SUSTAINABLE APPROACHES TO AD-HOC INFORMATION SHARING
FOR VIRTUAL ORGANIZATIONS

BY

NAOKI TANAKA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2014

Urbana, Illinois

Doctoral Committee:

Professor Marianne Winslett, Chair
Professor Carl Gunter
Professor Nikita Borisov
Professor David K. Y. Yau, Purdue University

Abstract

A virtual organization (VO) is a group of organizations that have banded together to achieve a common goal. Often a VO could function more effectively if its members were willing to share certain information with one another. However, some of the information may be sensitive, and a typical VO member will not want to share its own information with others, because the member will not benefit directly from the information's reuse, yet will be blamed if the reuse turns out badly.

Many of the challenges in trying to encourage VO members to share information have roots in traditional approaches to authorization, which try to *eliminate* risk for individual VO members, rather than *maximize VO productivity while bounding risk*. In this thesis, we explore two approaches for VOs to encourage ad-hoc information sharing in an economically sustainable manner without taking on excessive risk. These two approaches can be mixed and matched as appropriate for a particular VO.

The first approach, *portfolio optimization*, maximizes the VO's benefits from sharing, while bounding the volatility (risk) associated with those benefits. This framework addresses two core problems not handled by prior work. The first is to account for VOs with different decision making styles characterized by a risk aversion index. The second is the assessment of risk from the perspective of the entire VO, including the impact of correlated transactions whose risks may be super-additive or sub-additive.

In the second approach, *insured access*, the VO uses an insurance scheme to reimburse damages to VO members attributable to sharing their own information. We show how to estimate the risk associated with an insured access, i.e., the probability distribution of future damages to the VO member providing the information. We also show how reinsurance

can cap the risk associated with rare events, and propose profit-sharing and fee-for-service schemes to ensure that information providers directly benefit from insured access.

Because human decision-makers are influenced by many factors other than the mathematical formulations that underlie insured access, we conducted experiments with humans through a crowd-sourcing service. Our experiments found that over half of the participants chose to use insured access to obtain information that was highly likely to significantly improve their performance in a simulated supply chain scenario, even though the price of the insurance required for the access was subjectively high. We also found that three-quarters of all information producers in our experiments agreed to share sensitive information about their business with insured access when simple administrative procedures, straightforward accountability for and recognition of harm in the rare cases where it does occur, appropriate compensation levels for harm, and attractive profit-sharing are in place. This suggests that insured access can benefit a VO, even when human decisions are involved.

To my parents and brother.

Acknowledgments

First of all, I would like to thank my advisor, Professor Marianne Winslett. She gave me the freedom to pursue research areas that interest me and also provided me with valuable guidance over the course of my PhD research. My research topic was unique in that methods require knowledge in unfamiliar areas, but she helped me in such a way that both she and I were able to learn new things and apply them to my research. She also provided me with many opportunities to grow as a researcher, such as reviewing papers and mentoring undergraduate students. One exceptional opportunity I got from her is the chance to stay in Singapore for a year. Thanks to this, I was able to meet with researchers there and learn many things from them. Without all of this support from her, I could not have been able to finish this thesis.

I also would like to express my gratitude to the other members of my PhD committee: Professor Carl Gunter, Professor Nikita Borisov, and Professor David K. Y. Yau. I have worked with Professor Carl Gunter since my very first semester at UIUC. It was a great learning experience for me to go through all the research processes from deciding a topic to writing a paper for the course project of his class. I thank Professor Nikita Borisov for providing me with insightful feedback that guided me to think how to improve my future study. Professor David K. Y. Yau also has helped me a lot since the very early stage of my PhD research. My stay in Singapore was very fruitful because I was able to discuss research with him face to face.

I thank my collaborators. I had many Skype meetings with Professor Adam J. Lee at University of Pittsburgh, and he always gave me valuable feedback and questions on my research. I also got much help from him when I presented my paper at CODASPY 2013.

When I was in Singapore, Dr. Feng Bao provided me with freedom to work independently and I appreciate his guidance during that time.

Thanks are also due to my coauthors: Chris Ma, Dong Jin, Yuhao Zheng, David C. Bergman, Joshua Juen, and Andrew Wright for their hard work to publish papers together. I am also thankful to my group members Ragib Hasan, Sruthi Bandhakavi, Arash Termehchy, Huong Vu Thanh Luu, Yodsawalai Chodpathumwan, Amirhossein Aleyasin, Joana M. F. da Trindade, Yun Hee Lee, and Kazuhiro Minami for providing me with useful feedback on my work and sound advice for my presentations. I had valuable discussions with my undergraduate intern Heyu Xiong, whose expertise in finance helped me to get a head start of my thesis research project.

Donna Coleman provided invaluable assistance with all administrative tasks I needed, from the arrangements for conference travel to the complicated processes for using crowdsourcing services. I also would like to acknowledge the help I received from Mary Beth Kelly, Rhonda McElroy, Kathy Runck, Holly Bagwell, Kimberly Elaine Bogle, and Mas Saleh.

I appreciate the support I got from my friends at UIUC: Maksym Katsev, Qieyun Dai, Gourab Kundu, Kora Bongen, Wan Chen, Nikita Spirin, Tanvir Amin, Shen Li, Xiaoqian Chen, Aiko Takazawa, Yuka Bannai, Yukiko Kaneko, Miyako Shiraishi, Saiko Rosenberger, Tomonori Yamada, Masaki Wakabayashi, Takayuki Kawai, Tomohiro Shiraki, Shinichi Hatanaka, Tatsuya Sakurahara, Nobutoshi Ota, Kohei Shintani, Tomohiro Sengoku, Norihisa Washitake, Norimasa Otsuki, Naoki Shibata, Akira Uchiyama, Yusuke Takahashi, Ryuta Komaki, Yukako Kumada, Tomoki Ozawa, Tokiha Masuda, and Nana Arizumi. I had a wonderful time with them in Champaign-Urbana.

My parents Yasunori and Eiko have always supported me and allowed me to pursue what I want. I cannot appreciate their understanding enough. I finish these acknowledgments with a sad note on the unexpected demise of my older brother Masaki. He was an excellent brother for me. I would not have been what I am now without him. I miss all the time we spent together so much, and am thankful for all he did for me.

My research presented in this thesis was supported by the National Science Foundation under grants CNS-0963943, CNS- 0964295 and CNS-0963715.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

A virtual organization (VO) is a group of organizations that have banded together to achieve a common goal. Travica [85] defines a VO as a new organizational form that manifests itself as a temporary or permanent collection of geographically dispersed individuals, groups or organizational units, all of which may or may not belong to the same parent organization. Example virtual organizations include a consortium of companies responding to a business opportunity, agencies working together to respond to a flood or nuclear disaster, and intelligence organizations trying to combat terrorism. Any sufficiently large organization operates as a VO, because its internal divisions have their own vested interests that do not always align with the VO's best interests.

For example, in the computer industry, Sun Microsystems (now owned by Oracle) is a highly decentralized organization comprised of independently operating companies [29], each of which may be considered a member of the overarching Sun VO. Sun sets up additional VOs in the form of "SunTeams," whose members operate across time, space, and organizations to address business issues. Sun managers identify key customer issues and then form teams with the skills and knowledge needed to address the issue. This team might include Sun employees from sales, marketing, finance, and operations from multiple countries, and even customers and suppliers from outside Sun. For a SunTeam virtual organization to be successful, it needs to include all the key stakeholders and expertise, have a clear purpose that binds its members together, and establish efficient channels of communication between the members.

Some members in a VO may hold information that is useful to other members, and VO members often need to share information with each other to be successful in achieving the shared goal. If the members are able to access information from each other, they will be able to do their jobs better and complete tasks that are beneficial to the VO as a whole. For example, the customers on a SunTeam may have a much better understanding of their requirements than do other members of the SunTeam, and this understanding may be critical for achieving the SunTeam's mission. However, the useful information may also be sensitive. For example, the Sun customer's requirements may involve details of future software products that have not yet been made public, or sensitive information about projected sales, critical bugs, market evolution, or competitive analyses. Accesses to such sensitive data must be carefully guarded, and permissions to use the data must be subject to rigorous control.

Information sharing in a VO usually requires a VO member to take information that it assembled for its own internal purposes and goals, and release it to another member for a different purpose. Since the first member – the information *producer* – has set the access control policy to match its original intended internal use for the information, sharing requires policy changes. Further, the producer will usually be blamed if another VO member – the information *consumer*[1] – misuses the information, and will not directly benefit if the consumer makes good use of the information.

Thus, a VO's producers are often reluctant to share, which is rational from a self-interested point of view. For example, this is one reason why the FBI and local law enforcement agencies are reluctant to share information with one another. As another example, the research arm of an enterprise often finds it hard to get real user data from the product arm, even though the enterprise as a whole might benefit. Another example is a consortium of big pharma. All participating pharmaceutical companies can benefit from past clinical trial data for diseases such as cancer to speed up the lengthy drug research and development process, but they are reluctant to share such data because of the concerns about giving away competitive advantages and also because they fear data sharing would violate patients' privacy.

The misaligned incentives for sharing stem from traditional approaches to authorization, which try to *eliminate* risk for individual VO members, rather than *maximize VO productiv-*

[1]A member may both produce and consume information.

ity while bounding risk. Traditional access control frameworks treat data requests as binary yes-no decisions, which may be too rigid. Surely, accesses that are useful and carry no risks should be granted, while accesses that pose clear and serious risks should be denied. In less clear-cut situations, however, individual requests need to be assessed for their trade-offs in costs associated with risks against benefits. Presumably, requests whose expected benefit-cost ratios are higher than a threshold should be allowed. Determining the right threshold can be a difficult problem, however, since a large number of risks that make sense individually may pose a serious concern when they are added up. Technically speaking, it is not even clear what "adding up" risks means exactly, since with correlated data accesses, their risks may be super-additive or sub-additive. Hence it is not possible to determine a priori whether a certain data request should be granted or not, by considering each such request in isolation. A liberal organizational policy to set the binary threshold may make the VO vulnerable to excessive risk taking. Overly conservative organizational policies, on the other hand, may lead to undesirable practices such as permission creep [79]. Further, a slow review process for requests for information sharing may make a request irrelevant by the time if it is granted or denied. For example, aerial photographs of a city may be taken for one purpose during a war, and classified accordingly. A second mission for which they would also be helpful will most likely be long over by the time a request to access the maps for the second purpose is reviewed.

More recently, a paradigm of *risk-based access control* was advocated in the JASON report [35]. In risk-based access control, a VO allocates "risk tokens" to its members, which they can use to "pay" for risky accesses to information that would not otherwise be allowed. This paradigm allows more flexibility in granting accesses to sensitive data, compared with binary control. It recognizes that certain data accesses may carry risks, but it may still be prudent to grant them as long as the resulting risks can be appropriately managed. However, it is not clear why producers would want to participate in a risk-token-based economy, or whether the VO would really benefit. If another VO member misuses the shared information, why would the tokens paid for access be considered adequate recompense for the harm caused to the producer? Token-based economies also face other challenges as well. VO members may hoard tokens in anticipation of future shortages, spend leftover tokens carelessly at the end of a fiscal period, or be unable to obtain tokens when they truly need them.

To summarize, we want an information sharing scheme for a VO that meets the following criteria:

- Decisions on whether access is to be granted, i.e., a particular piece of information is be shared with a particular requestor, should be made quickly.

- Producers should be compensated fairly for the benefits to the VO that result from sharing their information, and for any damage they incur as a result of sharing.

- The information-sharing scheme should encourage consumers who are good risk-takers and discourage those who are bad risk-takers, as defined by the benefit and harm that their sharing produces for the VO and its members.

- The VO must be able to limit the chance that harm to the VO or its members due to information sharing will exceed a given maximum amount. This limit must be valid even if never-before-seen harmful events occur.

In recent decades, the business community has benefited from the use of actuarial methods to manage many kinds of business risks, but information sharing has not been among them. In this thesis, we address this gap by introducing approaches to incentivize information providers to allow risky accesses that are likely to benefit the VO. More precisely, we propose the following:

- *Insured access*, an insurance-based approach to access control that compensates producers for harm attributable to sharing their information.

- *Portfolio optimization* to allow a VO and its members to maximize the likely benefits of risky accesses to shared information, while capping the exposure to risk.

Risk management is a pervasive concern for leaders of any organization. Insured access and portfolio optimization focus on two different aspects of risk management: compensating the VO's producers for harm and ensuring that the risky accesses allowed by the VO are those are likely to benefit it most. The two techniques can be mixed and matched as appropriate for a particular VO and information sharing scenario.

1.1 Research Contributions

The main contribution of this thesis is to show that VOs can encourage ad-hoc information sharing in an economically sustainable manner without taking on excessive risk.

Portfolio Optimization Our work on the portfolio optimization approach makes the following contributions:

- We propose an approach to information sharing that seeks to maximize expected benefits while bounding risks below a given threshold.
- We show how portfolio optimization is adaptable to VOs with different decision making styles, as characterized by a risk aversion index.
- We show how portfolio optimization supports the assessment of risk from the perspective of the entire VO, including the impact of correlated transactions whose risks may be super-additive or sub-additive.
- We provide effective algorithms for deciding which accesses to allow under portfolio optimization.

Insured Access Our work on insured access makes the following contributions:

- We propose economically sustainable methods for a VO to determine the price (insurance policy premium) for a particular information access and decide whether to grant a particular access request, given a bound ε on the risk that the VO and its members are willing to tolerate.
- We show that under insured access, information providers can expect to benefit from sharing and providers will have recourse when information sharing turns out badly for the provider, with probability closely related to ε.

- We demonstrate the use of insured access in a simulated map-sharing scenario, and show that the system behaves as predicted.
- We demonstrate the use of insured access in a simulated supply chain scenario, and show that insured access can benefit a VO, even when human decisions are involved.

1.2 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 gives background on risk management in the context of access control and discusses related work. Chapter 3 describes the portfolio optimization approach, which maximizes expected benefits while limiting risk. Chapter 4 presents insured access, which aims at protecting information producers against damage they incur due to sharing their information. Then, Chapter 5 discusses how crowdsourcing experiments done with humans validate the use of insured access. Finally, Chapter 6 concludes the thesis and provides directions for future work.

Chapter 2

Background

In this chapter, we give essential background for this thesis. First, we discuss the access control problem and define the context in which the proposed optimization framework is expected to apply. Then, we discuss related work in optimal selection of portfolios of financial transactions.

2.1 Risk Management in Access Control

Traditional access control inside a single organization treats access requests as binary yes-no decisions. When a new data item is created, its producer traditionally makes a one-time static decision on who will be allowed to access that item in the future. Usually this decision is aligned to the purpose for which the data was originally gathered. Potential consumers from outside the organization are not usually allowed access. In a VO, the situation is different. Typically, the VO cannot accomplish its mission effectively without some degree of sharing of resources. The question becomes, what information should be shared, to further the purpose that has brought the VO together? From a traditional perspective, accesses to data that are beneficial and carry no risks should be granted. On the other hand, if a request, once granted, will raise major security issues, violate the law, breach ethics, or likely lead to abuses, then the request should be denied. In less clear-cut situations, however, one may have to evaluate a data request for both its potential benefits and potential harm.

