knowledge brokering

A closer look at the Netflix Prize challenge offers empirical evidence of the two knowledge sharing behaviors of interest in this study: free revealing and knowledge brokering. We document these behaviors through qualitative data analysis and using data from a survey of challenge participants.

Qualitative evidence: illustrative examples Studying the Netflix Prize forum, the Netflix Prize leaderboard, and the anecdotal evidence surrounding the progress prize awards, we found qualitative evidence of free revealing and knowledge brokering behavior. For instance, on January 7, 2007 a participant team made its entire source

[4]Netflix began offering streaming video content one year after the launch of the Netflix Prize challenge. At the end of the challenge (2009), the company continued to rent out the majority of their movies on DVD, although the 'on demand' streaming video business was already expanding rapidly. The new possibilities that customers had, to instantly watch movie trailers online before renting a movie, or to instantly switch movies half way, would soon dramatically change how customers would choose the next movie to watch. These new behaviors were non-existent in the customer database that Netflix had cumulated during the DVD era and shared for this challenge, likely limiting the future applicability of the solutions developed by challenge participants.

free revealing

code freely available to everyone, which sparked an open exchange of source code, explanations, and extensions to the code, among a number of participants in a single forum thread with 77 postings that lasted until May 1, 2009 (Netflix Free Revealing 2007). The codified information made available online constitutes a knowledge commons from which all communities may benefit. In another instance, as a progress prize milestone approached, two teams from historically different communities of practice (Dinosaurs and Gravity) joined together and combined their technologies into one which furthered their position on the Netflix Prize Leaderboard (Netflix Knowledge Brokering 2008). In fact, knowledge brokering of this kind occurred mostly before the awarding of each yearly progress prize, and with greater frequency as the award date of the final prize got closer. At the outset of each award, this knowledge brokering activity resulted in a better performing algorithm judging from the achieved performance of the combined solutions on the leaderboard.

knowledge brokering

An illustrative example linking free revealing and knowledge brokering to measurable performance is when the first-year progress prize was awarded to KorBell (USA). They presented their work to the Netflix Prize community at large, revealing in verbal and in codified form how their solution actually combined several different approaches developed by others (Bell and Koren 2007; Bell et al. 2007). This occurred once again in reaching the second-year progress prize, awarded to "BellKor in BigChaos". Finally, on September 21, 2009, the winning team was announced: "BellKor's Pragmatic Chaos". It was a combination of three teams, "BellKor", "BigChaos", and "Pragmatic Theory". This evidence suggests that free revealing and knowledge brokering appear to have played a significant role enabling the recombination of knowledge to achieve better outcomes.

free revealing a knowledge brokered solution

I can imagine that three teams for to build a game interface in Part 2 that they then make available to all

Survey of participants: the big picture To better understand the extent of the open disclosure of codified knowledge (free revealing), and the importance of the need for new knowledge and the recombination of knowledge across domains of expertise (knowledge brokering), we surveyed a sample of 211 Netflix Prize contest participants in November 2009 (see Table 1).

The answers to the survey show that contestants appreciated that Netflix freely revealed its customer preference database to facilitate their work (Table 1, Question 1). In addition, participants recognized their need for new knowledge when preparing a submission to the Netflix Prize (Table 1, Question 2). Specifically looking at free revealing behavior, the survey reveals that over the course of their participation in the Netflix Prize, 19 % of participants published the algorithms they developed and 13 % open sourced their software code, while 44 % of participants acknowledged to have incorporated other people's software code into their own solutions (Table 1, Questions 3–5). As far as knowledge brokering behavior is concerned, 51 % of respondents indicated that they discovered completely new fields or techniques from other participants, 55 % said they exchanged ideas with others in the community, and 55 % of participants admitted to having incorporated completely new knowledge into their own work (Table 1, Questions 6–8).

Table 1 Netflix Prize survey responses on free revealing and knowledge brokering

Nr.	Question	Mean	Min	Max	Obs	Type
	(Likert scale 1–7: 1 = strongly disagree, 4 = neutral, 7 = strongly agree)					
1	The data Netflix put online for download strongly motivated my participation	5.13	1	7	211	FR
2	While working on a submission to the NP I sometimes needed knowledge I did not have	6.04	1	7	135	KB
	(Yes or No dummies: 1 = yes, 0 = no)					
Over the course of my participation in the NP:						
3	I published my NP algorithms (online, conference, journal)	0.19	0	1	141	FR
4	I 'open sourced' my NP software (made my code available online)	0.13	0	1	141	FR
5	I incorporated other's software code into my software	0.44	0	1	142	FR
6	I discovered a completely new math or computational technique/field from others	0.51	0	1	142	KB
7	I shared and exchanged ideas with others in the NP community	0.55	0	1	142	KB
8	I incorporated a completely new math or computational technique/field into my own work	0.55	0	1	141	KB

Notes: NP refers to the Netflix Prize *not nondetermistic polynomial!?*

Not least, among the 135 complete responses to the survey we could clearly identify disjoint communities in the sense defined in Sect. 2: mathematicians, economists, computer scientists, chemists, lawyers, physicists.

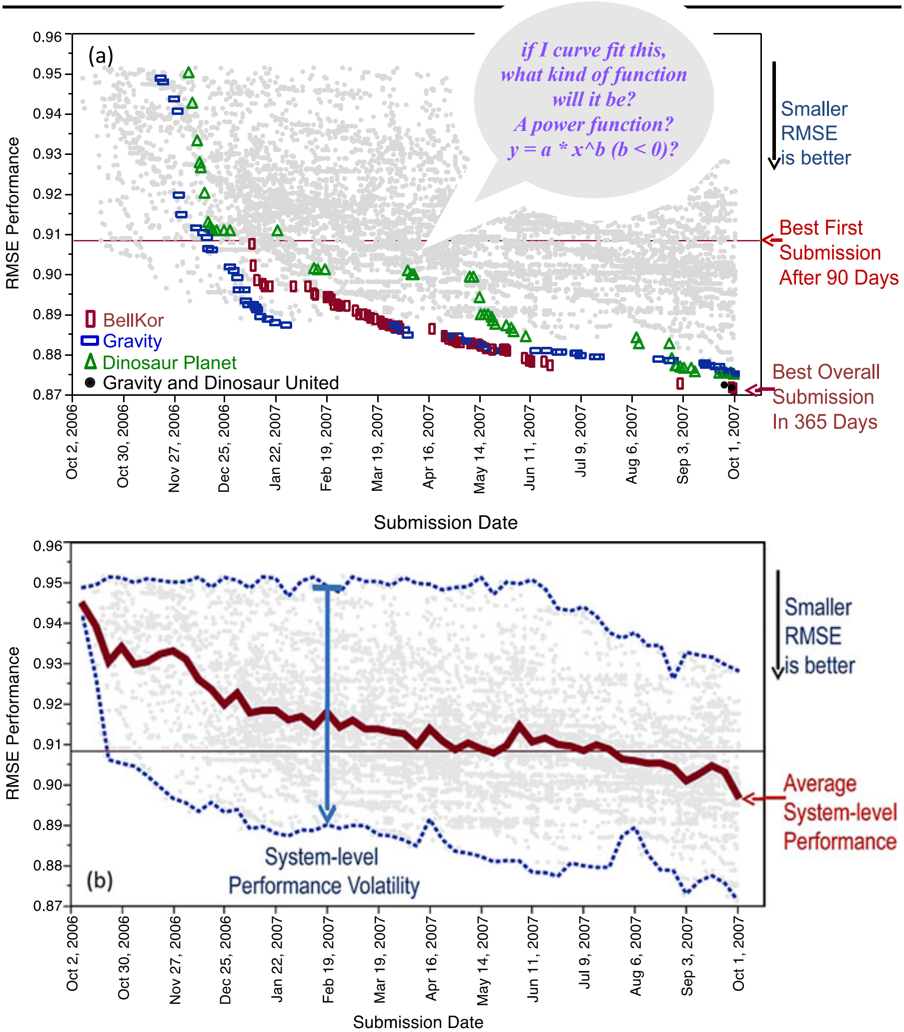
These data show that both free revealing of codified information and knowledge brokering behavior were widespread among contest participants, and further suggest that these behaviors played an important role in the development and improvement of solutions submitted to the Netflix Prize.

3.3 Innovation and learning performance

In order to understand the performance implications of the evidence discussed in Sect. 3.2, we studied 365 days of data from the Netflix Prize Leaderboard covering 7,533 valid submissions to the Netflix Prize starting from the official launch of the contest. Each valid submission in the Netflix Prize leaderboard performed better than the initial Netflix recommendation system, CineMatch (as measured by RMSE, specifically used to compare performance in this contest).[5] These data are depicted in Fig. 1. Figure 1a shows the progress made by 878 registered teams during this one-year period. We defined a reference threshold line for the best-performing first-time

Google it and know the formula and applicability

[5] A smaller RMSE, Root Mean Square Error, means smaller predictive error for movie ratings. A movie recommendation is more or less accurate depending on how close the algorithm's predicted rating—based on historical data—is for the user, relative to the actual observed rating by the user. In other words, if a recommendation system predicts a certain user will rate the movie "Star Trek 27" 5 stars (the highest rating) and we know the user actually rated the film 4 stars, there is a difference of 1 star. These differences are squared and averaged across all predicted-actual pairs (Netflix keeps a separate testing database that it uses to test the performance of proposed algorithms).



Notes: In one year, participants in the Netflix Prize provided 7,533 unique valid submissions that reached the leaderboard. Namely, each of these submissions outperformed the company's own movie recommendation system, Cinematch. A smaller value on the vertical axis of the graph means better performance (i.e. smaller RMSE, Root Mean Square Error, means smaller predictive error for movie ratings).

Fig. 1 Scatter plot of Netflix Prize Leaderboard entries. (**a**) Color adjustment for BellKor. (**b**) Volatility line and label adjusment

submission to the contest within 90 days of its launch. We assume this threshold to represent the state-of-the-art performance (globally) available at the time the contest was launched. We refer to it as the '*pre-innovation performance*' threshold.