The VO may then grant the request if the potential benefits significantly outweigh the potential negative consequences of the access. Determining the bar for *significant enough* is a difficult question at best, however, if not outright impossible. While it is clear that the bar should be high enough to prevent accesses that are obviously bad. When potential accesses are considered individually, however, it could be dangerous to determine that a risk is justifiable because it looks small enough, or small enough compared to the opportunity that it enables. This is because as the VO allows more and more accesses whose risks all look benign individually, these risks will add up so that the VO may enter dangerous ground without realizing it.

In practice, a VO that uses binary access control may be motivated to set the bar high so that it will never conceivably accumulate more risks than it should take on. An excessively high bar, however, restricts the sharing of useful information unnecessarily, thereby impeding the ability of the VO to exploit valuable opportunities that can help it accomplish its mission. Moreover, members of a long-standing VO, such as a large company, may be tempted to counter an overly conservative organizational policy by accumulating access rights over time, even if they have changed job functions or the material operating conditions have changed, leading to the problem of permission creep [79].

To address the problem of encouraging appropriate sharing of sensitive defense-related information in a VO, the MITRE JASON report [35] proposed a *risk-based access control* approach. In their approach, principals use *risk tokens* to purchase access rights to data. Each token embodies the risk associated with "one-day, soft-copy-only access to one document by the average Secret-cleared individual" [35]. The access price is the expected value of damages due to this access, as determined by an appraiser. The VO decides how much risk it can handle during the next fiscal period, creates that many tokens, and allocates them to its members. Among other shortcomings, this formulation of risk-based access control does not effectively control the worst-case aggregate damages, and does not distinguish between good and bad risk-takers. Limiting the number of tokens in circulation sets an upper bound on the *average* risk taken on, but does not control the worst-case exposure. The price of an access is the same, regardless of the track record of the individual and organization making the access, so those with good track records subsidize the accesses of those with poor records.

The JASON report inspired follow-on papers addressing specific aspects of a risk-based approach, such as how to integrate trust and risk into RBAC [43], consider relative security risks in RBAC [73], balance the risks and benefits [89], combine risk-based access control with fuzzy logic [39], and allocate risk tokens to VO members [71]. These prior works on risk-based access control manage risky accesses on an *individual* basis. VOs need, however, a decision framework that evaluates the qualities of data accesses from a more global point of view in order to enable productive information sharing between VO members, while ensuring prudent risk taking.

To address this problem, in this thesis we assume that the potential benefits and harm of individual access requests can be measured or at least estimated with reasonable accuracy. This assumption allows a VO to quantify the effects of alternative access decisions and compare them in a definite, albeit probabilistic, manner during decision making, and measure the quality of the resulting decisions.

The above assumption is used also in proposed market-based systems for access control [71], which treat data accesses like they are financial transactions with measurable outcomes. Compared with their work, ours makes uncertainty management explicit by treating the outcomes of data accesses as probability distributions, which also allows us to consider interesting effects such as correlated transactions and hedging. Indeed, we view the issue of correlated transactions as a fundamental weakness that has not been well addressed so far in the market-based approach in [71]. While the market approach seeks to bound the global risk in the system by limiting the number of risk tokens that are in circulation, it is not clear how a VO can fairly and efficiently price a risky access based on its global impact on the VO. This is because the fair price will depend on other accesses that have been bought successfully by all the VO members, including those which are not involved in the transaction in question.

One important issue that remains to be addressed is why a self-interested VO member would be willing to share information it produces. As mentioned earlier, a self-interested producer may be hurt by sharing, and is unlikely to benefit. An important related issue is the need for a clear bound on the worst-case damages that the information producers in the VO may suffer as a result of sharing. A third open question is how to ensure that consumers who abuse shared resources are treated differently from those who use them well. With no incentive for consumers to try to use information well, they may expose the VO to unnecessary risk. We address these three issues in this thesis.

Token-based approaches face other challenges as well. VO members may hoard tokens in anticipation of future shortages, spend leftover tokens carelessly at the end of a fiscal period, or be unable to obtain tokens when they truly need them. Cash-based schemes such has ours will lessen these problems, assuming that VO members spend their cash in the way that brings them greatest utility, i.e., they behave rationally.

Recent work proposes ways to price access to personal data, such as online behavior and demographic characteristics, using differential privacy and auction mechanisms [49, 76]. Like the *insured access* approach in this thesis, these works propose pricing schemes for sensitive information. However, these works assume that upfront payments are sufficient to entice providers to participate. This is appropriate for the low-risk settings these works target, where an individual's aggregate income from sharing is highly likely to exceed their aggregate damages, and a very lightweight scheme is required for efficient deployment. In contrast, insurance is appropriate for situations where there is a very small chance of very high damage; insured access targets this high-stakes lawsuit- and investigation-prone territory. The introduction of an insurance scheme could allow private data pricing schemes to expand their scope of applicability into riskier territory.

Conversely, insured access could use auction-based data pricing to bring providers higher profits than the fee-for-service and profit-sharing schemes proposed in this thesis. However, to reach a fair price, auctions need multiple potential bidders. Insured access is aimed at ad-hoc, non-routine information needs that are too atypical to be institutionalized into a VO's role-based access control system. Single-expected-bid auctions will default to the minimum allowed bid, which is a fee-for-service model. If multiple potential bidders are likely, then the information need is probably sufficiently routine for the VO to adopt lighter-weight methods than insured access.

Finally, insured access can benefit from ways to reduce the sensitivity of shared information. For example, if a consumer needs only part of a sensitive map, then removing the other parts may greatly reduce the expected damages and hence the price of insurance. A second technique is to generalize the shared information, e.g., releasing the approximate location of a gas pipeline rather than its exact location. Differential privacy is a third technique, but today it is practical only when the consumer needs the result of a statistical analysis over many data items. Differential privacy is not yet effective when the consumer needs the data items themselves, such as a set of maps or phone numbers.

Hoo [54] was perhaps the first to indicate the potential of using actuarial methods to manage computer security risk of all types. Hoo's focus was much broader than ours, and he did not consider issues specific to access control and information sharing, but his working paper remains an excellent introduction to the topic.

In summary, we claim that insured access is the first complete, economically sustainable system for encouraging appropriate information sharing in VOs.

2.2 Portfolio Management

Techniques for optimizing the return on investments such as stock and bond portfolios are a topic of intense interest in the finance industry. A VO's portfolio can be viewed as the set of accesses to shared information that the VO has permitted, out of all the requests it has received. Then the optimal portfolio is the one that is expected to benefit the VO as much as possible, while not exposing it to more risk than it is comfortable with.

The Markowitz mean-variance optimization framework [66], which explicitly manages the risk dimension of investments quantified as the variance of returns, is considered seminal work in portfolio selection for financial investments. The Markowitz optimization aims to minimize the variance of returns subject to a minimum expected return, or maximize the expected return subject to a limit on the variance. The latter formulation is similar to the OPT-VAR problem in Chapter 3, although our problem is discrete and we also propose another OPT-TAIL problem to allow a VO to focus on particular tail regimes of the return distribution.

The Markowitz optimization can be solved as a quadratic program using standard techniques in convex optimization [66, 67]. However, later work has recognized that the basic Markowitz framework does not admit several important constraints for practical investments, such as trading costs and turnovers [74, 70]. The majority of solutions in response do not apply metaheuristic local search methods, despite their advantage of good practical performance for problems without a "nice" mathematical structure. An exception is the work of Crama and Schyns [40], who use simulated annealing to optimize portfolio selections with attention to trading and turnover constraints. As in the Markowitz problem, they focus on variance as the only notion of risk, whereas we consider broader definitions. The trading constraints in their environment are not important in our problem context, which considers risky data accesses allowed by a VO. Further than their work, we model explicitly the different natures and degrees of risk aversion of VOs that will impact significantly their decision making.

An analysis of simulated annealing in our specific problem context is important, since it is well known that the structures of specific problems will impact the design and complexity of effective search strategies in basic ways [78]. In particular, we show that (i) finding a feasible and good initial solution to seed the search in our problem is non-trivial, and (ii) our problem exhibits disconnected feasible regions that present challenges for "local search" steps to find the global optimum with high probability. We overcome these problems by a novel unified two-phase algorithm.

Besides its modeling power, the spectrum of risk aversion we consider (in contrast to prior work in portfolio selection) also has implications on the solution methods. In particular, for the OPT-VAR problem under independent transactions, we show that risk-neutral and a type of *risk-invariant* decision makers can employ an efficient hybrid greedy algorithm that (i) can solve the problem to within a constant approximation factor of the optimal analytically, and (ii) has close-to-optimal performance in various numerical experiments.

Chapter 3

Portfolio Optimization

Given a set of requested accesses to shared information in a VO, which should be granted and which denied? In this chapter, we show how to optimize the expected net benefits to the VO of the set of granted accesses, while limiting the potential damage/harm of the accesses to a level that the VO is comfortable with, according to its risk tolerance. In this chapter, we show how financial portfolio optimization techniques can be used to optimize a VO's set of granted cross-organizational accesses. In contrast to prior work on risk-based access control that manages risky accesses on an *individual* basis [35, 39, 43, 89], our portfolio optimization techniques manage such risks in a *global* sense.

We use the term *consumer* to refer to a functional unit of the VO that requests to access data owned by another unit in the same VO, which we refer to as the *producer*. The same functional unit can be both a producer and consumer of shared data. When a consumer wants to access shared data, it submits an access request to the VO for approval. We call such a request a transaction if it is granted. The set of granted transactions forms a *portfolio* owned by the VO.

A transaction is inherently uncertain in that its long-term financial impact on the VO is not known at the time the data access is approved. We assume that from historical data and other sources, the VO knows the probability distribution associated with potential outcomes of each transaction. Specifically, each transaction, say i, will realize a *profit* (in dollars) for the VO drawn from a probability distribution denoted by X_i, and losses (i.e., negative profits) are possible. For a transaction to be helpful to the VO, its expected profit must be positive, and we assume that only such transactions will be considered for inclusion in the portfolio. However, a positive profit in the expected sense is not sufficient by itself, since the

VO is also concerned about the possibility of losses that it may incur. It aims therefore to bound the probability that the portfolio will produce a net loss, or more generally a profit that falls short of some threshold denoted by L (in dollars).

To be able to apply portfolio optimization techniques, a VO must consider multiple access requests at the same time. For example, consumers can submit their data requests for approval by the VO in rounds. In each round, all the new data requests will compete for inclusion in the portfolio (which may already have some existing transactions). In resolving this competition, the VO analyzes the cost-benefit of each candidate transaction. We recognize that transactions can be correlated in general, i.e., X_i and X_j may not be independent for $i \neq j$. Hence, the cost-benefit analysis may need to consider groups of transactions as units, rather than individual transactions. For example, if two transactions are positively correlated, their risks amplify each other and are thus *super-additive*, e.g., a database join that binds the names of patients to diseases. On the other hand, for two transactions that are negatively correlated, hedging comes into effect to reduce the the overall risk, in which case the transactions' risks are *sub-additive*, e.g., access to financial transactions used to derive an annual report will reduce the risk of access to the report itself.

A basic measure of the *benefit* of a portfolio is its overall expected profit for the VO. By the linearity of expectation, this expected profit is simply the sum of the expected profits of all the transactions that make up the portfolio. Because a portfolio has probabilistic outcomes, however, its expected profit is uncertain. An individual may adjust her value of the expected profit according to the variability of the actual profit. How she carries out the adjustment depends on her attitude towards risk, called *risk aversion* [75]. In general, if a risk-averse individual is offered the choice between a guaranteed profit x and a probabilistic (i.e., lottery-like) profit of expectation y, she will demand a premium of y over x for her to value both choices as equally attractive. The premiums demanded by risk-neutral and risk-seeking individuals are zero and negative, respectively.

The *cost* of a portfolio is designed to measure its overall risk, which reflects the probabilities and extents of unfavorable outcomes. For example, a risky access that went wrong could lead to punitive indemnities for damages or a lawsuit. We assume that each possible negative event can be quantified as a financial loss, and is reflected in the portfolio's probability distribution of profits. A simple but common notion of risk is the uncertainty or variability of the outcomes. The celebrated Markowitz portfolio optimization problem [66], for example, aims to maximize the return of a portfolio subject to a risk limit measured as the variance of the returns. The converse problem of minimizing the variance subject to a minimum return can be similarly defined.

Besides variability, some decision makers may further focus on particular regimes over which the outcomes vary. For example, a VO may be primarily concerned about the regime of loss, or even the regime of large losses that threaten its survival. In this case, the variance of X (the probability distribution of profits for the portfolio) can be assessed relative to the mean of X. For example, the VO may demand that $E[X] - c \times \sigma[X]$ exceed some threshold L, for some small constant c, where $E[\cdot]$ and $\sigma[\cdot]$ refer to the mean and standard deviation of a random variable, respectively. For example, if X is normally distributed, $c = 3$, and $L = 0$, the VO's risk management goal is to ensure that the portfolio will be profitable with a probability of at least 99.7%.

In this chapter, we formulate the portfolio optimization of risky transactions as a constrained optimization problem, i.e., we aim to maximize the overall benefit of the portfolio subject to a given risk budget, where the risk budget reflects the risk tolerance of the VO. Unlike traditional portfolio optimization problems, our problem is discrete and combinatorial in nature. Specifically, a transaction that requires b amount of the risk budget must be allocated that whole amount for it to be feasible. Any partial allocation less than b to the transaction would be completely useless. For a VO whose initial risk budget is $c (c \geq b)$, its risk budget becomes $c - b$ if it allows this transaction. This combinatorial nature makes our problem reducible to the NP-hard knapsack problem [68], for which a plausible cost-denominated greedy selection strategy may in fact lead to a solution arbitrarily far from the optimal.

To illustrate, consider a risk budget of \$100 and two competing transactions T_1 and T_2 whose benefit/cost numbers are \$10/\$1 and \$100/\$100, respectively. The cost-denominated

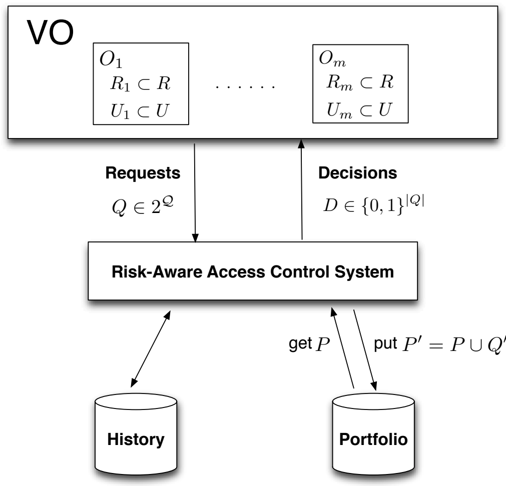


Figure 3.1: System Model for Portfolio Optimization Approach

greedy strategy first includes the transaction with a higher benefit per unit cost, namely T_1. After that, however, the remaining available cost budget is \$99. This remaining budget is completely wasted because it is less (albeit just a little bit less) than the cost required by T_2, although T_2 would have provided a \$100 benefit had it been included in the portfolio.

3.1 Problem Formulation

In this section, we formulate the *risk-aware access control* problem. We first define our system model for the portfolio optimization approach (see Figure 3.1), then we define the optimization problems.