To unveil the system-level characteristics in the Netflix Prize submissions data, we computed the weekly average of all submissions and the weekly min-max envelope around the scatterplot of all submissions over the one-year time period. These are depicted in Fig. 1b, and show that (i) the average performance of the submissions made describes a system-level learning curve, and (ii) the volatility of the perfor-

(ii) high volatility at the mid range of learning is ubiquitous

mance of the submissions made first increases and then decreases toward the end of the period observed. In the last two months, the average performance of submissions consistently outperforms the pre-innovation performance threshold. This shows the importance of the introduction of this challenge as an organizational innovation that allowed the system as a whole to learn and produce solutions that on average outperformed the pre-innovation performance threshold.

Taken together, the importance of free revealing and knowledge brokering unveiled by the survey data and the patterns of system-level performance gains just discussed lead us to conclude that contest participants learned a great deal while developing submissions for this initiative. On the one hand, at the outset of one year the majority of participants surpassed the once unknown state-of-the-art performance level prevalent at the time the contest was launched. On the other, the smaller performance volatility observed over the last four months suggests that there is system-level convergence, in spite of having a larger number of participants and submissions over that period.

Generally, a good example of collective learning

3.4 Research question

The evidence presented in this section shows that free revealing and knowledge brokering were both important elements in shaping the solutions submitted to the Netflix Prize challenge. Our analysis of this evidence further suggests a relationship between free revealing, knowledge brokering, and an observed systematic improvement on average learning and a clear pattern affecting the volatility of outcomes over time. Given these results, there is a larger question for research in this competitive context: *How do combinations of different levels of free revealing and knowledge brokering affect system-level learning performance?* In the next section, we develop a model to explore those effects beyond the scope of this one case study, in an effort to generalize the insights derived from it.

4 Model description

Let us begin by considering agents representing individual contributors who: (1) join in teams to execute projects, and, as they successfully execute them, (2) form into communities (spanning contributors to different projects). Membership in project teams evolves in response to new project opportunities. Membership in communities evolves over time when individual contributors engage in knowledge brokering behavior. Moreover, individual contributors may freely reveal a fraction of the knowledge they possess, so that it becomes available to everyone else through a knowledge commons. The model we describe next aims at exploring how these micro behaviors, namely free revealing and knowledge brokering, influence the evolution of learning patterns for the universe of contributors.

The model comprises a total of N contributors, namely $\{1, \ldots, i, \ldots, N\}$, initially assumed to be independent agents with learning capabilities (see Fig. 2). There are M specialized roles, each denoting a specific ability, i.e., $\{R_i, R_j, \ldots, R_M\}$, assumed to be useful to execute a project. A given contributor belongs to one of M roles

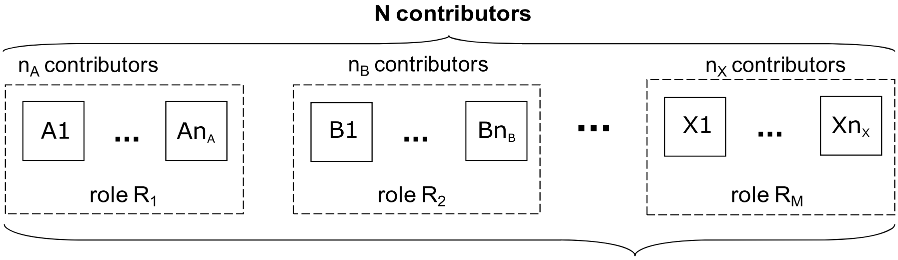


Fig. 2 Specialized contributors as agents in the model

($M \ll N$). In the model, we assume an even number of contributors per specialized role, namely N/M contributors per role. Role assignments are permanent. By contrast, membership in a community results from executing projects with others: upon successfully executing a project with others, a given contributor i becomes a member of a community (e.g. C_i). A community's membership comprises contributors from any role, and community membership evolves as projects opportunities are successfully executed.

4.1 Project opportunities

Opportunities for innovation in our model are represented by the random generation of new project opportunities P. Some are successfully executed (hence feasible and realized). Others cannot be executed (hence possible but not realized). A stochastically generated new project opportunity P involves M contributors who are prospective members of that project's team—recall that there are N/M contributors per role. Initially (at time t_0), within each role, a project contributor i is chosen with uniform probability $p_{i(t0)} = p_0 = 1/(N/M) = M/N$. At this point, note that if p_i remained constant, the emergent network of contributors who faced project opportunities together would be random. Nonetheless, it is now widely acknowledged that most real networks exhibit preferential connectivity (Barabási and Albert 1999). Another way to describe this phenomenon is that "success breeds success". Therefore we add to our model a "preferential attachment" policy, implemented via a probability adjustment function.

Contributors such as i who succeeded at innovation—namely, successfully executed a new project opportunity P—form into a project team and see their probability of being selected again rise[6] by $\Delta prob = p_{adj} \bullet (M/N)$. Hence, if i is successful at executing a project, his new probability of being chosen is adjusted

[6]Our model uses a probability adjustment value of $p_{adj} = 0.5$. We experimented with different values for p_{adj} in the interval (0 to 0.5], with highly consistent results. Extreme values were avoided since unrealistic. Note, for example, that a value of $p_{adj} = 1$ would—upon failure to execute a project involving a first time contributor—lead to having $p_{i(t1)} = p_{i(t0)} - p_{adj} \bullet (M/N) = M/N - 1 \bullet M/N = 0$, which eliminates all failing first-time contributors from immediate future project opportunities. By contrast, a value of $p_{adj} = 0$ would lead to uniformly random draws of contributors for each new project opportunity, regardless of their history of successes or failures, since $p_{adj} = 0$ implies no adjustments are made to the probabilities of being chosen from one project opportunity to another.

upwards as follows: $p_{i(t+1)} = p_{i(t)} + \Delta prob = p_{i(t)} + p_{adj} \bullet (M/N)$. Simultaneously, all others $(N - M)$ see their likelihood of being selected again go down[7] by $\Delta prob/(N/M - 1)$. Conversely, if i is unsuccessful at executing a project opportunity, its probability of being chosen in the next round is adjusted downwards as follows: $p_{i(t+1)} = p_{i(t)} - \Delta prob$. All those outside the project team would have their likelihood of being selected again in the future go up by $\Delta prob/(N/M - 1)$.

At all times, it is probabilistically possible for any contributor to be selected again. Only the probability of this occurring varies. As some unsuccessful contributors see their probabilities reduced—to possibly zero—they become only temporarily marginalized. Indeed, every subsequent project failure (not involving the already marginalized contributors) is sufficient to bring those marginalized contributors back under consideration, since their probabilities go up. Note that whenever a project is executed, it is considered successful since some knowledge and learning are derived from it. Only when the project is not executed is it considered unsuccessful.

The stochastic generation of new project opportunities is intended to model a universe of innovative opportunities; some opportunities more likely to involve successful contributors from the same community, but also other opportunities involving otherwise unrelated contributors from different communities. Whether those contributors successfully execute the opportunity depends on the context and their behavior, as further discussed in Sect. 4.4. In the long run, more new project opportunities will tend to be presented to historically successful contributors and to those contributors who have not yet been involved on a project, while new project opportunities involving previously unsuccessful ones will tend to occur less often.

4.2 Independent and interdependent learning productivity and performance

Our model of learning in individual agent contributors uses an approach related to 'learning by doing', which Levitt and March (1988, p. 321) in their exploration of organizational learning, claimed "[t]his equation, similar in spirit and form to learning curves in individuals and animals, has been shown to fit production costs ... reasonably well in a relatively large number of products, firms ...". The functional form we use represents learning as a progress function (Dutton and Thomas 1984), which builds on the general form of the learning curve described in Yelle (1979). Consistent with this literature on learning, we define every contributor i to have an individual learning productivity factor Π_i reflecting his individual learning ability, as a function of the number of projects executed:

$$\Pi_i = \Pi_0 \cdot (1 + n_i)^{L_i} \tag{1}$$

where:

$\Pi_0 =$ initial productivity factor for individual work
$n_i =$ total number of projects executed by contributor i

[7] There are M contributors who are successful together, so the total upwards adjustments add up to $M \bullet \Delta prob$. Simultaneously, there are $(N - M)$ others who see their probability adjusted down by $\Delta prob/(N/M - 1)$, so that the total downwards adjustments add up to $(N - M) \bullet \Delta prob/(N/M - 1) = (N - M) \bullet M \bullet \Delta prob/(N - M) = M \bullet \Delta prob$.

$L_i = \log_2 \lambda_i =$ characteristic 'learning index' for contributor i
$\lambda_i =$ individual 'learning rate'[8] for contributor i

This mathematical function is characteristic of the family of learning curves discussed by Yelle (1979). For illustration, given contributor $i = 1$, with a learning rate $\lambda = 0.5$ and an initial productivity factor $\Pi_0 = 1$, its individual learning productivity factor is $\Pi_1 = (1 + n_1)^{\log 2(0.5)} = (1 + n_1)^{-1} = 1/(1 + n_1)$. This function corresponds to a typical rectangular hyperbola translated by one unit: intersecting the ordinates axis at point (0,1), namely when $n_1 = 0$ then $\Pi_1 = 1$; and having a horizontal asymptote on the abscissa axis at $\Pi_1 = 0$. As contributor 1 executes projects, his associated individual learning productivity factor, Π_1, evolves over the path described by this learning curve function, thereby leading to better performance from project to project that the contributor has successfully executed, but never reaching the asymptote. In short, a smaller value of Π_1 implies better performance.

In addition, we introduce a joint learning productivity factor Π_{ij} reflecting the learning cumulated between i, j when working on projects together:

$$\Pi_{ij} = \Pi_{00} \cdot (1 + n_{ij})^{L_{ij}} \tag{2}$$

where:

$\Pi_{00} =$ initial productivity factor for collaborative work
$n_{ij} =$ number of projects executed together by contributors i, j
$L_{ij} = \log_2 \lambda_{ij} =$ characteristic learning index between contributors i, j
$\lambda_{ij} =$ joint learning rate between contributors i, j

For illustration, given contributors $i = 1$ and $j = 9$, with a joint learning rate $\lambda_{1,9} = 0.5$ and an initial productivity factor $\Pi_{00} = 1$, their joint learning productivity factor is $\Pi_{1,9} = (1 + n_{1,9})^{\log 2(0.5)} = (1 + n_{1,9})^{-1} = 1/(1 + n_{1,9})$. Similar to our previous example, this function corresponds to a typical rectangular hyperbola translated by one unit: intersecting the ordinates axis at point (0,1), namely when $n_{1,9} = 0$ then $\Pi_{1,9} = 1$; and having an asymptote at $\Pi_{1,9} = 0$. As contributors 1 and 9 execute projects, their associated joint learning productivity factor, $\Pi_{1,9}$ evolves over the path described by this hyperbolic function, thereby leading to better performance from project to project that is successfully executed, but never reaching the asymptotic value of 0. As in the previous example, a smaller value of $\Pi_{1,9}$ leads to better performance.