Consider a virtual organization $VO = \{O_1, \ldots, O_m\}$ comprising m member organizations. We further assume the existence of finite sets U and R denoting users and resources, respectively. Let U and R be finite sets denoting the VO's users and resources, respectively. Finally, we let T represent the set of valid timestamps.

Most accesses within the VO will be governed by the use of a standard access control scheme (e.g., RBAC [47, 77]). However, risk-aware access control can be used to override default sharing decisions when atypical sharing may lead to unexpected profit opportunities.

We employ a fairly generic model of risk-based access control that subsumes several proposals in the literature [44, 35, 89, 38, 82]. In particular, let $\mathcal{Q} = U \times R \times T \times \mathcal{X}$ denote the set of access requests, where $(u_i, r_j, t, X_{ij}) \in \mathcal{Q}$ denotes that user u_i wishes to access resource r_j at time t, where X_{ij} is a probability distribution over the profit (in dollars) that the VO expects to derive from the requested access.

We assume that the risk-aware access control scheme records two types of long-term state: an active portfolio and a history of allowed accesses. The VO's *portfolio* is an element of $2^{\mathcal{Q}}$ that represents the risky accesses currently granted by the VO, but whose profits are not yet realized. *History*, on the other hand, is a record of risky accesses that have been granted by the VO and whose profits are known. History is represented as an element of $2^{\mathcal{H}}$, where $\mathcal{H} = U \times R \times T \times \mathbb{Z}$; an element $(u_i, r_j, t, p) \in \mathcal{H}$ denotes that user u_i was granted access to resource r_j at time t, and the (possibly negative) profit from that transaction was p.

The actuarial literature shows that risks can be modeled nicely using the exponential family of distributions. By assuming risks follow such distributions, we can use maximum likelihood estimation techniques [37] to estimate parameters of distributions X_{ij} with the VO's history providing ground truth. From the risk analysis standpoint, negative tail parts of distributions corresponding to rare events are of interest, and extreme value theory can help estimate the tail distributions that lack sufficient history [50]. Thus, it is certainly possible to estimate risk distributions. We do not show an example for this due to the difficulty of obtaining real data. However, real organizations routinely estimate risks using actuarial science, and we are demonstrating the application of these techniques to access control.

As explained earlier, to employ portfolio optimization techniques, requests to access resources are batched by the risk-aware access control scheme, which then adjusts its portfolio in *epochs* (e.g., hourly or daily). Data accesses which were active at the start of an epoch, but whose profits became known within the epoch, will be removed from the portfolio at the epoch's end and recorded in the history. Within this context, we attempt to answer the following question:

Given a portfolio $P \in 2^{\mathcal{Q}}$ at the end of the current epoch, and a set of incoming

Table 3.1: Notations Used in Chapter 3

Symbol	Meaning
$E[\cdot]$	Expected value of a random variable
$Var[\cdot]$	Variance of a random variable
$\sigma[\cdot]$	Standard deviation of a random variable
$Cov[\cdot,\cdot]$	Covariance of two random variables
$U(\cdot)$	A VO's utility function for money
$U'(\cdot)$	First derivative of a function
$U''(\cdot)$	Second derivative of a function
$Val(\cdot)$	Objective value achieved by a portfolio
$r(\cdot)$	VO's index of risk aversion

requests $Q \in 2^{\mathcal{Q}}$, what is the optimal portfolio $P' = P \cup Q'$, where $Q' \subseteq Q$ for the VO to carry into the next epoch?

We now formulate the optimization problems for a VO to select the optimal P'. Without loss of generality, we assume that the original portfolio P is empty. To simplify notations, we enumerate the granted data accesses $\{X_{ij}\}$ as a list of transactions indexed by i, and use X_i to denote the profit distribution of the ith transaction. For a portfolio with n transactions, the profit distribution of the whole portfolio is given by

$$X = \sum_{i}^{n} X_i.$$

If the transactions are independent, X is given by the *convolution* of the X_i's. Furthermore, if each X_i is normally distributed, then X is also normally distributed and can be completely characterized by $Var[X] = \sum_i Var[X_i]$ and $E[X] = \sum_i E[X_i]$. If the transactions are correlated, it will be hard to characterize X completely, although the portfolio optimization problem can still be solved by key first- and second-order statistics of X. We now proceed to formulate two important forms of the portfolio optimization problem, using the notation in Table 3.1.

3.1.1 Benefit Model

As a first approximation, a portfolio is more attractive if it has a higher expected profit, i.e., it is likely to make more money on average. Beyond average performance, however, the perception of decision makers towards possible uncertainty of the performance is important. We account for this perception by the notion of *risk aversion* in this section.

The notion of risk aversion can be well illustrated by a popular TV game show called *Who Wants to Be a Millionaire.* In the game, a contestant randomly selects a box among n boxes in the beginning. Each box holds an amount of money, and the amounts of the set of boxes are known. At each round of the game, a box not held by the contestant is opened to reveal its dollar amount. The host then offers to buy the contestant's box for x dollars. If the contestant accepts the offer, she wins x dollars. If the contestant does not accept any offers until the end of the game, she wins the money in the box she selected at the beginning.

When given an offer, the game contestant chooses between the certainty of receiving x dollars and a "lottery" of winnings whose expected value is y. Typically, x is significantly less than y, but in many cases, real contestants find the choice difficult and some will accept the host's offers. Clearly, the audience takes interest in observing how a contestant will make her decisions, perhaps while also comparing her decisions with those they would make in the same situation.

A contestant can be viewed as considering the "premium" she is willing to pay in order to avoid the uncertainty of a lottery and receive the expected winning with certainty instead. If $y - x$ is less than the premium, the contestant takes the offer, or else she keeps playing. Different individuals have different degrees of risk aversion. If a utility function exists for an individual to rank her preference between any two lotteries, her decision under uncertainty of the outcomes can be completely determined. Specifically, if the function returns a higher number for one lottery than the other, she chooses the first one. She is indifferent if the utility numbers are the same.

VOs are organizations of people, and we expect that a culture of risk taking will develop for them. Hence, utility functions characterizing the risk preferences of VOs are meaningful.

According to Pratt [75], the premium p a VO is willing to pay to avoid the lottery X is given by

$$U(E[X] - p) = E[\{U(X)\}],$$

where $U(\cdot)$ is the utility function of the VO for money. If the VO is risk-averse, U is a concave function, i.e., the marginal utility of more money decreases with the original amount of money. The function is convex if the VO is risk-seeking, and linear if the VO is risk-neutral. In the context of access control, VO members access data to achieve VO goals. As such, good accesses carry monetary benefits (positive utility) and improper accesses or misused information carry with them monetary costs (negative utility). This notion of utility is more general than access control, but as long as access control has wealth implications, how one values money is a basic consideration.

Solving for p, we can show that the premium the VO is willing to pay (up to first order in a Taylor series expansion) is proportional to the *index of risk aversion* [75] $r(\cdot)$ evaluated at $E[X]$. Specifically, we have

$$p = \frac{1}{2} Var(X) \times r(E[X]) + \text{lower order terms},$$

where $r(x) = -\frac{U''(x)}{U'(x)}$. For our portfolio optimization problem, we use the main term of the premium p to discount the value of an expected profit by the variability of the actual profit. Specifically, we define the objective function of our problem as

$$E[X] - \frac{1}{2} Var(X) \times r(E[X]), \tag{3.1}$$

where X is the random variable for the distribution of profits of the portfolio. Note that our variance notion of risk is associated with the variance of profit distribution of the set of allowed transactions that embody a portfolio, and **not** for individual accesses independently.

Notice that U' is non-negative since U is an increasing function, i.e., more money is always welcome by the decision maker. Hence, the sign of the second term depends on the sign of U''. Recall that for a risk-averse individual, U is concave and hence U'' is negative. This means that risk aversion results in a true (positive) discount to the expected profit as benefit. On the other hand, the discount is zero for a risk-neutral individual, and it is negative for a risk-seeking one (i.e., the risk-seeking decision maker in fact derives a positive value from the uncertainty). Moreover, the index of risk aversion does not have to be a constant function. For certain risk-averse decision makers, for example, $r(x)$ may be a decreasing function of x, i.e., the decision maker becomes less risk averse as her wealth increases.

One form of utility function that is widely used in the insurance industry is the *exponential utility function*:

$$U(x) = 1 - e^{-\alpha x}.$$

In the above, α denotes a constant index of risk aversion for the decision maker. Since α is positive, this utility function models a risk-averse decision maker, whose level of risk aversion does not change with her wealth. For this reason, we call a decision maker whose utility function can be modeled using the above exponential function *risk-invariant*.

3.1.2 Cost Model

The VO aims to maximize the objective function (3.1) subject to a cost constraint. The constraint allows the VO to bound the amount of risk in its portfolio to within a certain risk tolerance. We now define two interpretations of the risk amount. These interpretations define the meaning of the units in the risk budget.

Variance as notion of risk

We start with a basic interpretation of risk, which is used in some classical approaches such as the aforementioned Markowitz optimization problem [66]. In this basic interpretation, the amount of risk associated with a portfolio of access requests corresponds to the uncertainty of returns associated with granting these requests, which can be quantified as the variance of the portfolio's probability distribution of profits. Formally, the cost constraint for the optimization of (3.1) is given by

$$Var[X] \leq B, \tag{3.2}$$

where B is a constant giving the maximum variance tolerated by the VO.

If the transactions in a portfolio are independent, their variances are additive. Hence, we have

$$Var[X] = Var[X_1] + \ldots + Var[X_n].$$

In this case, including a certain transaction in the portfolio commits a fixed amount of the risk budget, irrespective of the other transactions already in the portfolio.

If the transactions are correlated, we must also consider their covariances, and we have

$$Var[X] = \sum_i Var[X_i] + \sum_{i,j} Cov[X_i, X_j].$$

In this case, when a transaction, say T, is added to an existing portfolio, say P, its effect on the total risk will depend on the exact transactions originally in P. Specifically, T will reinforce the risks of those transactions with which it is positively correlated, and mitigate the risks of other transactions with which it is negatively correlated. Risk reduction due to negatively correlated transactions gives the principle of hedging. The net effect of adding T will depend on the competitive effects between both types of transactions, and the VO aims to best exploit the hedging effect to accommodate more productive transactions.

Definition 1. *We use Problem OPT-VAR to refer to the optimization of objective function (3.1) subject to constraint (3.2).*

Tail regime of profits as notion of risk

Further to the variability of profits, we believe that some decision makers may focus on particular regimes over which the outcomes vary. In particular, risk may be regarded as the chance that a significantly unfavorable profit (i.e., a large loss) will be realized. To accommodate this view of risk, we notice that actual outcomes of a portfolio generally vary about the mean of the outcomes. Hence, a higher average profit will provide a larger buffer before increased profit variability will pose a significant concern. The variance of profits should thus be assessed relative to the mean of profits.

Formally, we can impose the following constraint on the optimization problem:

$$E[X] - c \times \sigma[X] \geq L, \tag{3.3}$$

where c is some small positive constant and L is a constant giving a profit threshold below which the VO's concern will be triggered. It is expected that if the condition is satisfied, then the probability that the portfolio generates a profit less than L will be sufficiently small to pass the VO's risk tolerance test. This probability will depend on the distribution of X assumed by the VO. Given the distribution, c can be made large enough to give a small enough probability. For example, if X is normally distributed, then the probability is 5% for $c = 2$ and 0.3% for $c = 3$.

The constraint (3.3) can be written in a similar form as (3.2) to make the risk budget apparent:

$$c \times \sigma[X] - E[X] \leq B, \tag{3.4}$$

where $B = -L$. However, unlike the sum of variances in (3.2), the standard deviations of component transactions are not additive even if the transactions are independent.

Definition 2. *We use Problem OPT-TAIL to refer to the optimization of objective function (3.1) subject to constraint (3.4).*

3.2 Solution Methods

In this section, we present solutions to the problems defined in Section 3.1. First, we focus on Problem OPT-VAR under independent transactions. We show that certain interesting special cases of this problem can be solved by an approximation algorithm operating in a hybrid-greedy manner. We then address the general problem, and propose the use of simulated annealing as a basic solution technique. We analyze the structure of our problem, and devise effective search strategies for the simulated annealing.

3.2.1 OPT-VAR for Independent Transactions

Under independent transactions, setting the index of risk aversion $r(\cdot)$ to a constant function (e.g., that of a risk-invariant VO) reduces Problem OPT-VAR to the knapsack problem. Hence, the problem is NP-hard. As illustrated in the beginning of this chapter, a cost-denominated greedy algorithm, though plausible, can in fact lead to a solution that is arbitrarily far from the true optimal. We use S_{CG} to denote the solution of this cost-denominated greedy algorithm.

An examination of the S_{CG} solution in the example mentioned in the beginning of this chapter might suggest that we use a simple greedy algorithm instead, in which we repeatedly add a transaction to the portfolio whose benefit is the highest until we run out of the risk budget. We denote the objective value achieved by the portfolio as S_G. Unfortunately, S_G can also be arbitrarily far from the true optimal. To see why, consider 11 transactions of which 10 of them have benefit/cost numbers \$10/\$1 and one has benefit/cost numbers \$20/\$10. For a cost budget of \$10, the greedy solution includes the \$20/\$10 transaction and exhausts its budget. However, selecting all the other transactions instead for the portfolio would have resulted in a total benefit of \$100.

Although each of S_{CG} and S_G can be arbitrarily suboptimal, it turns out that a hybrid of these two solutions will provide a good approximate solution for the special cases of risk-neutral and risk-invariant VOs. For this, we state the following two lemmas.

Lemma 1. *For a risk-neutral or risk-invariant VO, the objective function in Problem OPT-VAR under independent transactions is* monotonic*, i.e., adding a new transaction to a portfolio will not reduce the objective value achieved by the portfolio.*

Proof. Since only transactions with positive expected profit will be considered for inclusion in the portfolio, a candidate transaction, say i, must satisfy (i) $E[X_i] > 0$ for a risk-neutral VO, or (ii) $E[X_i] - \frac{a}{2}Var[X_i] > 0$ for a risk-invariant VO. The result follows. □

Lemma 2. *For a risk-neutral or risk-invariant VO, the objective function in Problem OPT-VAR under independent transactions is* submodular*, i.e., if P_1 and P_2 are two portfolios and*

$P_1 \subseteq P_2$, then for any transaction T, $Val(\{T\} \cup P_1) - Val(P_1) \geq Val(\{T\} \cup P_2) - Val(P_2)$, where $Val(\cdot)$ denotes the objective value achieved by a portfolio.

Proof. Because of the independence assumption, $Val(\{T\} \cup P_1) - Val(P_1) = Val(\{T\} \cup P_2) - Val(P_2) = Val(\{T\})$, which satisfies the submodularity property. □

Lemma 2 still holds even if transactions with nonpositive expected profit are allowed

We are now ready to state Theorem 1, which shows that OPT-VAR can be solved to within a constant approximation factor by a $O(n \log n)$ time *hybrid-greedy* algorithm, where n is the number of candidate transactions.