Let's now define T_{Ri}, a constant value representing the baseline performance of a typical project task performed by a typical contributor in role R_i. Based on this, we further define the actual individual productivity-adjusted performance for contributor i executing that particular project task as:

$$T_i = \Pi_i \cdot T_{Ri} = \Pi_0 \cdot (1 + n_i)^{L_i} \cdot T_{Ri} \tag{3}$$

This is, T_i represents the actual individual productivity-adjusted performance relative to the constant T_{Ri}. Note, for example, that in the case where this is the first project task (namely $n_i = 0$) executed by contributor i, and if for simplification we

[8] For a thorough discussion on "learning rates" and learning curves in general, please refer to Yelle (1979).

were to assume $\Pi_0 = 1$, we would have: $T_i = T_{Ri}$. Continuing with this example, as i accumulates experience through the successful execution of projects (namely $n_i > 0$), T_i follows a learning curve as defined by Eq. (1), amplified by the constant T_{Ri}.

Similarly, we define T_{ij} as the joint productivity-adjusted performance of a collaborative task performed by i with j:

$$T_{ij} = \Pi_{ij} \cdot X_{ij} \cdot T_{Ri} = \Pi_{00} \cdot (1 + n_{ij})^{L_{ij}} \cdot X_{ij} \cdot T_{Ri} \tag{4}$$

where X_{ij} is a constant, representing the symmetric 'dyadic affinity'[9] between i and j. From the perspective of agent i, the dyadic affinity constant X_{ij} multiplied by the individual baseline productivity constant T_{Ri}, yields the constant defining the baseline productivity of i in a collaborative task with j. This is, $T_{ij} \neq T_{ji}$. Note, as an example, that in the case where this is the first project task executed by contributors i and j (namely $n_{ij} = 0$), and if for simplification we were to assume $\Pi_{00} = 1$, we would have from the perspective of agent i: $T_{ij} = X_{ij} T_{Ri}$ (and from the perspective of j: $T_{ji} = X_{ij} T_{Rj}$). As i and j accumulate joint experience through the successful execution of common projects (namely $n_{ij} > 0$), T_{ij} follows a learning curve as defined by Eq. (2) amplified by the constant $X_{ij} T_{Ri}$. Similarly, T_{ji} follows a learning curve as defined by Eq. (2) amplified by the constant $X_{ij} T_{Rj}$. We assume that the accumulation of project task execution experience, individual and joint, is equivalent to the accumulation of knowledge, an asset stock in the sense of Dierickx and Cool (1989, p. 1506). As an individual agent executes independent project tasks—requiring no interaction with others—he develops experience reflected in Eq. (3). As an individual executes project tasks that require interaction with another agent, he develops experience reflected in Eq. (4).

4.3 Project performance outcomes

We define a project as always having (1) an individual task component and (2) a dyadic task component, to which the corresponding learning components—individual and dyadic—are associated. We then further assume (1) all individual learning associated to individual tasks to be independent of any dyadic learning component, and (2) all different dyadic learning components to also be independent from one another. These assumptions allow for a simple linear model of project performance. The first assumption implies that the learning that an individual i develops from working on independent project tasks does not transfer into that individual i being better able to work jointly with another individual. The second assumption implies that the joint learning that an individual i developed working jointly with another individual j does not translate into that individual i being better able to work jointly with yet another individual k.[10]

[9] The dyadic affinity relationships amongst i, j pairs are represented by a symmetric matrix $\mathbf{X}$ of all X_{ij} elements.

[10] We acknowledge that the independence assumptions introduce important limitations. For example, the second assumption rules out the possibility that T_{ij} could influence future dyadic interactions T_{ik}, where $j \neq k$. However, we believe the simple independence model represents performance at the project-level reasonably well, e.g., when i gains experience executing n projects with j, and then moves onto a new project where i should work with k for the first time, the learning gained from the prior n projects is reflected in the new project $n + 1$ thanks to T_i.

Based on the above independence assumptions, the performance associated to a project P involving M contributors is then defined by the sum of all the individual performances as given in (3), and all the joint performances among the contributors involved in the project as given in (4). This is:

$$T_P = \sum_{i \in P} T_i + \sum_{i \in P} \sum_{j \in P} T_{ij}, \quad i \neq j \tag{5}$$

This formula makes it explicit that the learning performance of a project depends both on the performance ability of each of the M individual contributors (first term in the equation), and on the joint performance ability of each pair of contributors (second term in the equation) who are in the team executing the project.

This has some implications. For example, a project involving one very inexperienced contributor, say $i = 1$ that has no experience, would lead to a lesser project performance *beyond* that reflected by the individual contributor's performance T_1 alone. The complete lack of experience of contributor $i = 1$ gets reflected also in his lack of joint experience working with other contributors j who are also participating in project P, through T_{1j} (where $j \neq 1$). Conversely, a project involving one very experienced contributor could lead to a better project performance *beyond* that reflected by the individual contributor's performance alone, whenever the experience of i comprises past experiences working with other members j currently in the team. When the individual i has significant overall experience, but no relevant experience involving other members j of the current team, then his otherwise significant experience would only be reflected in the individual performance factor (first term in Eq. (5)).

As projects get successfully executed, each contributor i develops a unique knowledge asset stock—individual and joint—as a result of his own independent experience and his joint interactions with various other contributors j. The total knowledge asset stock for i is therefore reflected by both n_i, which is knowledge developed and held exclusively individually (individual knowledge asset stock); and all of the n_{ij} ($j \neq i$), which constitute knowledge developed jointly when interacting with others and thereby held jointly with each of them (joint knowledge asset stock). Note that n_i accumulates for each and every successful project involving contributor i, whereas n_{ij} does not.

For a given project opportunity P, comprising M contributors we define its normalized performance:

$$\overline{Perf}_P = \frac{\sum_{i \in P} T_i + \sum_{i \in P} \sum_{j \in P} T_{ij}}{\sum_{i \in P} T_{Ri} + \sum_{i \in P} \sum_{j \in P} X_{ij} T_{Ri}}, \quad i \neq j \tag{6}$$

This is the value computed for each project outcome in the model.

4.4 Community formation

Individual contributors form into communities through their execution of projects. At first, there are N independent contributors and no communities. Independent contributors, as we refer to them in this article, do not belong to any community. Then, the simplest case of community formation is that resulting from the successful execution of a first project P_0, or *seed opportunity*, involving a group of M independent contributors i. These contributors become the first members of a new community C_0. This is:

$$i \in C_0, \quad \forall i \in P_0 \tag{7}$$

Our model further assumes that members of an already existing community, such as C_0 (assuming the seed opportunity has already taken place, and therefore Eq. (7) had already been applied to all i), will influence other independent contributors (unaffiliated newcomers) with whom they subsequently work, by virtue of their relative greater community experience and membership. This means that, following the successful execution of another project P_n involving independent contributors j and established members i of an existing community C_0, the up-to-then unaffiliated newcomers j become members of community C_0 as well:

$$j \in C_0, \quad \forall j \in P_n \tag{8}$$

As new project opportunities are executed, several disjoint communities form, namely $\{C_0, \ldots, C_x, \ldots\}$. Each community represents the social boundaries of a knowledge domain, with its own membership and experience paths (cf. Sect. 2, Definition). In our *baseline* model, membership in communities is strict in order to establish the existence of competing communities (cf. Table 2, Scenario 1). In this *baseline*, a stochastically generated new project opportunity involving a heterogeneous group of contributors—each already belonging to a different community—always fails to be executed, and hence does not affect membership composition in the communities facing the (failed) new project opportunity.

4.5 Competing communities

Competing communities C_x, C_y (where: $C_x \cap C_y = \varnothing$, $x \neq y$) arise from every new seed project opportunity involving M independent contributors. This leads to having several disjoint, co-existing communities each with different memberships and experience paths. The knowledge held by the community is that held by its members. To further define our baseline model, we assume that each community as a whole competes to gain *exclusive* experience and knowledge from executing projects, striving to develop its own practice before that of others. Hence, in our baseline model—void of free revealing and knowledge brokering—a project opportunity involving members of the same community can be successfully executed, while that involving a heterogeneous team of contributors cannot.

Recall that from a purely modeling perspective, new project opportunities are stochastically generated to encompass a group of M contributors (cf. Sect. 4.1). There are no deterministic constraints forcing these M contributors to belong to the same community when they are selected. Therefore, it is statistically possible that a new project opportunity could involve members of different communities. This is where conflict arises amongst strictly competing communities. For project opportunities involving more than one community, the default outcome of our baseline model is to forego the opportunity.

Foregoing the execution of new project opportunities entails foregoing the knowledge and experience that could otherwise have been gained by choosing to work on the project. However, not only is it theoretically the case that strictly competing communities may choose not to cooperate, it can also be argued that it is difficult for individual contributors from different experienced communities to effectively work on a project, given the significant differences in practice that may exist between communities. Both these reasons could make collaboration across communities less likely to occur, which is what our baseline model of strictly competing communities reflects.

4.6 Free revealing behavior

Free revealing behavior implies the sharing of codified knowledge by an individual contributor j through a knowledge commons. This knowledge then becomes accessible to all contributors in the system regardless of their community affiliation. It follows from this that all contributors can maintain their membership in their respective community, while benefiting from the cumulative knowledge stock shared by other contributors from any of the communities. To model free revealing of codified information we extend the individual learning model to account for the additional learning a given contributor i can derive from the shared knowledge commons. This is operationalized by modifying Eq. (1) as follows:

$$\begin{aligned}\Pi_{\substack{i\\ \mathrm{FR}}} &= \Pi_0 \cdot \left(n_i + \sum_{j, j\neq i} \beta_j \cdot n_j\right)^{L_i} \\ &= \Pi_0 \cdot \left(n_i + \sum_j (\beta_j \cdot n_j) - \beta_i \cdot n_i\right)^{L_i} \\ &= \Pi_0 \cdot \left((1-\beta_i)\cdot n_i + \sum_j \beta_j \cdot n_j\right)^{L_i}, \quad 1 \leq j \leq N \end{aligned} \tag{9}$$

where:

$\beta_j =$ degree of openness of contributor j towards sharing codified knowledge,

$$0 \leq \beta_j \ll 1$$

$n_i =$ number of individual projects executed by contributor i
$n_j =$ number of individual projects executed by contributor j

In the model set forth here, β_j represents the amount of codified knowledge that j contributed into the commons. It is expressed as the percentage of the individual knowledge gained by contributor j upon executing projects, which had been shared with the system at large. The sum of all this shared knowledge (i.e. the summation term in Eq. (9)), originally produced by each of the N contributors and eventually collectively accessible by all of them, constitutes what we refer to as the *knowledge commons*. The codified knowledge that is freely revealed outlasts any membership ties of a contributor with a particular community. Note that if we had $\beta = 0$ for each and all individual contributors in the model, then Eq. (9) would simply revert back to Eq. (1).

Earlier, we argued that one of the reasons preventing contributors from different communities from working together was their widely differing knowledge. Following the free revealing of codified knowledge into a commons, however, there is public awareness of the knowledge developed by each community. Thus, one may safely argue that this should diminish the knowledge gap and hence improve the likelihood of collaboration between members from disparate communities. This is when we can introduce knowledge brokering behavior in a systematic manner.

4.7 Knowledge brokering behavior

According to Hargadon (2002, p. 41), knowledge brokering behavior entails taking knowledge from different domains and linking it to new situations while building new social networks around the innovations hence created. Knowledge brokering requires "intensive interactions between individuals" (Hargadon 1998, p. 270). This is different from building a knowledge commons by assembling the individuals' freely revealed codified information. In our model, the knowledge broker is an individual member of one community—e.g., i, belonging to C_i—who brings his knowledge to execute a new project opportunity P_X involving members from another more dominant community (e.g. d belonging to C_d). C_d is assumed to be a community that is measurably more valuable than C_i, for knowledge brokering behavior to occur, consistent with preferential attachment theory (Barabási and Albert 1999), the preferential attachment policy in our model (cf. Sect. 4.1), and in line with the generally accepted economic assumption of utility maximization.

In practical terms, knowledge brokering takes place when i—facing a new project opportunity involving a member of another community d—joins the more dominant community C_d, thereby expanding the social boundary of the latter—in this case, membership of C_d increases. This behavior benefits i to the extent that he becomes part of a relatively more valuable community C_d. To operationalize this behavior at the level of the individual contributor we introduce an individual knowledge brokering threshold k, and use a measure of community value—namely, total community membership[11]—in the associated decision rule for the candidate broker i to join or not join a dominant community C_d involved in a new project opportunity. In this case the decision rule for the candidate knowledge broker i is based on the following ratio of community value, r, which is to be compared to the knowledge brokering threshold k:

$$r_{i,d} = \frac{\text{Membership}(C_i)}{\text{Membership}(C_{d,d\neq i})}, \quad i, d \in P_X, \tag{10}$$

where:

$C_i =$ the community to which the candidate knowledge broker i originally belongs,
$C_d =$ the dominant community present in the project ($C_d \neq C_i$), as represented by member d who belongs to C_d

[11] In addition to total community membership counts, we also explored total cumulated community knowledge. The results were similar. The intuition behind why these measures of influence yield similar results is because the greater the membership of a community, the larger the body of knowledge the community holds.

The threshold k is our proxy for the amount of knowledge brokering behavior displayed by an individual contributor. The decision rule based on Eq. (10) is then:

$$\begin{array}{ll} \text{if} & (r_{i,d} <= k) \\ \text{then} & i \text{ does incur in knowledge brokering behavior} \\ & \text{(hence executes the project opportunity)} \\ \text{else} & i \text{ does not incur in knowledge brokering behavior} \\ & \text{(hence gives up the project opportunity),} \end{array} \tag{11}$$

where:

$k =$ the knowledge brokering threshold (assumed to be a constant).

$$0 \leq k \leq 1$$

Knowledge brokering behavior occurs when the ratio $r_{i,d}$ is smaller than the threshold k. The relationship in Eq. (11) imposes that the relative value of C_d over C_i be at least $1/k$ times larger for i to incur in knowledge brokering behavior. For example, if $k = 0.1$, then C_d needs to be at least $1/0.1 = 10$ times bigger than C_i for i to incur in knowledge brokering behavior when faced with a project opportunity involving d. Another example, if $k = 0.5$, then C_d needs to be at least $1/0.5 = 2$ times bigger than C_i for i to incur in knowledge brokering behavior. At the extremes: if $k = 0$, then there is no knowledge brokering behavior by anyone, ever. If $k = 1$, then i will always incur in knowledge brokering behavior, regardless of the value of C_d.

Knowledge brokering activity (or lack thereof) has an impact on the success (or not) of new project opportunities P_X involving contributors from competing communities (e.g. i from C_i, d from C_d, etc. all involved in project P_X). On the one hand, a smaller threshold level k (i.e. closer to 0) limits the success of new project opportunities to a subset of projects with a substantial community value differential (when value of $C_d \gg$ value of C_i). For the individual knowledge broker i, this value differential can be interpreted as the gain that he obtains from working with the more valuable community. The cost to the individual knowledge broker i is the cumulated experience working with individuals from his previous community C_i that he cannot transfer to the new community C_d, since he has yet to develop project experience with all new individuals like d. On the other hand, a larger threshold level k (i.e. closer to 1) allows new project opportunities involving a more diverse pool of contributors belonging to a greater range of communities to be successfully executed.

Now that we have established the individual-level rules of behavior governing community membership, free revealing, and knowledge brokering, we can proceed to analyze how these contribute to overall system-level learning.

5 Experimental design and simulation results

Our experimental design comprises four model scenarios depicted in Table 2.

The baseline experiment, or Scenario 1, highlights agents implementing a basic learning model (see Eqs. (1), (2)) and forming into strictly competing communities (see Eqs. (6), (7)). This baseline serves as a benchmark with which to compare the scenarios introducing the independent variables of interest, namely free revealing

Table 2 Experimental design

Model scenarios	No free revealing	Free revealing (FR)
No knowledge brokering	• *Scenario 1* Baseline model of *competing communities* emerging from seed project opportunities, each evolving in an environment of shared purpose where individual contributors seek to build their own community	• *Scenario 2* Competing communities emerging, each evolving in an environment of shared purpose, with individual members incurring in *free revealing* of codified knowledge devoted to building a system-wide knowledge commons
Knowledge brokering (KB)	• *Scenario 3* Competing communities emerging and evolving in an environment of shared purpose, with individual contributors displaying *knowledge brokering* behavior	• *Scenario 4* Competing communities emerging and evolving in an environment of shared purpose, with individual contributors simultaneously incurring in *FR* and *KB* behavior

Notes: Four model scenarios exploring competing communities

(Scenario 2) and knowledge brokering (Scenario 3). Baseline Scenario 1 depicts a situation where from the onset individual contributors attempt to share in the creation and nurturing of a common community (see Eqs. (7), (8)). Yet, several competing communities emerge, as different subsets of independent contributors are presented with their first common project, referred earlier as seed project opportunity (cf. Sects. 4.4 and 4.5).

Model Scenario 2 builds on baseline Scenario 1 (expanding Eq. (1) into Eq. (9)). In Scenario 2, we introduce free revealing (whereby individual contributors share part of their knowledge with everyone else via a knowledge commons). The knowledge thusly shared by all contributors is assumed to be codified and cumulative in nature, and available to the system at large. The amount of knowledge shared by each individual depends on the free revealing parameter β (see Eq. (9)). Sticking to the strictly competitive nature of the baseline model, individual contributors from different communities forego opportunities to collaborate when facing a joint project opportunity (cf. Sect. 4.5).

Model Scenario 3 builds on baseline Scenario 1 and introduces knowledge brokering behavior (see Eq. (10) and decision rule (11)). In Scenario 3, knowledge brokers from different communities can each decide to collaborate on a project opportunity in spite of their knowledge domain differences. Such a decision depends on the knowledge brokering parameter k (see decision rule (11)). Finally, Model Scenario 4 explores the effects of simultaneously having free revealing and knowledge brokering behavior.

Informed by our study of the Netflix Prize challenge (cf. Sect. 3), we assume in our simulation that there is a perturbation to the system in the form of the introduction of an innovation (Taylor and Levitt 2007; Taylor et al. 2009). In our model, the innovation is the introduction of an organizational change involving communities, free revealing, and knowledge brokering. As we observed from the Netflix Prize data, there was a (unknown existing best) *pre-innovation performance* level Π_{pre} before the contest was launched, which was revealed through the initial wave of submissions in the first 90 days (cf. Sect. 3.3 and Fig. 1a). For practical purposes, in our simulation we assume Π_{pre} to be 1, and in our discussion we refer to it as the *reference performance threshold*. Similarly, we assume the (initial best) *post-innovation productivity* $\Pi_{post} = \Pi_0$ to be 1.5. Other simplifying assumptions we made, which do not affect the main findings of our study, are described in the next section along with the results.

We implemented the model described in Sect. 4 as an agent-based model written in the Python object-oriented scripting language, version 2.6.1. For each of the scenarios in our experimental design, we examine the patterns of simulation results for a sequence of project performance outcomes (cf. Eq. (6)). In each scenario, when the average system-level learning productivity improves to 1, we assume the system at large has benefited from the innovation or organizational change introduced (analogous to what was observed in the analysis of the Netflix Prize submissions data. Cf. Sect. 3.3 Fig. 1b). Note that while the model specification in Sect. 4 does not impose a limitation with regard to learning that could take place in parallel to a particular project, our simulation implementation assumes a sequential execution of project opportunities in each simulation run. A large number of simulation runs of each model setup unveil the patterns of results that are possible.