Theorem 1. *For a risk-neutral or risk-invariant VO under independent transactions, the solution* $\max\{S_{CG}, S_G\}$ *is within a constant factor of the optimal solution of Problem OPT-VAR. Specifically,*

$$\max\{S_{CG}, S_G\} \geq \frac{1}{2}(1 - \frac{1}{e}) \times S^*,$$

where S^* is the optimal solution.

Proof. Since the objective function of Problem OPT-VAR is monotonic and submodular under the given conditions, the result follows directly from Theorem 3 in [64]. □

Unfortunately, Theorem 1 does not apply for general types of decision makers or if the access requests comprising a portfolio are not independent. These other cases of Problem OPT-VAR will be addressed in the next subsection, which proposes a simulated annealing approach for the more general problem.

3.2.2 General Problems

Simulated annealing [61] is a general methodology for finding optimal solutions using local search, where randomness is used in the search to avoid getting stuck at local optima so that a globally optimal solution will be achieved with high probability. We use simulated annealing to solve the general case of the portfolio optimization problem, particularly the OPT-TAIL problem and instances of the OPT-VAR problem for which Theorem 1 does not apply. To use simulated annealing for specific applications, three design decisions are important: (i) how to find a good feasible solution as a starting point to seed the optimization process, (ii) how to define the neighborhood structure of candidate solutions that will guide the solution process, and (iii) how to define the cost function that will allow to rank the candidate solutions.

For (iii), our problem statement in Section 3.1 provides a natural definition of the cost function, which is the benefit (in dollars) achieved by a candidate solution. For the neighborhood structure in (ii), we define a solution S' to be a *neighbor* of a certain solution S, if S' can be obtained from S with a change that preserves the feasibility of the solution. Specifically, with N candidate solutions, we may view a solution as a vector of N bits where the ith bit is 1 if the ith transaction is in the solution and it is 0 otherwise. From this point of view, flipping k bits in this vector corresponds to a possible change to the original solution. In particular, if $k = 1$, the change can be considered *minimal*, and many existing simulated annealing solutions define their neighborhood structures to consider only minimal changes. Using only minimal changes in our problem is sub-optimal, however, since it exhibits disconnected feasible regions (Section 3.3.1). When the feasible regions are disconnected, the minimal changes cannot move pass the region boundaries, and so the quality of the final solution will be heavily dependent on where the search starts. Hence, although we still prefer changes that are minimal ($k = 1$), we do occasionally consider *long jumps* with $k > 1$ also. Specifically, in a search step, we draw i from $\mathcal{N}(0, 1)$, i.e., the normal distribution whose mean is zero and variance is one, and assign k to be $\lfloor |i| \rfloor + 1$. If $k = 1$, the change will always be evaluated further. If $k > 1$, the change will be considered further with probability p_{lj}.

Problem (i) of seeding the optimization with an initial feasible solution is also worthy of attention. This is because in our constrained optimization problem, finding any feasible portfolio that satisfies the constraint can be non-trivial. To see why, consider the case of correlated transactions under Problem OPT-VAR. Since the variance of a portfolio may *decrease* with adding more transactions to the portfolio, in principle we may have to examine each element in the powerset of candidate transactions individually to assess feasibility. In particular, we may not be able to initialize the search with the empty portfolio, or with any single candidate transaction if each of the transactions has a variance higher than the cost budget.

To solve the problem, our algorithm runs in two phases. In Phase 1, subject to a given computational budget, we aim to find an initial portfolio that satisfies the problem constraint and that has reasonably good performance. This initial solution is then used to seed the Phase 2 search that progressively improves the quality of solution until we use up another given computational budget. We use a unified simulated annealing approach to conduct both phases, in which the aforementioned long jumps ($k > 1$) are allowed occasionally to overcome the myopia of nearest-neighbor explorations. The two phases have different basic design objectives, however. As mentioned, the first phase is mainly concerned with initializing the search. On the other hand, the second phase is mainly concerned with carrying on the search based on the initial solution computed by the first phase. To account for their different objectives, we set the probabilities of considering long jumps to be $p_{lj} = p_{lj}^1$ and $p_{lj} = p_{lj}^2$ for Phases 1 and 2, respectively, where $p_{lj}^1 > p_{lj}^2$. By using a higher value for p_{lj}^1, we encourage the search in Phase 1 to explore new information. By being more conservative with long jumps in Phase 2, due to the smaller p_{lj}^2, we encourage the Phase 2 search to stay more focused in improving a promising solution.

The Phase 2 simulated annealing algorithm is implemented as follows. (The Phase 1 implementation is essentially identical, and we omit it in this thesis.)

1. Initialize the candidate portfolio to be the Phase 1 solution.

2. Randomly draw i from $\mathcal{N}(0,1)$ and assign k to be $\lfloor |i| \rfloor + 1$. If $k > 1$, then with probability p_{lj}^2, accept k. If k is not accepted, repeat Step 2.

3. Randomly generate an N-bit update vector with k of the bits equal to one. If the ith bit of the vector is 1, reverse the status of the ith transaction (i.e., whether it is included or not) in the candidate portfolio. Otherwise, keep the status of the ith transaction the same.

4. Update the candidate portfolio according to the update vector if doing so will not violate the constraint, and either (i) the objective value is increased after the update

or (ii) $e^{\frac{-\Delta O}{l \log t}} > rand()$, where ΔO is the absolute difference between the original objective value and the objective value after the update, l is a constant, t is the number of iterations executed so far, and $rand()$ is a random number uniformly chosen from [0,1].

5. If the candidate portfolio has a better objective value than the best one found so far, remember the candidate portfolio as the best one so far.

6. Go back to Step 2 if the computation budget is not yet reached. Otherwise return the best candidate portfolio found so far.

Notice that in Step 4, we always accept a random change to the candidate portfolio if doing so will improve the benefit without violating the constraint. Even if a change may decrease the portfolio's benefit, however, we may still accept it with non-zero probability. This is necessary because the solution space may have local maxima, in which case the random acceptances of less good candidate solutions help us escape local maxima in the solution process. At any time, however, the probability of accepting a less good solution decreases with the quality of the solution due to the presence of ΔO in $e^{\frac{-\Delta O}{l \log t}} > rand()$. Moreover, this probability decreases as the search time represented by t increases. The decreasing acceptance probability with t simulates *cooling* in the simulated annealing process [61].

The convergence of simulated annealing has been assured analytically [52]. The analysis is mostly of theoretical interest, however, since the convergence speed of real applications is usually much faster. Further to convergence which affects the number of iterations, the time complexity of an iteration is an important consideration for performance. In our problem, this time complexity is determined by that of evaluating the effects of adding or removing a transaction to or from the candidate portfolio. For both problems OPT-VAR and OPT-TAIL, the time cost is constant under independent transactions and linear in the number of candidate transactions under general correlated transactions.

3.3 Experimental Results

We present experimental results of both one-round and multiple-round simulations to illustrate various aspects of our problem formulation and methods. As already mentioned, testing the techniques for estimation of distributions from history data is beyond the scope of this thesis as real data is not available to us.

3.3.1 One-Round Simulation Results

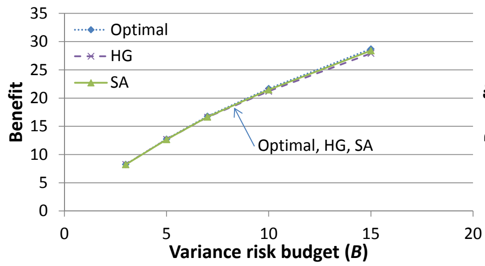
We present experiments to illustrate the performance of the hybrid-greedy (HG) algorithm (Section 3.2.1) and simulated annealing (SA) algorithm (Section 3.2.2), according to the portfolio values they achieve for objective function (3.1). For a baseline of comparison, we use exhaustive search, whose solution is necessarily optimal but extremely inefficient. Exhaustive search is clearly not a practical method in general, but where it returns a solution in our experiments (for up to 15 transactions), it provides an optimality benchmark.

We consider both independent and correlated transactions. For independent transactions, the profit of each transaction is normally distributed with mean chosen uniformly randomly from [1,10] and variance chosen uniformly randomly from [1,12.25]. For correlated transactions, the first one is normally distributed with mean and variance chosen uniformly randomly from [1,10] and [1,12.25], respectively. The profits and variances of the remaining transactions are each a combination of the first transaction with another normal random variable by the Cholesky method with a correlation factor uniformly distributed in [-1,1].

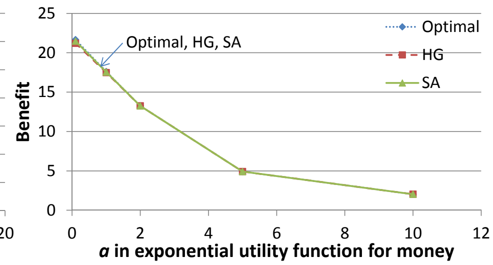
A solution's *competitiveness* is the ratio of its objective function to that of the optimal solution. Unless otherwise stated, we use a risk-invariant VO with parameter α and present results that are averages of 1000 independent runs.

Problem OPT-VAR

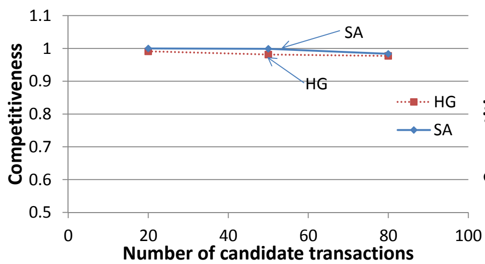
In this section, we report experiments for variance as the notion of risk (Definition 1). We first consider the case of 15 independent transactions. Because we use a risk-invariant VO,



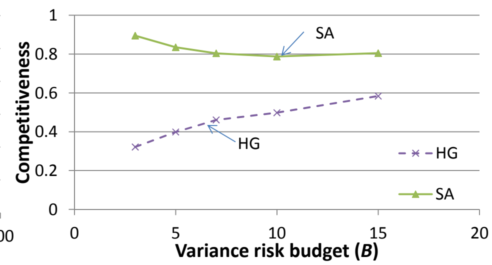
(a) 15 independent transactions, $\alpha = 0.1$



(b) 15 independent transactions, $B = 10$



(c) $\alpha = 0.1, B = 5$



(d) 15 correlated transactions, $\alpha = 0.1$

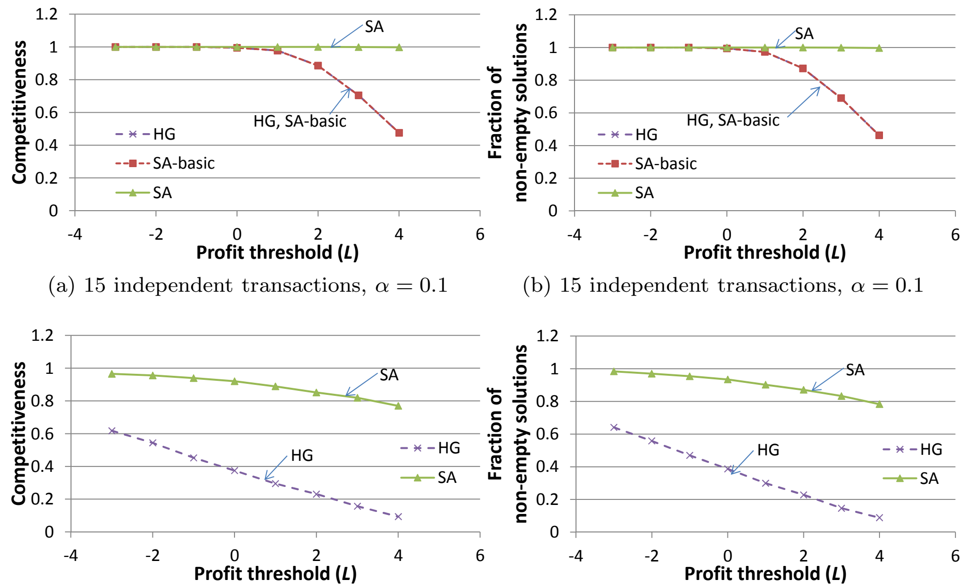
Figure 3.2: Problem OPT-VAR Simulation Results

Theorem 1 applies to the HG algorithm. The SA computational budget is 8000 accepted update vectors, in which Phase 1 has a budget of 100 accepted update vectors. This budget corresponds to a running time of about 0.007s on an Intel 2GHz dual-core machine with 2GB of RAM.

Fig. 3.2a plots the HG, SA, and optimal solutions for $\alpha = 0.1$ and varying risk budget B. Fig. 3.2b shows the corresponding results when we vary α and fix $B = 10$. Fig. 3.2a shows that as the risk budget increases, the benefit of the optimal portfolio increases, because more productive transactions can be included. On the other hand, Fig. 3.2b shows that as α increases, the VO becomes more risk averse, and the benefit of the optimal portfolio drops. This shows that a more conservative VO will achieve a smaller benefit in general.

Both Figures 3.2a and 3.2b show that SA is able to find the optimal solutions. HG's performance is also extremely close to the optimal, which significantly exceeds the lower bound in Theorem 1. These results show that HG is highly effective when the conditions of Theorem 1 apply.

For more than 15 transactions, exhaustive search fails. However, OPT-VAR under in-



(a) 15 independent transactions, $\alpha = 0.1$ (b) 15 independent transactions, $\alpha = 0.1$

(c) 15 correlated transactions, $\alpha = 0.1$ (d) 15 correlated transactions, $\alpha = 0.1$

Figure 3.3: Problem OPT-TAIL Simulation Results

dependent transactions can be formulated as a mixed integer linear program (MILP). We use CPLEX [56] to solve the MILP for 25–80 transactions and provide an optimality benchmark. Fig. 3.2c shows that both HG and SA achieve competitiveness close to 1 even when the number of transactions is large.

We now consider OPT-VAR under correlated transactions. We use 15 correlated transactions. We plot the HG and SA solutions in Fig. 3.2d when $\alpha = 0.1$ and the risk budget B varies. In this case, with the small computational budget, the SA solutions achieve a competitiveness of more than 0.8. On the other hand, the gap between the HG solution and the optimal becomes large. When B is small, the HG solution is only about 30% of the optimal.

Problem OPT-TAIL

In this section, we present results that focus on the tail regimes of the profit distribution with profit threshold L (Definition 2). First, we report results for 15 independent transactions. We use an SA computational budget of 1000 accepted update vectors, in which Phase 1 has a budget of 100 accepted update vectors.

Fig. 3.3a compares the competitiveness of the SA and HG solutions as a function of L. Notice that the competitiveness of the HG solution drops to about 50% as the profit threshold L becomes larger (more stringent). By comparison, the SA solutions have a competitiveness close to one throughout the range of L.

It is instructive to understand the significant underperformance of HG. It is because under a tail constraint on the profit distribution, more than one transaction may be needed to achieve the profit threshold L. In this case, since HG adds transactions to a candidate portfolio one at a time, and ensures the feasibility of the candidate portfolio in each step, it will never be able to find a portfolio that is feasible. This phenomenon can be observed in Fig. 3.3b, which plots the fractions of runs in which HG terminates with a non-empty portfolio. As shown, the fractions can be low and mimic the sub-optimality of HG in Fig. 3.3a.