5.1 Results

The performance results we report for the nth project opportunity P_n, are the statistical average and variance of the normalized performance outcomes resulting from 1,000 simulation runs as measured by Eq. (6). The aggregate result for all projects unveils a distinct pattern of evolution of project performance outcomes. The discussion of the model's results focuses on the statistical patterns, mean and variance, observed from these simulations. This is analogous to the analysis that we developed for the Netflix Prize's project submissions data (cf. Sect. 3.3).

The results of 9,000 simulation runs, 1,000 for each combination of parameters, with $M = 4$ (cf. Sect. 2.1), are summarized in Fig. 3 (we also ran the simulations with $M = 2$, 4, and 8 and found similar patterns of results). Each plot summarizes the range of project performance outcomes observed after n project iterations ($n = 1, \ldots, 100$), with a *reference performance threshold* line defined at 1.0 for comparison purposes. The simulations address the four scenarios in our experimental design (Table 2). As in our empirical analysis of the Netflix Prize case data (cf. Sect. 3.3), we focus on the system-level average learning rate and learning rate volatility. These are the central dependent constructs in our theoretical arguments. We evaluate and compare each result according to three criteria: (1) *how soon* the average performance surpasses the reference threshold (after how many projects), (2) *the percentage* of performance outcomes above (or below) that threshold, and (3) the *best performance outcome* reached at the end of 100 projects.

Model Scenario 1, depicted in Fig. 3a, constitutes the *baseline* to which we compare all other model variations. This scenario examines strictly competing communities (namely, no free revealing and no knowledge brokering behaviors in individual

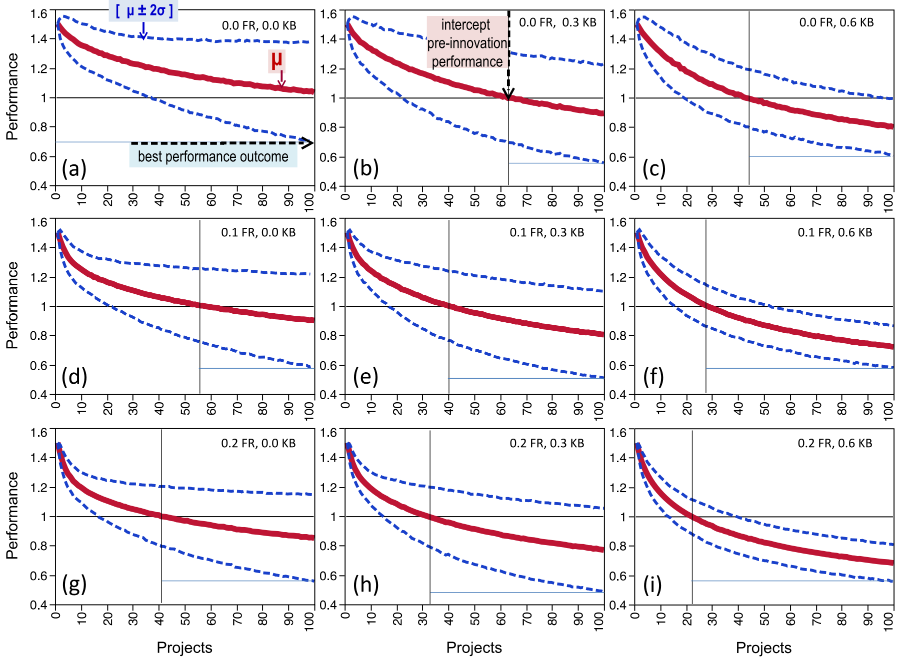


Fig. 3 Simulation results of system-level learning performance across competing communities. (**a**) Scenario 1: $\beta = 0, k = 0$; (**d**) Scenario 2.1: $\beta = 0.1, k = 0$; (**g**) Scenario 2.2: $\beta = 0.2, k = 0$; (**b**) Scenario 3.1: $\beta = 0, k = 0.3$; (**e**) Scenario 4.1: $\beta = 0.1, k = 0.3$; (**h**) Scenario 4.2: $\beta = 0.2, k = 0.3$; (**c**) Scenario 3.2: $\beta = 0, k = 0.6$; (**f**) Scenario 4.3: $\beta = 0.1, k = 0.6$; (**i**) Scenario 4.4: $\beta = 0.2, k = 0.6$. Parameter values used are informed by empirical studies and are fixed in the simulations. $N = 40$ contributors, $M = 4$ roles, $p_{initial} = 0.1$, $T_{Ri} = 1$, $X_{ij} = 0.2$, $\lambda = 0.8$ (typically between 0.7 and 0.9), $\Pi_0 = 1.5$ (initial post-innovation performance). Normalized pre-innovation performance is 1.0. There are 1,000 simulation runs per parameter combination. The interval between *dotted-lines* covers 95 % of all simulation outcomes

contributors) that are formed of individual contributors who take every *seed opportunity* (cf. Sect. 4.4) to create a common community. In spite of the individual contributors seeking to form a common community, a number of concurrent communities—unaware of each other—arise from independent seed opportunities. The strictly competing communities that emerge in this baseline case prevent project opportunities involving contributors from disjoint communities from being pursued. Given this scenario, the resulting *average learning curve for the system* does not cross the *reference performance threshold* even after 100 project iterations. On the one hand, over 50 % of the simulation outcomes end above the reference performance threshold. On the other hand, the volatility of the resulting paths is fairly large (Fig. 3a illustrates the spread of outcomes $[\mu - 2\sigma, \mu + 2\sigma]$ which is 0.67 units wide[12] on the vertical axis at the end of 100 project iterations). This latter result suggests that there is an important

[12]This 0.67 spread is used as a benchmark against which to compare the results in the subsequent model scenarios.

gap between the best and the worst performers when there is (a) no social interaction amongst the contributors, and (b) no knowledge exchange across communities.

Model Scenario 2 examines the effects of free revealing and is represented by Figs. 3d and 3g. These figures explore two levels of free revealing, 10 % and 20 % respectively, which is based on and consistent with observed estimates from the open source literature (Miller et al. 2006; Henkel 2006; Morrison et al. 2000). When free revealing behavior is introduced in the model, the *average learning curve for the system* not only consistently crosses the reference threshold at the end of 100 project iterations, but it does so after 56 and 41 projects, respectively. This is an important learning performance improvement over the results from model Scenario 1, where the average did not reach the threshold after 100 projects. Nonetheless, the volatility of the results remains quite high. For all levels of free revealing, the spread of outcomes $[\mu - 2\sigma, \mu + 2\sigma]$ is at least 0.6 units wide on the vertical axis at the end of 100 project iterations. Furthermore, a non-negligible 25 % (Fig. 3g) to 35 % (Fig. 3d) of project outcomes still end above the reference threshold. The spread of these results quantitatively show—in a competitive setting—that free revealing alone, while generally beneficial for everyone in terms of the average learning achieved, does little to reduce the gap between outperformers and underperformers.

Model Scenario 3 studies the effects of knowledge brokering as depicted in Figs. 3b (30 %) and 3c (60 %). Knowledge brokering has a positive effect on the average learning curve for the system, which surpasses the reference threshold after 63 and 43 projects, respectively. However, the most important effect is on the volatility of the learning curve outcomes, which is significantly reduced as a result of increasing knowledge brokering. For knowledge brokering at 60 %, the spread of outcomes $[\mu - 2\sigma, \mu + 2\sigma]$ is 0.4 units wide on the vertical axis at the end of 100 project iterations. Furthermore, all simulation outcomes consistently end below the reference threshold (Fig. 3c). Hence, as most contributors share the system-level benefits that ensue from knowledge brokering, it appears as a desirable behavior to promote across communities. In turn, this would suggest that weaker competition amongst communities may be more desirable when the goal is that everyone achieve similarly good results (through knowledge spill-overs resulting from knowledge brokering activity).

Model Scenario 4 explores the effects of combining free revealing and knowledge brokering at different levels (Figs. 3e, 3f, 3h, and 3i). First, the average learning rate improves for all combinations of free revealing and knowledge brokering (compared to all previous scenarios). Second, the conclusions regarding the volatility of the performance outcomes remain the same as in previous scenarios. On the one hand, this means that having higher levels of free revealing, while leaving knowledge brokering unchanged, leads to a negligible change in volatility (e.g., contrast Figs. 3e and 3h). On the other hand, increasing knowledge brokering, while leaving free revealing unchanged, leads to a significant decrease in volatility (e.g., contrast Figs. 3e and 3f). Third, for every level of knowledge brokering, higher levels of free revealing consistently lead to superior best performance outcomes (e.g., compare the lower bound, $\mu - 2\sigma$, across Figs. 3b, 3e and 3h).

Considering the nine simulation results (Fig. 3) of the four scenarios, it becomes apparent that the best performance outcome attainable at the end of 100 project iterations follows an inflexion point as knowledge brokering increases for each level

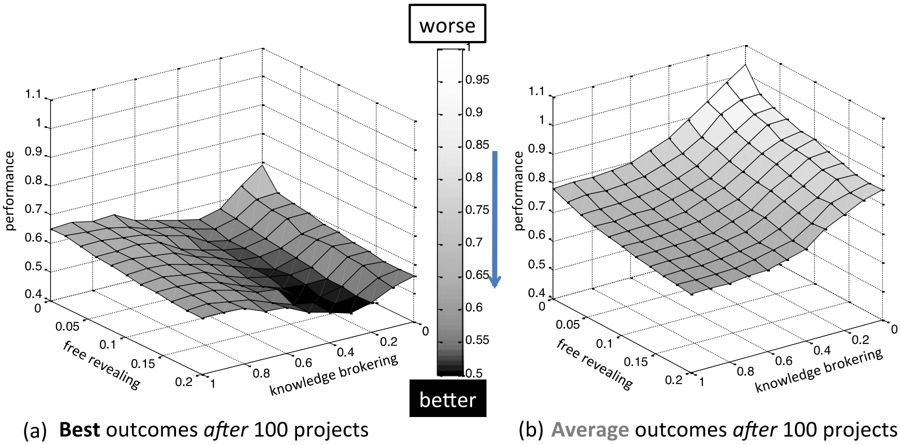


Fig. 4 An important trade-off is unveiled when combining different levels of knowledge brokering and free revealing: to achieve the best average performance for the system vs. to achieve the best overall performance outcome from the system

of free revealing (e.g. compare the best performance outcomes for the 100th project in Figs. 3d, 3e and 3f). For any given level of free revealing, initially an increase in knowledge brokering leads to a superior best performance outcome (compare Figs. 3d and 3e); later, a further increase in knowledge brokering leads to an inferior best performance outcome (compare Figs. 3e and 3f). This shows that the reduction in volatility that accompanies higher levels of knowledge brokering ultimately reduces the likelihood of reaching superior best performance outcomes. This last observation suggests that there is a tradeoff between reaching better average system performance and reaching best overall performance outcomes from the system.

To explore this phenomenon in greater detail, we ran an additional series of simulations to plot—for all values of free revealing (between 0 and 20 %) and all values of knowledge brokering (between 0 and 100 %)—*both* (a) the best performance outcome and (b) the average performance outcome attained at the 100th project iteration. The plot of results for best performance outcomes (Fig. 4a) show that—for each level of free revealing—increasing levels of knowledge brokering initially enhance the best possible performance outcomes attainable; followed by a tipping point after which an increase in knowledge brokering actually reduces the best performance outcomes that can be achieved. The plot of results for the average performance outcomes (Fig. 4b) show how both free revealing and knowledge brokering always contribute to achieve better average performance outcomes for the system.

6 Analysis and discussion

The experimental design (cf. Table 2) guiding the results discussed here first introduces a *baseline* model scenario where individual contributors are presented with new project opportunities (as described in Sect. 4.1) while they attempt to grow a common community. In spite of this, several disjoint communities emerge. In this context, without knowledge brokering behavior and without free revealing behavior, members of one community default to non-cooperation with members of another. The simulation results for this baseline model scenario could well represent learning performance across a traditional setting of strictly competing firms: namely, individual members of one firm taking on new project opportunities involving members of the same firm, while dismissing new project opportunities that, to be executed, would require the involvement of individuals from other strictly competing firms.

While reality may be more nuanced than our baseline model scenario, this baseline ensures that we have a clear benchmark against which we can compare the relative effects of free revealing and knowledge brokering behavior. We observe that each of these two behaviors have clearly distinct effects from those of contributors in a strictly competitive setting. The most relevant results from our simulation scenarios of competing communities, summarized in Figs. 3 and 4, show that, *at the system-level*:

Proposition 1 *Free revealing behavior*

(a) *has a positive effect on the average learning rate*, *while*
(b) *having little if any effect on the volatility of the learning rate*.

Proposition 2 *Knowledge brokering behavior*

(a) *reduces learning rate volatility*, *and*
(b) *has a* (*moderately*) *positive effect on the average learning rate*.

Proposition 3 *There is a trade-off involving knowledge brokering and free revealing behaviors*, *whereby*:

(a) *knowledge brokering initially enhances the best possible performance outcome attainable by the system*, *and later penalizes it*, *and*
(b) *free revealing positively moderates the effect of knowledge brokering leading to better outcomes in general*, *both for average and best performance*.

Proposition 1(a) is in line with theoretical arguments on the benefits of free revealing. von Hippel and von Krogh (2006) argue that free revealing of detailed product information, etc., can make good economic sense for innovators and society alike, since users can not only learn, but also improve upon what is revealed. Henkel argues that selective revealing offers business firms "considerable potential for efficiency gains" (Henkel 2006, p. 967), by reducing the duplication of effort and avoiding transaction costs of commercial licensing. Our model of free revealing implements the partial sharing of codified knowledge of individual project contributors (cf. Eq. (9)), and the simulation results explicitly show the efficiency gains that are possible at varying levels of revealing (contrast Figs. 3d and 3g relative to Fig. 3a). Through our simulation results, we observe that introducing free revealing (e.g. consider Fig. 3d with free revealing at 10 % relative to Fig. 3a without revealing) substantially reduces the number of project iterations required for the system-level average performance to reach the *reference performance threshold*. As the literature conceptually predicts (von Hippel and von Krogh 2006), our model quantitatively shows how free revealing is good (i) for individuals (innovators) who learn and build upon what is revealed, and (ii) for the overall system (society) that benefits from system-level learning efficiencies.

Furthermore, Proposition 1(b) offers an additional insight not found in empirical investigations of free revealing, nor in the extant literature in open source (to the best of our knowledge). Proposition 1(b) implies that free revealing, by virtue of not significantly affecting learning rate volatility, does little to reduce the gap between outperformers and underperformers (compare Figs. 3c and 3b with Fig. 3a). Associated with this gap is the existence of far reaching outperformers—typically from a dominant community, our model suggests—who benefit from the free-revealing of all others—particularly from communities that are not dominant. Empirically, our survey of Netflix Prize participants (cf. Sect. 3.2) shows that 38 % of respondents admitted to having incorporated software code from others without having themselves open sourced any code. In our simulation results, the bulk of contributors who underperform are typically from non-dominant communities. In fact, the volatility of outcomes observed results from situations of non-cooperation leading to failure to execute a project opportunity—which is more likely to occur among communities that are valued similarly.

The results summarized in Proposition 2(a) and 2(b) are generally in line with the qualitative findings of Hargadon (1997, 1998, 2002) in that knowledge brokering has a positive effect on average performance. More importantly, however, our model brings a more nuanced understanding of the nature of its effects. Our simulation results show that the main effect of knowledge brokering is in reducing the volatility of learning performance, as opposed to achieving faster learning on average. Indeed, when contrasting the results of Figs. 3e and 3f, we observe that although the level of knowledge brokering is increased substantially, the average learning curve for the system improves only marginally. At the same time, the volatility of the learning curves diminishes significantly, which is a new insight these results contribute to the literature on knowledge brokering.

Propositions 3(a) and 3(b) result from our study of an identified phenomenon involving Propositions 1 and 2 in achieving desirable performance outcomes for the system, either focusing on achieving (i) best overall performance outcomes or (ii) good overall average performance. Our results show that different combinations of free revealing and knowledge brokering affect the performance outcomes attainable by the system in important ways. Figure 4 depicts the effects of various levels of free revealing and knowledge brokering on both (i) best system-level performance (Fig. 4a) and (ii) average system-level performance (Fig. 4b). Interestingly, the results of our simulations show that there is a tipping point after which an increase in knowledge brokering behavior is worse from an innovation performance standpoint—where the focus is in achieving the best overall performance possible. This knowledge-brokering inflection point is consistent with evidence from patent citations found by Hsu and Lim (2011). In addition, for any level of knowledge brokering behavior, higher levels of free revealing only mildly contribute to reducing the gap between (i) best possible system-level performance and (ii) average system-level performance.

The results presented in Fig. 4a, corresponding to Proposition 3(a), are more intuitive if we consider the properties of the model underlying the simulation. In the extreme theoretical case of no knowledge brokering ($KB = 0$), all cross-community project opportunities fail to be executed and learning is therefore limited to successful intra-community project opportunities only. Given limited learning opportunities, the best performance that can be achieved after a finite number of projects remains similarly limited, hence poor on average. As some knowledge brokering is introduced ($0 < KB \ll 1$), a number of cross-community projects become viable and are successfully executed. Additional learning materializes through the projects involving the individual knowledge brokers. In turn, these successful projects contribute to the growth of dominant communities, which—by virtue of being successful—concentrate more learning opportunities than other less dominant communities. Better project performance can be achieved amongst successful individuals in dominant communities than amongst less successful individuals in less dominant communities. Beyond a certain point (as KB gets closer to 1), however, too widespread knowledge brokering behavior has the effect of making cross-community projects no different from intra-community projects. While there are additional learning opportunities, knowledge is increasingly distributed across the general population of contributors. All project performance outcomes *converge* towards the system-level average.

The above propositions contribute new insights to the literature in open source (von Hippel and von Krogh 2003) and knowledge brokering (Hargadon 1997, 1998, 2002; Hsu and Lim 2011), which have thus far evolved mostly independently from each other. In particular, each research stream has independently shown that higher levels of free revealing on the one hand and knowledge brokering on the other hand, would positively contribute to enhance learning performance, on average. By using a simulation approach our research simultaneously explored the effects of free revealing and knowledge brokering on *both* the average and the volatility of learning performance outcomes. Finally, the findings of this research extend previous results and offer new insights into the tradeoffs involved in combining free revealing with knowledge brokering behavior to achieve best overall performance outcomes across competing communities, a matter of fundamental interest in the pursuit of innovation at the system-level (society).

6.1 Model validity

There is agreement, for validation purposes, that computational models of social systems be rooted on established theoretical grounds and that they can be compared against real life data or other models (Burton 2003; Carley 1996; Davis et al. 2007). The model developed in this article is based on established learning curve theory (Argote and Ingram 2000; Wright 1936; Yelle 1979), situated learning theory (Brown and Duguid 1991; Lave and Wenger 1991), and the literature on open source (von Hippel and von Krogh 2003; Harhoff et al. 2003; Henkel 2006), which addresses Burton's notion of *informal docking*, a first step toward model validation (Burton 2003, p. 102). The model's independent parameter values used are informed by empirical studies of open source communities (Henkel 2006; Krishnamurthy 2002, 2003), and the performance outcomes produced by the model reproduce patterns of learning performance found in case data from the Netflix Prize competition. Hence, the simulation has face validity. These arguments, linking our model to empirically observed data, further contribute towards achieving internal validity.

The model also follows Handley et al. (2006, p. 64), who argue that "individual learning (in communities) should be thought of as *emergent*, involving *opportunities to participate* in the *practices* of the community as well as the development of an *identity* which provides a sense of belonging and commitment". Furthermore, the model is in line with the empirical findings of Harhoff et al. (2003) and Henkel (2006) in that individuals and the overall system simultaneously benefit from free revealing. Our simulation results further show that, even when contributors from competing communities forego opportunities to cooperate, free revealing behavior has a strong positive effect on average system-level learning performance. Our model's results on learning brought about by knowledge brokering behavior resonate with empirical results found in the literature (Hargadon 1997, 1998, 2002; Hsu and Lim 2011). Finally, also consistent with our findings, Miller et al. (2006)—building upon March's (1991) work—find that "as distant learning becomes more common, population-wide diversity dissipates quickly" (Miller et al. 2006, p. 714). Knowledge brokering and free revealing enable distant learning across communities, dissipating diversity.