In fact, SA could be susceptible to the same problem like HG. Consider a basic variant of SA, which we call SA-basic, which has one single search phase only. The search starts from the empty portfolio if it is feasible (i.e., $L \leq 0$), and uses a neighborhood structure that allows minimal perturbations only (see Section 3.2.2). If the transactions are such that no single one satisfies the risk constraint by itself, but there is a set of multiple transactions that does, SA-basic will return the empty portfolio although it is clearly sub-optimal. Fig. 3.3b demonstrates this problem, where SA-basic behaves like HG. The illustration shows that our problem structure exhibits *disconnected feasible regions* based on minimal perturbations. It explains why long jumps are incorporated into SA algorithm in Section 3.2.2.

Consider the same scenario above, but now $L > 0$ so that the empty portfolio is itself infeasible. In this case, finding an initial feasible portfolio to seed the search is non-trivial. This motivates the two-phase structure of the SA algorithm in Section 3.2.2, where Phase 1 employs long jumps more aggressively to look for a good and feasible initial solution to seed the Phase 2 search. The good performance of SA in Fig. 3.3b is enabled by the two-phase algorithm design with allowable long jumps.

We now present results for 15 correlated transactions. Fig. 3.3c compares the competitiveness of the SA and HG solutions as a function of L. Fig. 3.3d plots the fractions of runs in which SA and HG return a non-empty solution. The results show that even in the case of complicated correlated transactions and a low computational budget, SA has good performance and significantly outperforms HG.

3.3.2 Multiple-Round Simulation Results

The one-round simulation experimental results in the previous section confirmed the effectiveness and efficiency of the hybrid-greedy (HG) algorithm and simulated annealing (SA) algorithm, according to the portfolio values they achieve for objective function (3.1). Although these experiments are meaningful to examine how our portfolio optimization approach works with a set of candidate transactions, they are solely for one-round cases. To evaluate how it works in an actual systems environment, we present multiple-round experiments using discrete event simulations in this section. This experimental setup is closer to real-world scenarios than one-round simulations shown above and gives us interesting insights into our problem and methods.

We model the arrival of access requests using a separate Poisson process for each request and solve the portfolio optimization problems *at each round* to determine which transactions should be allowed. For the allowed transactions, we model the arrival of actual profits using one Poisson process for each transaction. The process starts only after the transaction is allowed, and the actual profit arrives only once per transaction. For any Poisson process, the time between each pair of consecutive events is exponentially distributed.

Unless otherwise stated, we use normally distributed profits with mean chosen uniformly randomly from [1.0, 2.5] and standard deviation chosen uniformly randomly from [1, 3.5], and a risk-invariant VO with parameter $\alpha = 0.01$ and initial capital 10. The λ parameter of exponential distributions for transaction arrival is set to 0.5 and that for profit arrival is set to 0.75, respectively. We present results that are averages of 1000 independent runs whose duration (# of rounds) is 250.

Fig. 3.4 shows average capitals at the end of multiple-round simulations with independent transactions using variance notion of risk (Definition 1). In this experiments, there are at

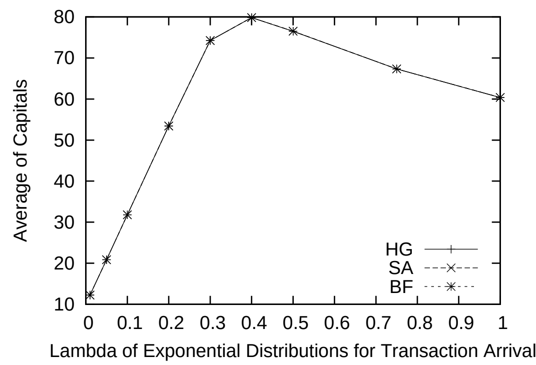
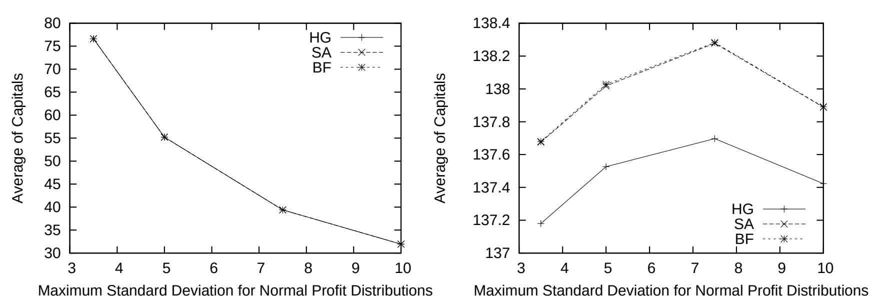


Figure 3.4: Average Capital with Independent Transactions: Maximum standard deviation for normal profit distributions is 3.5.

most 100 requests and arrived requests disappear immediately if they are not accepted. We used various λ values of exponential distributions for transaction arrival in this experiment. When large λ values are used, new transactions arrive quickly (i.e., waiting time is short), and thus we see more transactions in multiple-round simulations. We can see from this figure that there is a peak around $\lambda = 0.4$. In the region where $\lambda < 0.4$, capitals tend to be larger when λ value is larger. This trend is as expected because there are fewer transaction arrivals in simulations when λ value is smaller. However, in the region where $\lambda > 0.4$, capitals tend to be smaller when λ value is larger. This is because many transactions arrive at once in early rounds if λ is very large and because not all requests in each round cannot be allowed.



(a) Independent Transactions (b) Correlated Transactions

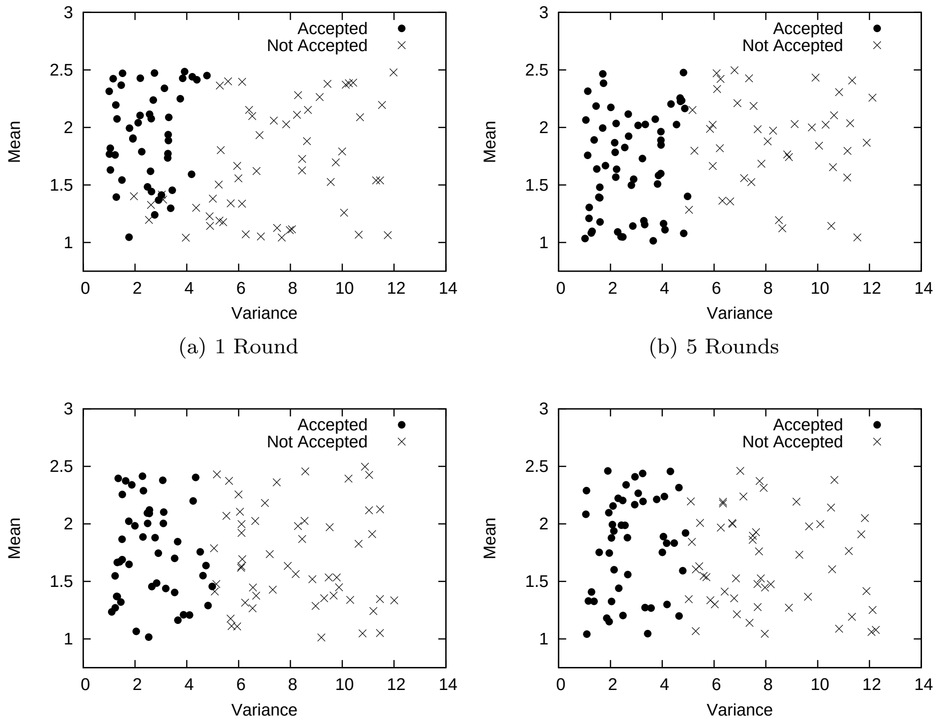
Figure 3.5: Average Capital: λ of exponential distributions for transaction arrival is 0.5.

Fig. 3.5a and 3.5b show average capitals at the end of multiple-round simulations using variance notion of risk (Definition 1) with independent and correlated transactions, respectively. We used various maximum standard deviations for normal profit distributions in this experiment. When large maximum standard deviations are used, costs for transactions are expected to be large. In the case of independent transactions shown in Fig. 3.5a, the capitals at the end of the simulations become smaller when larger maximum standard deviations are used, which is just as expected. In the case of correlated transactions shown in Fig. 3.5b, however, the same trend does not necessarily hold. This is because we can enjoy some hedging effects of negatively correlated transactions in this case. We saw the same trend when tail regime notion of risk (Definition 2) was used.

In multiple-round simulations, the brute force approach (BF in the graphs) still produces the best results *on average*. However, we saw some runs of multiple-round simulations in which other methods such as the simple greedy algorithm resulted in better capitals than the brute force approach. This is because our optimization problem is solved only *at each round* of multiple-round simulations and we don't consider the optimality in the overall simulation rounds. Because of this, it's possible that we use up all budgets at some point of a simulation, precluding the acceptance of future transactions even if they are more attractive with larger benefits but smaller costs than previously allowed ones. This is a generic online problem that occurs in many contexts. It's possible to incorporate future transaction arrivals into our optimization problem if we can assume that there is some characterization of future transaction arrivals such as a statistical model. There are some standard prediction methods such as ARIMA [53]. Although this is an interesting issue that is worth exploring, it is out of the scope of this thesis.

Heterogeneous Transaction Scenarios

With multiple-round experiments, we expect to make observations about whether there are starvation or liquidity problems. In our context, liquidity means an information sharing system's ability to facilitate a reasonable access request being accepted quickly. For that, we create heterogeneous transaction scenarios, e.g., a high-value, a high-risk transactions



(a) 1 Round (b) 5 Rounds

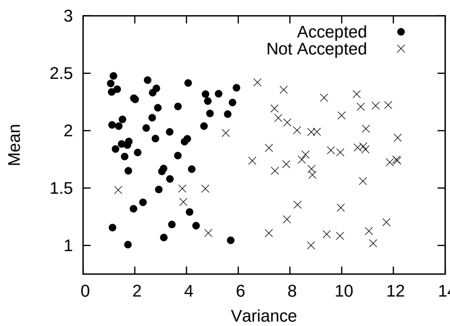
(c) 10 Rounds (d) 15 Rounds

Figure 3.6: Independent Transactions with Variance Notion of Risk: The x-axis and y-axis are the variance and mean of a transaction's profit distribution, respectively. Each transaction can stay in the system up to 1, 5, 10, and 15 rounds, respectively, as illustrated in (a) through (d). The budget is set to 5.

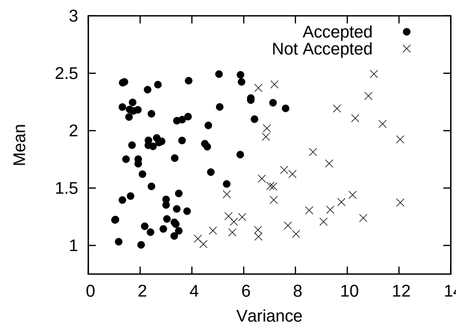
competing among many low-value, low-risk ones, etc. Under such conditions, we examine if the high-value, high-risk transaction will get a chance to run eventually, if it depends on how long that transaction can stay in the VO's information sharing program, and so on. Liquidity may also depend on how quickly the transactions realize their profits.

If they all realize their profits quickly, the turnover will be fast, and there may be some implications for the viability of the VO's information sharing program, as measured by the rate of growth of the VO profits attributable to information sharing. Thus, we also consider how different mixes of transactions (e.g., many short lived ones, vs. mostly sluggish ones) can affect the VO.

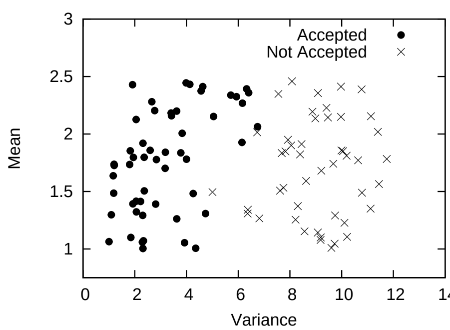
We set up our simulator so that transactions that are not accepted can stay in the



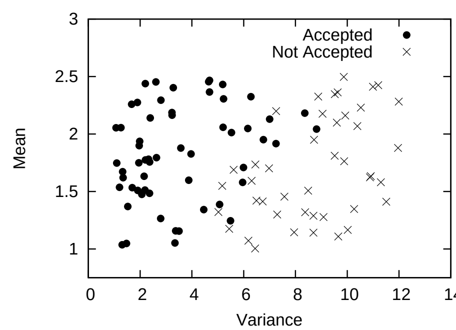
(a) 1 Round



(b) 5 Rounds

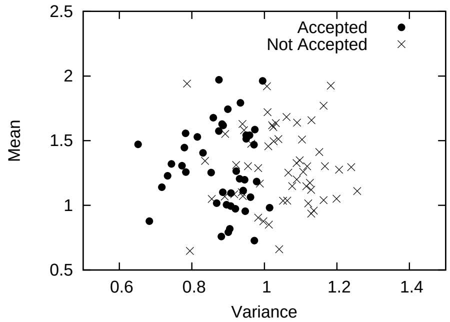


(c) 10 Rounds

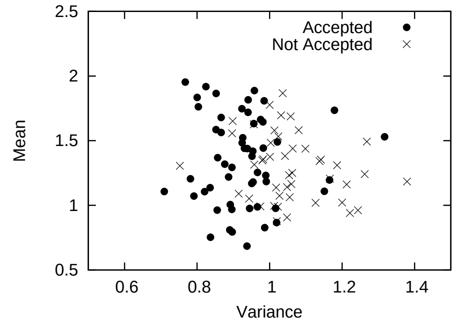


(d) 15 Rounds

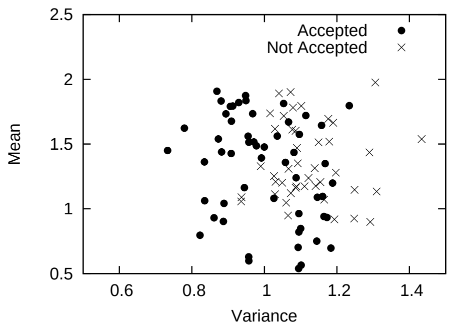
Figure 3.7: Independent Transactions with Tail Regime Notion of Risk: The x-axis and y-axis are the variance and mean of a transaction's profit distribution, respectively. Each transaction can stay in the system up to 1, 5, 10, and 15 rounds, respectively, as illustrated in (a) through (d). The budget is set to 5.



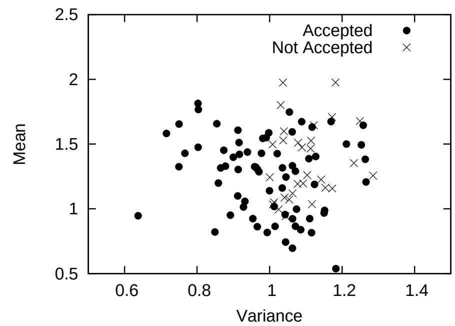
(a) 1 Round



(b) 5 Rounds

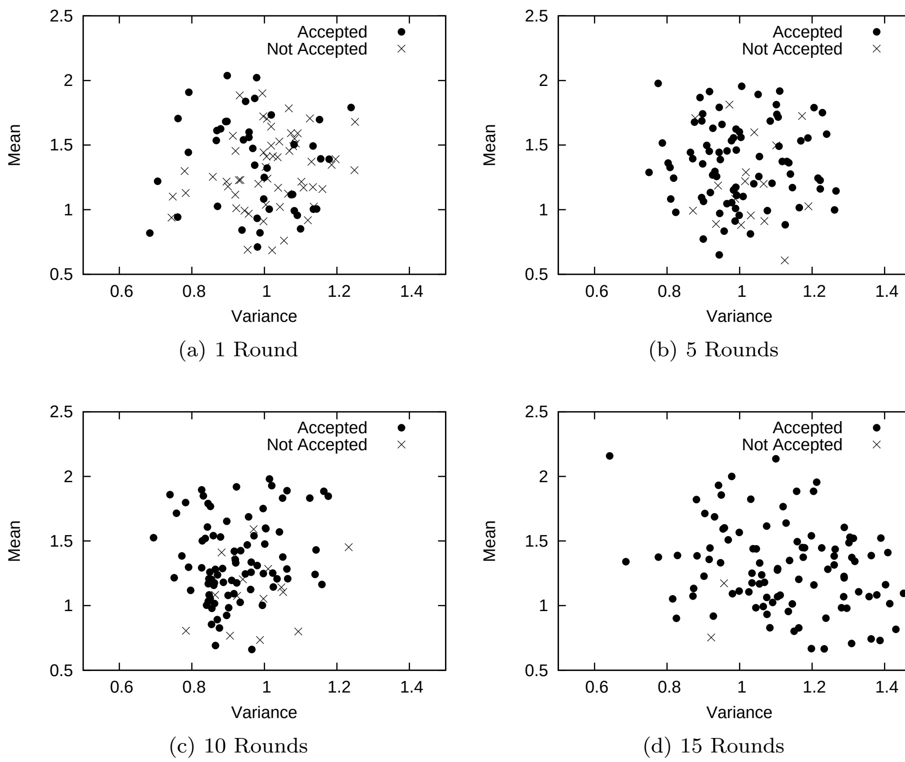


(c) 10 Rounds



(d) 15 Rounds

Figure 3.8: Correlated Transactions with Variance Notion of Risk: The x-axis and y-axis are the variance and mean of a transaction's profit distribution, respectively. Each transaction can stay in the system up to 1, 5, 10, and 15 rounds, respectively, as illustrated in (a) through (d). The budget is set to 1.



(a) 1 Round

(b) 5 Rounds

(c) 10 Rounds

(d) 15 Rounds

Figure 3.9: Correlated Transactions with Tail Regime Notion of Risk: The x-axis and y-axis are the variance and mean of a transaction's profit distribution, respectively. Each transaction can stay in the system up to 1, 5, 10, and 15 rounds, respectively, as illustrated in (a) through (d). The budget is set to 1.

system for some additional rounds. The maximum number of rounds for which transactions can stay in the system is a parameter in this experiment. We use 1, 5, 10, and 15 for this parameter and see how this value affects results. Note that value 1 for this parameter means transactions that are not accepted leave the system immediately, and value 5 means such transactions can stay in the system for 4 additional rounds, and so on. The budget is set to 5 for independent transactions, and 1 for correlated ones. The values of the other parameters are the same as before.

Figures 3.6 through 3.9 are scatter plots showing which transactions are accepted. We can see a trend that the longer transactions stay in the system, the more likely high-cost/high-return transactions can be allowed.

Independent Transactions with Variance Notion of Risk Because of the definition of the variance notion of risk and because transactions are independent, no transactions whose variances are bigger than 5 can be allowed.

Independent Transactions with Tail Regime Notion of Risk Because the standard deviations of component transactions used in the definition of the tail regime notion of risk are not additive even if the transactions are independent, some transactions whose variances are bigger than 5 are allowed and the number of such transactions increases as the maximum number of rounds for which the arrived transactions can stay in the system until they are accepted increases.

Correlated Transactions with Variance Notion of Risk Even with the definition of the variance notion of risk, some transactions whose variances are bigger than 1 are allowed because transactions are correlated and we can enjoy some hedging effects caused by negatively correlated transactions. The number of such transactions increases as the maximum number of rounds for which the arrived transactions can stay in the system until they are accepted increases.

Correlated Transactions with Tail Regime Notion of Risk Because of the definition of the tail regime notion of risk and also because transactions are correlated, some transactions whose variances are bigger than 1 are allowed and the number of such transactions increases as the maximum number of rounds for which the arrived transactions can stay in the system until they are accepted increases.

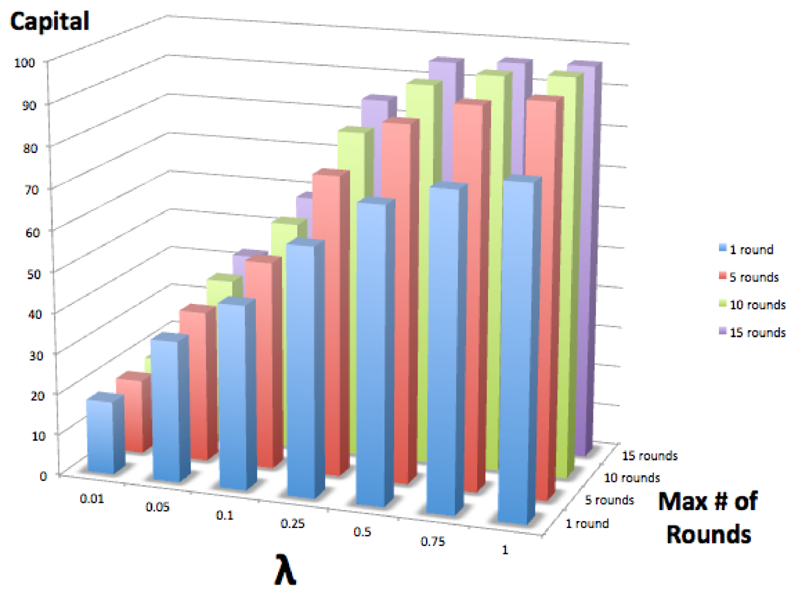
We also examine how liquidity depends on how quickly the transactions expire, i.e., how quickly the transactions' profits materialize. For that, we varied the λ value of the exponential distributions used to model the waiting time until transactions' profits materialize as well as the maximum number of rounds for which unaccepted access requests can stay in the system. Smaller λ means it takes longer until profits materialize.

The results are shown in Figures 3.10 and 3.11. The x-axis is λ, the y-axis is the maximum number of rounds, and the z-axis is capital.

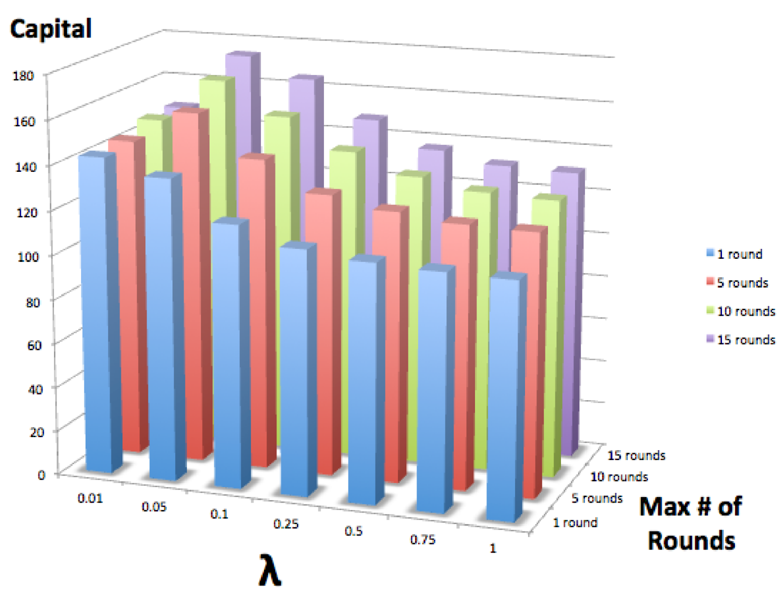
Regarding the number of rounds, the VO's capital increases as the number of rounds increases. This is because there are more chances to allow transactions when they stay in the system longer.

Regarding λ, the VO's capital increases as λ value increases when the variance notion of risk is used. This is reasonable because with large λ, i.e., profits materialize quickly, there are not so many transactions holding allocated risk budgets in the system. When the tail regime notion of risk is used, however, the VO's capital decreases as λ value increases, except when $\lambda = 0.01$. This is because of the definition of the tail regime notion of risk, i.e., $c \times \sigma[X] - E[X]$, in which standard deviation is used and it's not additive even if the transactions are independent. With the parameters used, the effect of $-E[X]$, which is additive, is stronger than that of $c \times \sigma[X]$. Thus, with small λ, i.e., transitions are sluggish and do not free allocated budgets for a long time, the VO can allow relatively many transactions.

Another thing we noticed is that during multiple-round simulations, costs can be bigger than the budget (5 for independent transactions and 1 for correlated ones). This is because costs can increase when transactions expire, i.e., transactions' profits materialize, even though costs are less than the budget when the transactions are accepted because of correlations of transactions and the definition of the tail regime notion of risk. Costs never exceed the budget when transactions are independent and the variance notion of risk is used.

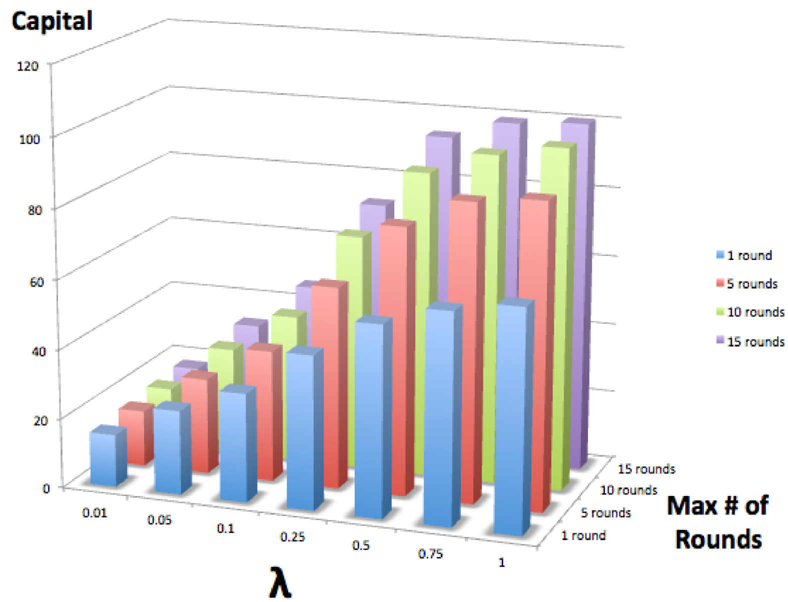


(a) Variance Notion of Risk

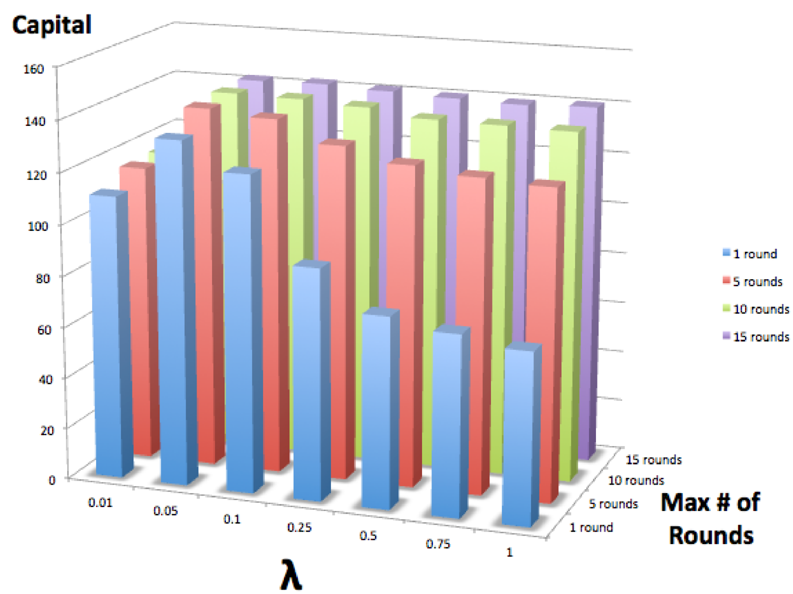


(b) Tail Regime Notion of Risk

Figure 3.10: Independent Transactions: The x-axis is λ, the y-axis is the maximum number of rounds, and the z-axis is capital. The budget is set to 5.



(a) Variance Notion of Risk



(b) Tail Regime Notion of Risk

Figure 3.11: Correlated Transactions: The x-axis is λ, the y-axis is the maximum number of rounds, and the z-axis is capital. The budget is set to 1.

Chapter 4

Insured Access

In this chapter, we show how we can encourage information sharing by protecting information producers against the harm that may come to them from sharing their information through an insurance scheme [82]. In recent decades, the business community has benefited from the use of actuarial methods to manage many kinds of business risks, but information sharing has not been among them. We address this gap by insured access.

4.1 Overview of Insured Access

Under insured access, a VO sets up an independent insurer that will manage the risk associated with accesses to shared information. In Figure 4.1, a producer Alice has information that can be shared with others. To obtain access, consumer Bob asks the insurer for a policy covering the specific information he wants to access (1). The policy will insure Alice against damages she might incur because she shared that information with Bob. If the insurer is

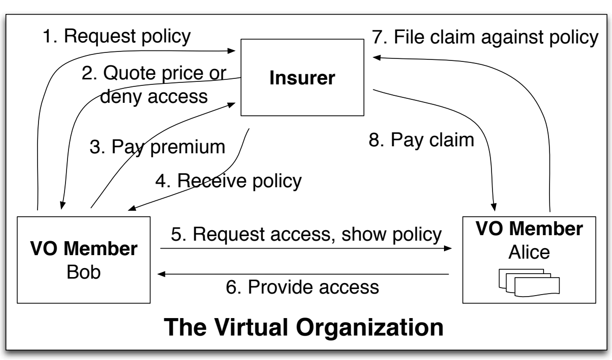


Figure 4.1: System Model for Insured Access

willing to issue the policy, it gives Bob a price (2), which Bob can choose to pay (3). We assume *consumers are rational, self-interested, and risk-adverse*, so Bob will only buy the policy if its price is less than the benefit he expects to gain from accessing the information. With the policy in hand (4), Bob asks Alice for access (5). Alice examines the policy and can give Bob the information (6). If Bob misuses the information and Alice suffers damages as a result, Alice can submit a claim to the insurer (7) and be reimbursed for her suffering (8). The policy can specify conditions of use, such as secure handling precautions or limits on the purpose of use. If Bob may have violated those rules, then the insurer can request reimbursement from Bob of the amount paid to Alice.

Insured access uses real money directly instead of relying on risk tokens. Cash can be coupled with decision theory to decide how much an access might benefit an information consumer or the VO [63]. Decision theory can show whether the likely rewards associated with a particular access exceed its potential risks as we do in our portfolio optimization approach explained in the previous section.

There are some obvious candidates for who will be using this decision theory:

- Individual employees of a VO member organization, who have been empowered (perhaps within limits) to request access to shared information;
- the VO member organization level;
- the VO level. This might be appropriate for a tightly knit VO with centralized control over many decisions, where timely access is not critical.