6.2 Limitations

The model presented in this article strives for simplicity and generality (Carley 1996; Weick 1979, 1995). Simplicity allows us to study the modeled constructs more carefully and derive clear propositions that inform our understanding of the phenomena modeled. Consequently, our model results offer a quantitative and qualitative understanding of the dynamic interplay of free revealing and knowledge brokering behavior in the context of emergent competing communities.

First, in our model of open source (see Eq. (9)), the β parameter representing free revealing is a coefficient that indicates how much knowledge each individual contributor shares with everyone else. On one hand, this parameter does not accurately reflect the 'selective revealing' concept discussed in the open source literature (Henkel 2006), which entails some "selection" criterion for what to reveal and what to keep private. On the other hand, the *partial* revealing implemented by this parameter in our model illustrates the middle-ground between private knowledge held by the individual contributor, and collective knowledge shared with the community at large (von Hippel and von Krogh 2006).

Second, in our model of knowledge brokering behavior, the decision parameter k (see Eq. (10)) used to determine whether to collaborate or not on a project depends only on the size of the community. Whenever contributors from disjoint communities are presented with an opportunity involving a highly dominant community, they resort to collaboration and thus contribute to the growth of the dominant community. This is consistent with models of preferential attachment, leading to empirically observed power-law distributions of membership (Barabási and Albert 1999; Clauset et al. 2007). Another possibility we tried in Sect. 4.7 with similar results was to consider a decision parameter based on the relative experience of the individual contributors facing the opportunity. A more accurate model would account for other individual-level differences, such as incompatibility of resources available or incongruity of principles and beliefs, on the decision to collaborate.

Third, the granularity of our model involves homogeneous and short-lived project opportunities providing the same amount of learning each time. Successful projects are considered equally in terms of the additional experience they provide to their contributors, without accounting for differences among projects or contributors. In the Netflix Prize case used to inform our model (cf. Sect. 3), submissions consistently address the same problem using the same data and it seems reasonable to consider such projects as homogeneous. Research on online social relationships shows that these social ties are typically weak (Cummings et al. 2002). And there is evidence that online projects are relatively short in duration (Krishnamurthy 2002). Hence, we found it acceptable to assume project collaborations that were short-lived and not substantially different on average. While it is possible to introduce additional heterogeneity into our model, we purposely chose to keep our model simple to avoid deviations resulting from added complexity.

7 Conclusions and implications

Drawing on a study of the Netflix Prize challenge we developed a formal model articulating the effects of combining different levels of free revealing and knowledge brokering on the learning performance of competing communities. We find that ever-increasing levels of free revealing do not imply optimal learning outcomes for all participants in the system (i.e. this does not lead to achieving better learning rates for all), nor do ever-increasing levels of knowledge brokering lead to achieving the best possible outcomes from the system (i.e. this does not automatically lead to achieving the best possible solutions). Rather, combining different levels of free revealing and knowledge brokering lead to different system-level learning patterns, which unveil important tradeoffs for innovation.

Our main findings show that, at the system level: (1) free revealing has a positive effect on the average learning rate, while having a negligible impact on learning rate volatility, (2) knowledge brokering reduces learning rate volatility, while having a moderately positive effect on the average learning rate. In addition, when looking at the best performance outcomes achieved by the system, we find that (3) knowledge brokering has an inverted-U effect on performance where higher levels of knowledge brokering initially enhance learning performance, but beyond a certain level penalize it. *Thus, there is an important tradeoff regarding the relative amounts of free revealing and knowledge brokering that yield the best learning performance for either a few individual contributors or for the system at large* (see Figs. 3 and 4). Taken separately, free revealing benefits some individual outperformers the most (relatively high volatility of outcomes), while knowledge brokering makes the system at large consistently more efficient (relatively low volatility of outcomes). Taken together, an appropriate combination of free revealing and—particularly—knowledge brokering can help achieve best overall learning performance outcomes from the system, hence leading to more effective innovation.

These findings provide managers who are strategically considering running an open competition with new insights regarding the importance of managing the knowledge brokering process, as well as details about how different types and levels of knowledge sharing are likely to affect the innovation outcomes from the system. Managers should specifically consider the usefulness of combining (or not): (1) the implementation of *free revealing*, or open source knowledge repositories, as a means to accelerate average learning for all, while keeping high volatility of learning outcomes, and (2) the risk mitigating effects of *knowledge brokering* facilitation, to embrace and consolidate learning from across communities, effectively reducing the volatility of learning outcomes. The informed manager should be aware of (3) the tradeoff of systematically supporting the crossing of knowledge boundaries which would lead to a tipping point, after which: an increase in knowledge brokering behavior is worse from an innovation performance standpoint, effectively yielding sub-optimal learning outcomes (see Fig. 4a). Hence, if the goal is to achieve the best possible learning performance outcome from the competitive system (as in the Netflix Prize challenge), then knowledge brokering needs to be actively managed.

The implications of this research go beyond that of the formal setting of the specific challenge discussed in this paper—even beyond the conscious implementation of free revealing or knowledge brokering. In the Internet-connected world we live in today, it is increasingly common for firms and individuals to have an online presence where valuable codified information is *freely revealed* online. This freely revealed codified knowledge quickly becomes accessible to everyone, customers and competitors alike, a process that is facilitated by search engines (e.g. Google, Bing). The Internet itself is the world's largest repository of codified knowledge as well as an increasingly important vehicle for mass interaction (e.g. Facebook, Twitter). Individuals and firms are actively searching for the most current codified knowledge—in other domains or from other communities—that could potentially be incorporated into their own, essentially incurring in *knowledge brokering* behavior with increasing intensity. Competing online communities are particularly important to an emergent category of 'online distributed organizations'[13] (Villarroel 2011; Villarroel and Gorbatai 2011), that rely heavily on the new knowledge generated from the managed interactions across these communities. These new organizations would benefit from developing mechanisms to actively manage the levels of knowledge brokering activity across the communities they work with to achieve the desired innovation outcomes.

Finally, the comparison of the four model scenarios and the three induced propositions outlined in this paper contribute to the theoretical conversation on organizational learning in general, and on competing communities in particular. We identified the existence of learning performance trade-offs in competing communities by exploring and comparing the effects of varying levels of free revealing and knowledge brokering behavior. With growing numbers of crowdsourcing initiatives, self-organized and firm-sponsored, our findings should offer valuable insights for managers involved in the orchestration of the innovation activities of the associated online communities. Future research should explore the comparative impact on learning performance of sources of heterogeneity such as differences in problem complexity and individual motivations, and types of competing communities (e.g., public and private), to name some. The results of such comparative research should expand our understanding of learning performance across competing communities, and contribute to the effective organization of online distributed innovation (Villarroel 2011).

[13] Firms whose core business is built upon a crowdsourcing platform.

Acknowledgements This research was first distinguished with a Best Student Paper Award from the North American Association for Computational Social and Organization Science (NAACSOS) in 2007 (http://www.casos.cs.cmu.edu/naacsos/awards.php). Further development of this work benefited from fellowships of the National Science Foundation of Switzerland (PBELP2-123027) and the MIT Sloan International Faculty Fellows program. The article incorporated helpful comments received at the Academy of Management Annual Meeting in 2008 and the Strategic Management Society 30th International Annual Meeting in 2010, where this research was presented. In particular, the authors would like to thank colleagues in the modeling community, Michael Prietula, David Sallach, and John Sterman, for their valuable feedback. The conceptual contribution of this work received constructive comments from Dietmar Harhoff, Eric von Hippel, Karim Lakhani, and Joel West, whom we thank dearly. Not the least, the authors extend their special appreciation to the editors of CMOT and the anonymous reviewers who contributed to improving the quality of this piece.

References

Ahuja G, Lampert C (2001) Entrepreneurship in the large corporation: a longitudinal study of how established firms create technological breakthroughs. Strateg Manag J 22:521–543

Argote L, Ingram P (2000) Knowledge transfer: a basis for competitive advantage in firms. Organ Behav Hum Decis Process 82(1):150–169

Austin RD (2004) Novell: open source software strategy. Case study. Harvard Business School Publishing, Boston. Prod. Nr. 9-605-009

Bagley CE, Lane D (2006) Black duck software. Case study. Harvard Business School Publishing, Boston, pp 9–806–9-121

Balka K, Raasch C, Herstatt C (2009) Open source beyond software: an empirical investigation of the open design phenomenon, R&D Management Conference

Barabási AL, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512

Bell RM, Koren Y (2007) Lessons from the Netflix Prize. ACM SIGKDD Explor Newsl 9(2):75–79. Special issue on visual analytics

Bell RM, Koren Y, Volinsky C (2007) The BellKor solution to the Netflix Prize. In: AT&T Labs–Research. http://www.research.att.com/~volinsky/netflix/ProgressPrize2007BellKorSolution.pdf

Bennett J, Lanning S (2007) Netflix Prize. In: Proceedings of KDD Cup and workshop 2007, August 12, 2007, San Jose, California, USA. Available at http://brettb.net/project/papers/2007%20The%20Netflix%20Prize.pdf

Brown JS, Duguid P (1991) Organizational learning and communities-of-practice: toward a unified view of working, learning, and innovation. Organ Sci 2:40–57

Brown SL, Eisenhardt KM (1997) The art of continuous change: linking complexity theory and time-paced evolution in relentlessly shifting organizations. Adm Sci Q 42:1–34

Burton RM (2003) Computational laboratories for organization science: questions, validity and docking. Comput Math Organ Theory 9(2):91–108

Carley K (1996) Validating computational models. Working paper, Carnegie Mellon University

Clauset A, Rohilla C, Newman M (2007) Power-law distributions in empirical data. Working paper, submitted Rev Mod Phys, June 7, 2007

Cummings JN, Butler B, Kraut R (2002) The quality of online social relationships. Commun ACM 45(7):103–108. doi:10.1145/514236.514242

Davis JP, Eisenhardt KM, Bingham CB (2007) Developing theory through simulation methods. Acad Manag Rev 32(2):480–499