Regardless of where the decision to request access is made, it may be subject to oversight and checks and balances not discussed in this chapter. For example, requests from individual employees might be sent through a portfolio optimization process at the VO member organization level, or even at the level of the VO itself.

In practice, though, a consumer may be unwilling to share detailed information about its planned use of the information, making it hard to do decision-theoretic analysis of risks and rewards at the VO level. In that case, portfolio optimization could be omitted or decentralized and restricted to the VO member level. In that case, maximizing the overall

VO benefits is a secondary concern, with the consumers focusing on their own expected benefit and the insurer focusing on limiting harm to producers.

The VO can either use an external insurer or self-insure, but we consider only the case of self-insurance because the VO will have to expose very sensitive information to the insurer. Before insured accesses begin, the VO members set up an internal VO insurance group (the *insurer*), agree on what kinds of information to share, and decide how much risk the insurer can assume. The VO members who become information consumers supply the startup capital for the insurer.

Insurance-based access control is a new idea in the security world and differs from the usual liability insurance, under which asset owners (i.e., providers) buy policies to protect themselves. With insured access, consumers pay for policies that protect providers. This means that consumers must consider potential damages when deciding whether to access information, as well as benefits; and providers need not consider damages when deciding whether to share. Because of this difference, existing actuarial methods cannot be directly applied to purchase decisions in insured access, and we propose appropriate formulas for this task.

One tricky thing with access to information, which is the main kind of shared resource we consider in this thesis, is that it can be hard to prove that damage was done, or to prove that the damage was caused by one particular thing. In contrast, if someone hits a car, or a fire shuts down a business, it is usually obvious what the immediate cause was, though tracing it back can be hard. In the case of information, recognizing the damage and tracing it back could be even harder. The difficulty might mean that although the insurance is there, it could be challenging to get a claim reimbursed, or to get an appropriate amount of damages awarded. We recognize this difficulty, but insured access is not intended for every situation; rather, it is for the cases where the VO sees a clear need and a relatively clear path to modeling the risk and adjudicating claims.

For instance, if a student organization wants after-hours access to a printer room using the school ID card of the organization's publicity chair, the department might have a clear understanding of the risks of wear, tear, and need for repairs to the printer, and a clear understanding of the costs of various repairs, and might issue a policy for that. If the student organization abused the printer, then renewal of the privilege next semester or next year or next month would cost a lot more. If they were very responsible, then the price of the associated policy would probably go down over time. This example and, more generally, the historical record of civilization suggest that there are good usage scenarios out there for insured access to information, because information is at heart just another kind of resource. But insured access is not the right approach for every situation.

The price Bob pays for a policy is called the *premium*, which depends on how risk-averse the insurer is, what information Bob wants to access, the history of past claims for that information, and the insurer's *premium principle* (pricing methodology). Based on Bob's track record, the insurer can reward Bob with lower premiums if Bob's other policies have no recent claims, and penalize him otherwise. The insurer can refuse to issue a policy to Bob because of his track record, or because the insurer needs to limit the risk associated with its current portfolio of policies.

If the VO members are very unlucky or the data used to set prices is insufficient, then the insurer might run out of funds to pay claims and be ruined. The insurer guarantees that at the moment it issues a policy, the new policy does not make the chance of eventual ruin cross a threshold ε agreed upon in advance by VO members. The insurer must continually review and update the data used to set prices, to ensure that it reflects the latest claims statistics. To estimate likely claim amounts for rare events that it has never seen, i.e., *tail events*, the insurer can use *extreme value theory* [50]. The insurer can also use stop-loss policies and reinsurance to limit its exposure to tail risk ([57], pp. 8-13) . We discuss them in detail later in this chapter.

If the insurer's chance of eventual ruin ever exceeds ε in spite of these precautions, the VO members have several options. They may choose to supply additional capital reserves, as Lloyd's of London does. They may choose to take no action, so that the insurer runs out of money if its luck remains poor, necessitating internal loans, unpaid claims, or delayed claim payments. The insurer may raise premiums to increase capital, as homeowner insurance companies often do after a major disaster. Or it may transfer some of its risk to a reinsurer.

Each VO member, and the VO as a whole, can expect to benefit from insured access, for the following reasons. A consumer can use decision theory to determine its expected financial benefit from accessing a piece of information. A consumer expects a net benefit from each insured access, as otherwise it would not buy the policy. Providers' losses attributable to sharing are reimbursed by the insurer's payments on their claims. So that providers directly benefit from sharing, an additional fee that goes directly to the provider can be included in each premium, to cover the provider's additional costs attributable to sharing, plus a profit. A second, complementary, and more conservative option is for the premium to include a fee retained by the insurer and intended as *future* profit for the provider. Then at the end of each fiscal period, the insurer's excess capital reserves can be shared with providers. If desired, the same two fee-based approaches can be used to cover the costs of running the insurer, or give it an expected profit; in this thesis, we assume that the insurer is non-profit and cost-free. We also assume that the benefits and damages attributable to sharing accrue only to the provider and consumer, without impact on other VO members, though insured access can also be used in more complex situations with potential collateral benefits and damages. Finally, as discussed later, the VO must set up the system so that each act of sharing is expected to benefit the VO as a whole, i.e., the shared purpose that binds its members together.

4.2 Details of Insured Access

4.2.1 How to Price Policies

Consider a particular insured access. The insurer does not know in advance what the total size of all claims on that policy will be, but it can represent this quantity by a random variable X that represents the *risk* associated with a particular consumer accessing a particular piece of information owned by a particular producer. More precisely, X represents the total amount of the claims that will eventually be paid to the producer under that policy. For now, let us assume that the insurer knows the probability distribution for X, and discuss how this information can be used to set the price of the policy.

Table 4.1: Common Utility Functions.

Name	Utility Function	Risk Aversion Index	Restriction
Linear	$u(z) = z$	$r(z) = 0$	
Quadratic	$u(z) = z - \frac{z^2}{2c}$	$r(z) = \frac{1}{c-z}$	$(c > 0$ and $z \leq c)$
Logarithmic	$u(z) = \log(z)$	$r(z) = \frac{1}{z}$	$(z > 0)$
Exponential	$u(z) = 1 - e^{-\alpha z}$	$r(z) = \alpha$	$(\alpha > 0)$
Power	$u(z) = z^c$	$r(z) = \frac{1-c}{z}$	$(0 < c \leq 1$ and $z > 0)$

The most widely adopted approach to pricing risk in general is the *Principle of Equivalent Utility*, in which the premium P is calculated by solving the following equation ([57], p. 4):

$$E[u_I(w_I + P - X)] = u_I(w_I). \tag{4.1}$$

Here u_I is the utility function of the insurer I, and w_I is its current capital. This principle says that the premium P should be set to the amount at which the insurer is equally happy whether or not the policy is issued, i.e., indifferent.

The exact formula to calculate P depends on the utility function $u(z)$ chosen for u_I in Formula 4.1, where z represents capital. Table 4.1 shows several well-known utility functions. Most are concave for $z > 0$, to reflect the diminishing marginal utility of money. Given $u(z)$, we can derive the *risk aversion index* $r(z)$ of a principal, which is defined as in the previous chapter: $r(z) = -\frac{u''(z)}{u'(z)}$. More risk averse principals have more concave utility functions.

With a linear utility function, $u(z) = z$ and $r(z) = 0$, meaning the principal is risk-neutral. Let π denote a premium pricing principle. If we use a linear utility function in the Principle of Equivalent Utility in Formula 4.1, we get the $\pi[X] = E[X]$, which is knowns as the *Net Premium Principle* in the actuarial literature. With the Net Premium Principle, the insurer sells a policy for the expected amount of its claims. In the long run, an insurer would tend to break even with the Net Premium Principle. However, in practice insurers usually prefer to set prices higher than the premium calculated by the Net Premium Principle, because the premium by the Net Premium Principle requires high capital reserves to avoid ruin.

With an exponential utility function $u(z) = 1 - e^{-\alpha z}$, for $\alpha > 0$, the risk aversion

Table 4.2: Premium Principles ($\alpha > 0$).

Name	Premium Principle
Net premium principle	$\pi[X] = E[X]$
Expected value principle	$\pi[X] = (1+\alpha)E[X]$
Variance principle	$\pi[X] = E[X] + \alpha Var[X]$
Standard deviation principle	$\pi[X] = E[X] + \alpha\sigma[X]$
Exponential principle	$\pi[X] = \frac{1}{\alpha}\log(m_X(\alpha))$

index $r(z) = \alpha$, which is a constant, although $r(z)$ is a function of z in general. Using the exponential utility function, we derive the *Exponential Principle* from the Principle of Equivalent Utility in Formula 4.1:

$$\pi[X] = \frac{1}{\alpha}\log(m_X(\alpha)), \tag{4.2}$$

where $m_X(\alpha) = E[e^{\alpha X}]$ is the moment generating function of a random variable X around α, and X represents the total claims associated with a policy. For this or any other principle, we could add an additional fee to $\pi[X]$ to support the provider or insurer, as discussed previously.

Table 4.2 summarizes these and other basic premium principles, where $\alpha > 0$. Although there are pros and cons for each principle, the Exponential Principle is particularly widely used in the actuarial literature [87]. Among its favorable properties [86], the following two are especially important:

Additivity for independent risks. $\pi[X+Y] = \pi[X] + \pi[Y]$, where X and Y are independent.

Superadditivity for positively correlated risks. $\pi[X+Y] \geq \pi[X] + \pi[Y]$, where X and Y are positively correlated.

For independent risks from separate acts of sharing, the Exponential Principle's additivity means that the price for one policy covering all of them is the same as the total price for separate policies for each. This means that an insurer can price a new policy without having to analyze its aggregate risk across all its policies, and significantly reduces the complexity of the problem in the case of independent risks, as it is easy to derive the distribution for the aggregate risk in the insurer's portfolio of policies.

With a non-additive premium principle, the insurer cannot easily price a policy for a new insured access request. Instead, it must compute the aggregate risk of its entire existing risk portfolio, i.e., the probability distribution of total claim sizes for all policies already issued, plus the proposed new policy, and use that to set the premium. If there are multiple new requests, their premiums will depend on the order in which they are calculated, because a new policy issuance affects the premiums of subsequent policies. Thus the moment of issuance could significantly affect the price of a policy, which consumers are likely to regard as unfair. The Exponential Principle avoids this complication.

Superadditivity for positively correlated risks is important because allowing several instances of information sharing can introduce a much larger risk than just the sum of each risk, if they are positively correlated. For example, military phone books are often classified, even if each number in them is unclassified. As another example, the risk from sharing the maps of an entire gas pipeline may be much greater than the sum of the risks from sharing the maps of each part of the pipeline. As an extreme example, the risk from telling someone the kth bit of a cryptographic key is quite low. But if all bits of the key are shared in this manner, the recipients can collude to reconstruct the key, which may be catastrophic. The Exponential Principle is one of just a few premium principles with superadditivity for positively correlated risks, because superadditivity is not needed for the most common types of insurance (e.g., houses, cars, medical, and traditional liability). The Exponential Principle also enjoys subadditivity for negatively correlated risks, i.e., it reflects the fact that diversification can reduce overall risk. Because of these favorable properties, we use the Exponential Principle in the remainder of this thesis.

Thus, given an access request (risk) X, the insurer uses Formula 4.2 to compute a premium, tacking on a fee to go directly to the provider if desired.

4.2.2 Tail Events, Ruin, & Reinsurance

The insurer groups similar risks into a *class*, as discussed in detail later, such that all risks X in the class follow the same probability distribution, and assigns the same premium to all risks in the class. The previous section assumes that the insurer knows this distribution. Actually, the insurer may only know its own history for the class, consisting of the details of every policy it issued (including date, producer, consumer, information, premium, and class) and every claim it received (policy, date, amount). From this data, the insurer can compute how many policies in the class have had total claims of no more than $\$K$, for each value of K. Dividing by the total number of claims produces an approximation to X's distribution. Often, X is known to belong to a particular family of distributions. In that case, the claims history can be used to estimate the parameters of the distributions, using maximum-likelihood estimation [37], including the popular expectation-maximization approach. In general, the longer the claims history, the better the approximation will be. With an extremely long history, one might expect a very good approximation.

Unfortunately, in the real world, damage-causing events have an extremely long-tailed distribution, where the tail includes many highly unlikely catastrophic events. A simple real-world example would be the chance of two vicious hurricanes in one year, which the major insurers had not experienced in the historical record and did not include in their models until recently. Another example would be the engineering of nuclear power plants to meet the worst dangers previously experienced. Each new disaster makes us retrofit existing plants to consider previously unseen tail risks. However, in the auto insurance industry, for example, probably the historical record regarding accidents includes relatively few unseen tail events, because there are so many cars out there and they are driven for so many miles, even though the industry is only about a hundred years old.

A claims history is a finite random sample from this long-tailed distribution. No matter how long the class history is, if an insurer prices premiums solely based on the history and without considering the unseen long tail, the insurer eventually faces ruin ([57], pp. 87-111). In other words, *no matter how long the class history is*, with high probability, if an insurer prices premiums based on the history, without considering the unseen long tail, the insurer will be ruined ([57], pp. 87-111). Using a longer history only delays the expected time to ruin, and does not prevent it.

The closest one can come to addressing the problem of unseen tail risks is the work in the area of economics and finance on estimating the magnitude of unseen tail risks. For instance, *extreme value theory* can help by providing a basis for statistical modeling of unseen tail events [50]. According to extreme value theory, the cumulative distribution function of aggregate claims above a certain threshold, which is called the conditional excess distribution function, will be well approximated by a generalized Pareto distribution. Still, extreme value theory only approximates the true risk distribution. This is a real-world problem for actuaries, but it is probably not a killer problem for many cases where a VO might like to share info.

To handle the risk of high-damage events it has never observed, an insurer can buy a stop-loss insurance policy for the class ([57], pp. 8-13) from a reinsurer. The insurer pays claims as usual until the total payout exceeds a threshold d specified in the stop-loss policy, e.g., 150% of X's expected value; the reinsurer pays subsequent claims. The stop-loss policy transfers tail risks to the reinsurer and lowers the variance of the insurer's portfolio. Reduced variance means that the insurer's profits are more predictable, which is helpful if the producers' benefits from insured access come from a profit-sharing scheme. Compared to the many other forms of reinsurance that the insurer could buy for the same price, stop-loss is provably optimal for reducing the variance of the insurer's claims (Theorem 1.4.3, [57], p. 11), when the reinsurer uses the Net Principle.

The reinsurer pools many risks, so a highly unlikely catastrophic event that the insurer has never experienced may be an everyday occurrence for the reinsurer. The threshold d can be chosen to maximize the insurer's expected utility at the end of a fiscal period [36].

The VO may not want to expose its inner workings to a reinsurer to obtain stop-loss coverage. If the VO does not purchase a stop-loss policy, its members must understand that if never-before-seen catastrophic events occur, the insurer's claims may exceed its capital.

Once the VO members have set a bound ε for their insurer's chance of eventual ruin and decided how risk-averse (α) the insurer is, then *ruin theory* ([57], pp. 87-111) specifies the minimum capital the insurer needs to keep the chance of ruin below ε.