Dierickx I, Cool I (1989) Asset stock accumulation and sustainability of competitive advantage. Manag Sci 35(12):1504–1511

Dutton JM, Thomas A (1984) Treating progress functions as a managerial opportunity. Acad Manag Rev 9(2):235–247

Fauchart E, Von Hippel E (2008) Norms-based intellectual property systems: the case of French chefs. Organ Sci 19(2):187–201

Franke N, Shah S (2003) How communities support innovative activities: an exploration of assistance and sharing among end-users. Res Policy 32:157–178

Giles J (2005) Internet encyclopaedias go head to head. Nature 438:900–901. Special Report. doi:10.1038/438900a

Handley K, Sturdy A, Fincham R, Clark T (2006) Within and beyond communities of practice: making sense of learning through participation, identity and practice. J Manag Stud 43:3
Hargadon A (1997) Technology brokering and innovation in a product development firm. Adm Sci Q 42:716–749
Hargadon A (1998) Firms as knowledge brokers: lessons in pursuing continuous innovation. Calif Manag Rev 40:209–227
Hargadon A (2002) Brokering knowledge: linking learning and innovation. Res Organ Behav 24:41–85
Harhoff D, Henkel J, von Hippel E (2003) Profiting from voluntary information spillovers: how users benefit by freely revealing their innovations. Res Policy 32:1753–1769
Henkel J (2006) Selective revealing in open innovation processes: the case of embedded Linux. Res Policy 35(7):953–969
Howe J (2009) Crowdsourcing: why the power of the crowd is driving the future of business. Crown Publishing Group, New York
Hsu D, Lim K (2011) The antecedents and innovation consequences of organizational knowledge brokering. Working paper, Wharton School and Melbourne Business School
Jeppesen LB, Lakhani KR (2010) Marginality and problem solving effectiveness in broadcast search. Organ Sci. Published online before print Feb 1, 2010. doi:10.1287/orsc.1090.0491
Knorr Cetina K (1999) Epistemic cultures: how the sciences make knowledge. Harvard University Press, Cambridge
Krishnamurthy S (2002) Cave or community? An empirical examination of 100 mature open source projects. Working paper
Krishnamurthy S (2003) A managerial overview of open source software. Business horizons. Indiana University Kelley School of Business
Lakhani KR, von Hippel E (2003) How open source software works: ‘free’ user-to-user assistance. Res Policy 32:923–943
Lave J, Wenger EC (1991) Situated learning: legitimate peripheral participation. Cambridge University Press, Cambridge
Lazaric N, Lorenz E (1998) Introduction: the learning dynamics of trust, reputation and confidence. In: Lazaric N, Lorenz E (eds) Trust and economic learning. Edward Elgar, Cheltenham, p 3
Lee GK, Cole RE (2003) From a firm-based to a community-based model of knowledge creation. Organ Sci 14(6):633–649
Levitt B, March JG (1988) Organizational learning. Annu Rev Sociol 14:319–338
MacCormack A (2002) Red Hat and the Linux revolution. Case study. Harvard Business School Publishing, Boston. Prof. Nr. 9-600-009
March JG (1991) Exploration and exploitation in organizational learning. Organ Sci 2(1):71–87
Miller KD, Zhao M, Calatone RJ (2006) Adding interpersonal learning and tacit knowledge to March’s exploration-exploitation model. Acad Manag J 49(4):709–722
Morrison PD, Roberts JH, von Hippel E (2000) Determinants of user innovation and innovation sharing in a local market. Manag Sci 46(12):1513–1527
Murray F, O’Mahony S (2007) Exploring the foundations of cumulative innovation: implications for organization science. Organ Sci 18(6):1006–1021
Mutch A (2003) Communities of practice and habitus: a critique. Organ Stud 24(3):383–401
Netflix (2007) $50,000 progress prize is awarded on first anniversary of $1 million Netflix Prize. Netflix press release. Nov. 13, 2007. http://www.netflix.com/MediaCenter?id=6422
Netflix Knowledge Brokering (2008) “When Gravity and Dinosaurs unite”. Teams Gravity and Dinosaurs join forces to reach the first place on the Netflix Prize Leaderboard with an 8.7 % improvement on February 12, 2008. http://www.netflixprize.com/leaderboard. Accessed 13 Feb 2008
Netflix Free Revealing (2007) Forum thread in general discussion category entitled “Free source code :)”. http://www.netflixprize.com/community/viewtopic.php?pid=3897#p3897. First accessed on January 3, 2007. Last accessed on June 30, 2011
Netflix Prize Community (2008) http://www.netflixprize.com/community/. Accessed 13 Feb 2008
Netflix Prize Rules (2006) http://www.netflixprize.com/rules. Accessed 2 Oct 2006
O’Mahony S, Cela Diaz F, Mamas E (2005) IBM and eclipse. Case study. Harvard Business School Publishing, Boston. Prod. Nr. 9-906-007
Pénin J (2011) Open source innovation: towards a generalization of the open source model beyond software, *DIME* Final Conference, 6–8 April, Masstricht
Roberts J (2006) Limits to communities of practice. J Manag Stud 43(3):623–639
Rosenkopf L, Almeida P (2003) Overcoming local search through alliances and mobility. Manag Sci 49(6):751–766

Schwen TM, Hara N (2003) Community of practice: a metaphor for online design? Inf Soc 19(3):257–270
Stam W (2009) When does community participation enhance the performance of open source software companies? Res Policy 38:1288–1299
Taylor J, Levitt R (2007) Innovation alignment and project network dynamics: an integrative model for change. Proj Manag J 38(3):22–35
Taylor J, Levitt R, Villarroel JA (2009) Simulating learning dynamics in project networks. J Constr Eng Manag 135(10):1009–1015
Terwiesch C, Xu Y (2008) Innovation contests, open innovation, and multiagent problem solving. Manag Sci 54(9):1529–1543
van der Waal S (2010) Creating a project on SourceForge. OSS watch, February 2, 2010. http://www.oss-watch.ac.uk/resources/sourceforge.xml. Accessed 23 Feb 2010
Villarroel JA (2008) Open Source Corporate Strategy (OSCS): unveiling the firm's open sources of competitive advantage. Doctoral dissertation No. 4173, Swiss Federal Institute of Technology. Available on: http://library.epfl.ch/theses/?nr=4173
Villarroel JA (2011) Strategic crowdsourcing: the emergence of online distributed innovation. In: Huff A, Reichwald R, Moeslein K (eds) Leading open innovation. Forthcoming with MIT Press, Cambridge
Villarroel JA, Gorbatai A (eds) (2011) Online distributed organization. Academy of management symposium, with Malone T, Lakhani K, Dahlander L, Tucci C, San Antonio, USA, August 16, 2011
von Hippel E, von Krogh G (2003) Open source software and the 'private-collective' innovation model: issues for organization science. Organ Sci 14(2):209–223
von Hippel E, von Krogh G (2006) Free revealing and the private-collective model for innovation incentives. R & D Manag 36(3):295–306
Weick KE (1979) The social psychology of organizing, 2nd edn. Addison-Wesley, Reading
Weick KE (1995) What theory is not, theorizing is. Adm Sci Q 40(3):385–390
Wenger EC, McDermott R, Snyder WM (2002) Cultivating communities of practice: a guide to managing knowledge. Harvard Business School Press, Boston
West J, Lakhani KR (2008) Getting clear about communities in open innovation. Ind Innov 15(2):223–231
Wright TP (1936) Factors affecting the cost of airplanes. J Aeronaut Sci 3:122–128
Yanow D (2004) Translating local knowledge at organizational peripheries. Br J Manag 15:S9–S25
Yelle LE (1979) The learning curve: historical review and comprehensive survey. Decis Sci 10:302–328

J. Andrei Villarroel is Assistant Professor at Católica-Lisbon School of Business and Economics (CLSBE), Research Affiliate at Massachusetts Institute of Technology (MIT) Center for Collective Intelligence, and Visiting Faculty in Organizational Dynamics at the University of Pennsylvania (UPenn). Prior to this, Prof. Villarroel held appointments at MIT Sloan School of Management as International Faculty Fellow, Visiting Scholar, and as Postdoctoral Fellow in Technological Innovation and Entrepreneurship. His research focuses on the effective organization of open and distributed approaches to work and innovation beyond the formal boundaries of the firm, also referred to as 'strategic crowdsourcing'. His research was featured at the Academy of Management, Strategic Management Society. He was also recognized with two consecutive Best Paper Awards from the North American Association for Computational Social and Organizational Science (NAACSOS). He holds a Ph.D. in management of technology and entrepreneurship from the Swiss Federal Institute of Technology, and a master of science from Carnegie Mellon University (CMU).

John E. Taylor is Associate Professor at Virginia Tech. Prior to this appointment, he was Assistant Professor and Director of the Project Network Dynamics Lab in the School of Engineering and Applied Science at Columbia University. Dr. Taylor's research combines experimental research and agent-based simulation to examine fundamental change processes at the intersection between human networks and engineered networks. Dr. Taylor received his Ph.D. in 2006 from Stanford University on the topic of innovation in inter-organizational networks and his research continues to explore the network dynamics that follow system level network perturbations. To date he has focused on network dynamics associated with information system integration, industry globalization, workforce virtualization, and energy conservation in buildings. His research is funded by the National Science Foundation, the Earth Institute, the Alfred P. Sloan Foundation, and other public and private funding sources. Dr. Taylor currently holds a 2009–2011 Alfred P. Sloan Foundation Industry Studies Fellowship.

Christopher L. Tucci is Professor of Management of Technology at the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland, where he holds the Chair in Corporate Strategy & Innovation. He received

the Ph.D. in Management from the Sloan School of Management at the Massachusetts Institute of Technology. Dr. Tucci's primary area of interest is in technological change and how waves of technological changes affect incumbent firms. He is also studying how the technological changes brought about by the popularization of the Internet affect firms in different industries. He is the co-author of the books Nurturing Science-Based Ventures and Internet Business Models and Strategies, and has published articles in, among others, Strategic Management Journal, Management Science, IEEE Transactions on Engineering Management, Research Policy, Communications of the ACM, and Journal of Product Innovation Management. In 2004, he was elected to the five-year division leadership track of the AOM's Technology and Innovation Management Division.