In a traditional ruin theory, an insurer's cash balance is modeled as the net effect of two cash flows on its startup capital, namely, the premiums paid by consumers and the damage claim payments made to producers. This stochastic process is defined as follows:

$$W(t) = w_I + ct - S(t), t \geq 0,$$

where $W(t)$ is the insurer's capital or wealth at time t, $w_I = W(0)$ is the initial capital, c is the constant premium income per unit of time, and $S(t) = X_1 + X_2 + \cdots + X_{N(t)}$ with $N(t)$ being the number of claims up to time t, and X_i being the size of the ith claim.

An important point to remember is that the classical ruin theory assumes that the premium income is constant at the rate of c per unit of time and that the insurer incurs a sequence of claims that are mutually independent and identically distributed with a common distribution function. The arrival of claims is assumed to follow a Poisson process.

Under these assumptions, we can derive the following inequality for the ruin probability $\Psi(w_I)$:

$$\Psi(w_I) \leq e^{-Rw_I},$$

where w_I is the initial capital and R is a parameter called the adjustment coefficient.

In the case of an exponential utility function, R is just its risk aversion index α that leads to an annual premium c. Thus, $\Psi(w_I) \leq e^{-\alpha w_I}$, or

$$\varepsilon = e^{-\alpha w_I}, \text{ i.e., } \alpha = \frac{1}{w_I} |\log \varepsilon|, \tag{4.3}$$

where α is the insurer's risk aversion index, w_I is its initial capital, and ε is the upper bound on ruin probability. Even if the insurer does not experience tail events, the chance of ruin may approach ε due to bad luck. For example, if there is a .001 independent chance of rain each day, it can still rain for ten days in a row. Given its current capital and α, an insurer can apply Formula 4.3 periodically to determine whether the current upper bound ε' on the chance of ruin is still below ε, and work to increase capital if not (reinsurance, higher premiums, capital injection, tighter rules on what can be shared and for what purposes).

Even when the models are correct, the insurer still might go bankrupt due to extreme bad luck, but the VO can cap the chance of that happening, at a level that the VO is comfortable with. The banking industry provides a real-world example of how to cap the risks. If all depositors want their money back from a bank, they might not get it back, because the bank's investments may have gone bad or might not be liquid. The bank regulators have adopted certain thresholds and related rules to help ensure that banks remain solvent and can meet the daily needs for cash. Banks do fail anyway, so bank regulators periodically inspect banks to detect the approach of failure and prevent it, and nations have the equivalent of the US Federal Deposit Insurance Corporation (FDIC) to help out when a failure does occur.

For correlated risks or unsteady premium income, the insurer will need to perform lengthy simulations to determine the chance of ruin, as discussed later.

4.2.3 Defining Classes of Risks

If the insurer does not know the probability distribution of a new risk X, it cannot use Formula 4.2 to set the premium. If risks are correlated but X's distribution is known, then the insurer can still use Formula 4.2 to set the policy premium for a new access request. However, Formula 4.3 no longer applies, so issuing the policy might push the chance of ruin above ε. To ensure that this does not happen, the insurer can run simulations to compute the probability of ruin, but this is a very lengthy process. Another potential problem is that after estimating the parameters of a distribution for observed claim data, tests of goodness of fit (e.g., chi-square, Kolmogorov-Smirnov, Anderson-Darling) may report that the observed data is unlikely to have been drawn from the distribution it has been fitted to.

Preanalysis offers a solution to these problems. The VO members can identify all the classes of insured access requests that they might like to allow in the future. Similar to traditional access control, each class might be represented by a type of information (e.g., clusters defined according to resource type or other metadata), a group of VO members allowed to access this data (e.g., clustered by roles or other certified attributes), and constraints on the context (e.g., when, where, and for what purpose) under which such accesses are to be permitted. For example, map producers may be willing to share their Asia maps with any appropriately insured VO member.

The insurer must subdivide classes until all risks (i.e., expected total policy claims) in the same class fit the same probability distribution, and then use that distribution to determine the (identical) premium for all policies to be issued in that class. Subdividing a class can be approached as a mixture model problem [45], where subclasses all belong to the same family of distributions, but with different parameters. Expectation maximization is popular for creating mixture models. Real-world claim sizes for many kinds of policies are exponentially distributed, so subdivision is not as daunting as it might sound. Without subdivision, high-risk and low-risk sharing may be lumped into the same class, and the insurer cannot give lower premiums to the former group. Then the low-risk policies will subsidize the high-risk ones, as if careful and accident-prone drivers had to pay the same amount for auto insurance. Since the VO's insurer is a monopoly, it need not fear *adverse selection*, where all the good risk takers move to an insurer that offers them lower premiums. But good risk takers will demand a fairer approach that gives them a class of their own, with lower expected claim sizes and hence lower premiums. The insurer can also benefit, through more accurate prediction of future claims, thus lower chance of ruin and steadier profits. But the classes must not become so small that the claims history for a class is too small for statistical significance of tests of goodness of fit, i.e., there is not enough data to compute its distribution's parameters within a desired error bound.

The insurer must periodically check that recent claims history is consistent with what it expected, by rebuilding its probability distribution for historical claims data for a class, and looking for changes and trends that may suggest premium changes. To help with this task, the insurer can employ *actuarial credibility theory* ([57], pp. 203-227), which helps a model-builder extrapolate from a small sample that is highly relevant (recent history), by exploiting a large set of data that is not quite so relevant (the rest of history). Actuarial credibility theory is an offshoot of *credibility theory*, which is devoted to determining how to weight a recent track record against the longer historical data for a class of risk. As usual, the parameters of the resulting distribution determine how to set premiums, and its goodness of fit determines whether the class boundaries need to be adjusted.

Preanalysis also helps the insurer with the problem of ensuring that its chance of ruin will not exceed ε once it issues a new policy. With correlated risks, computing the chance of ruin requires lengthy simulations. Positively correlated risks can significantly increase the overall risk, while the aggregate risk of policies with negatively correlated claim sizes can be lower than the sum of their individual risks. Thus as discussed in the previous chapter, the insurer can use portfolio management theory from the financial engineering community to reduce its overall risk by diversifying the types of risk assumed. The insurer can analyze historical data to estimate the correlations between risks of different classes in advance, run simulations to estimate the chance of ruin given particular numbers of policies in each class, and use the simulation results to cap the number of policies sold in each class. Setting the caps is an optimization problem that we do not consider in this thesis. In practice, many risks will diminish over time, so that no new claims on a policy will be expected after a certain point; subsequently, the policy can be excluded or discounted when determining whether the cap has been reached.

4.2.4 Purchase Decisions

A consumer can use decision theory to determine its expected financial benefit from accessing a piece of information. Then the consumer must decide whether the benefit is worth the price of the policy. Because the consumer's benefit is uncertain and the consumer is risk-averse, it is too simplistic to buy the policy as long as the expected benefit exceeds the premium. Instead, consider the following inequality, which is similar to the Principle of Equivalent Utility:

$$E[u(w + Y - P)] \geq u(w), \tag{4.4}$$

where u is the consumer's utility function, w is its capital (or wealth) that it can use to buy policies or anything else, and Y is a random variable representing the consumer's expected additional value (or revenue) from accessing the information. Let the consumer have an exponential utility function with parameter α_c. Then from Formula 4.4, we can derive the maximum premium P^+ the consumer is willing to pay:

$$P^+ = -\frac{1}{\alpha_c}\log(m_Y(-\alpha_c)), \tag{4.5}$$

where $m_Y(-\alpha_c) = E[e^{-\alpha_c Y}]$ is the moment generating function of a random variable Y around $-\alpha_c$. Thus, the consumer buys the policy if the premium is no more than P^+, reflecting the expected benefit and the chance that he might be worse off after using the information. As noted earlier, traditional actuarial methods do not provide this sort of decision theoretic formula to compare policy prices with possible monetary benefits.

4.2.5 Rewarding Good Risk-takers

In the discussion so far, the insurer sets premiums based solely on the class of risk, i.e., the type of consumer, information, and circumstances of access. The VO can benefit by encouraging good risk-takers, i.e., consumers whose insured accesses do not result in claims, by giving them lower premiums. This is called a *bonus-malus system* ([57], pp. 135-146), a branch of credibility theory. Though much more sophisticated methods are available, we adopt the simple and effective Dutch system ([57], p. 136-138), still used for auto insurance in the Netherlands.

Table 4.3 shows the Dutch bonus-malus system. The system has 14 steps, each with its own weight, which is a discount factor to be multiplied with the policy price obtained by a premium principle. Consumers in step 8, for instance, need to pay only half of the original premium price. New insureds enter at step 2, with premium level 100%. These steps are updated at policy renewal, using the transition table at the bottom of Table 4.3. Consumers with no claims in the previous period ascend one step and get lower premiums, but those with claims filed against their policies descend several steps, resulting in higher prices. For access control, this scheme can be applied in a coarse-grained manner, i.e., to each user across all their policies; or to each user based on their track record for a particular class of information and/or under particular circumstances.

Table 4.3: Dutch Bonus-Malus System: Frequent claim filing results in unfavorable price weights.

Step	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Weight	1.2	1.0	0.9	0.8	0.7	0.6	0.55	0.5	0.45	0.4	0.375	0.35	0.325	0.3
0 claims	2	3	4	5	6	7	8	9	10	11	12	13	14	14
1 claim	1	1	1	1	2	3	4	5	6	7	7	8	8	9
2 claims	1	1	1	1	1	1	1	1	2	3	3	4	4	5
$\geq$ 3 claims	1	1	1	1	1	1	1	1	1	1	1	1	1	1

4.2.6 Bootstrapping the Insurer

In deciding what types of sharing will be allowed, VO members must be careful to align members' individual incentives with the shared purpose that brought the VO together, so that a consumer's benefit is also a benefit for the VO. Otherwise, sharing will benefit individual members but not necessarily advance the purpose of the VO. If incentives are aligned, then every act of sharing has an expected net positive benefit for the VO, no matter how the risks of different acts may be correlated. To help align incentives, the VO can offer an incentive scheme that rewards consumers with *wages* when their use of shared information benefits the VO as a whole. Molloy et al. [71] present an abstract model of a VO member's wage as a function of her own profit and the profit of all members, plus a base salary. The key for this scheme is how to choose the function so that making optimal decisions for the VO as a whole is in the member's best interest. With the right function, rational members will try to request insured accesses whose expected outcomes are aligned with the common goal of the VO.

The benefits directly attributable to insured access must be weighed against the opportunity cost to the members who contributed the insurer's capital reserves. As an extreme example, if no sharing takes place, then the opportunity costs far outweigh the non-existent benefits from sharing. However, if insured access is as popular and beneficial among VO members as they expected it to be when they set up the insurer, then the opportunity costs of capital reserves for consumers will be offset by their realized gains due to sharing. The VO can start with modest capital reserves, and add to them as insured access becomes more popular.

At startup, an insurer may have no historical claim information of its own. It may be able to use historical data from other organizations. If some relevant data is available for modeling a risk, but not enough for statistical significance, the insurer can use actuarial credibility theory to extrapolate from the relevant data plus a large set of slightly related data, to provide better risk estimates. With regard to rare events, the insurer can also use extreme value theory to obtain better estimates of the tail parts of risk distributions from available data. Even without historical data, actuaries manage to estimate future claim amounts for new classes of risks, including such exotic risks as alien abduction and damage to the legs of Heidi Klum, Michael Flatley, and Mariah Carey [1]. Thus we can assume that actuaries will be able to get the system off the ground.

At startup, the insurer must determine the fees and/or profit-sharing scheme for the producers. Perhaps the simplest profit-sharing approach is to wait until the end of a fiscal period, calculate the level of capital that the insurer must retain for probability of ruin ε, and distribute the excess capital among the producers. More sophisticated methods distribute the insurer's funds in excess of the *optimal dividend barrier*, which maximizes the total expected present value of the distributions (dividends) before ruin; there is a simple closed-form formula for the optimal dividend barrier under the common assumption that claim sizes follow an exponential distribution [48]. Once the barrier is set, VO members must decide whether to distribute the profit evenly or according to each producer's amount of sharing, risk assumed, benefit derived by consumers, or any other factor.

4.2.7 Techniques for Estimating Probability Distributions for Damages and Benefits

In this thesis, we are assuming that perfect information is available, i,e., we know the probability distributions of random variables for damages and benefits. However, obviously this is not the case most of the time. Thus, we need to estimate them through past experiences, and we present one way to do this from a historical record below.

For the sake of simplicity, we limit the following discussion to the case where damage $c_{i,j}$ occurs to producer P_i with probability $q_{i,j}$ and damage $c'_{i,j}$ occurs with probability

$(1 - q_{i,j})$ when consumer C_j purchases a policy to access the information. In other words, the expected damages to producer P_i when consumer C_j purchases a policy to access the information is represented by

$$q_{i,j}c_{i,j} + (1 - q_{i,j})c'_{i,j}.$$

$c_{i,j}$ will likely be a large value to reflect the damage to P_i for releasing information that results in a realized risk (intuitively, $q_{i,j}$ is the probability that a rare bad thing happens due to insured access to information). We try to estimate the probability $q_{i,j}$ from the historical record.

We denote the estimated values of $q_{i,j}$ after we observe past results k times as $q^k_{i,j}$. Before we observer the first result, we need to initialize $q^0_{i,j}$. If we have prior knowledge, we can set these initial values accordingly. If not, we may use arbitrary values such as 0.5. After estimating these probabilities $q^k_{i,j}$, we can use them as $q_{i,j}$ for the subsequent insured access. We show how to estimate $q^k_{i,j}$ below.

Suppose we observe that a bad thing happened to P_i r times and that it didn't happen $k - r$ times. Based on these observations, we estimate the probability $q_{i,j}$. The simplest way to do this is to use maximum likelihood estimation. We get the estimated probability

$$q^k_{i,j} = \frac{r}{k}$$

when we use maximum likelihood estimation.

For example, suppose the insured access occurred three times and the damages to P_i when C_j buys a policy for insured access are $c'_{i,j}$, $c_{i,j}$, and $c'_{i,j}$ (i.e., $c_{i,j}$ appears once and $c'_{i,j}$ appears twice). Then, $q^3_{i,j} = 0.33$.

Although maximum likelihood estimation is a reasonable way, it does not work so well if the number of samples (i.e., observations) is small. This is especially true if the actual probability $q_{i,j}$ is very small and thus we cannot observe that a bad thing happens during the repeated plays of the game. In such a case, if we use maximum likelihood estimation, $q^k_{i,j}$ becomes 0, which is obviously not what we want.

To address this problem, we can use smoothing techniques. All smoothing methods try to (1) discount the probability of phenomena (i.e., a bad thing happened or didn't happen) observed in the repeated plays of the game, and (2) re-allocate the extra counts so that unobserved phenomena will have a non-zero count. We show some of these techniques below.

- **Additive Smoothing** In this technique, a constant δ is added to the counts of each phenomenon. When $\delta = 1$, it is called Laplace Smoothing. The estimated probability with Laplace Smoothing is

  $$q_{i,j}^{k} = \frac{r+1}{k+2}.$$

  Here, 2 is added to k in the denominator because there are 2 types of phenomena (i.e., a bad thing happened or didn't happen) in an insured access. Generally speaking, we add the number of possible phenomena to the denominator.

  One problem of this technique is that all unobserved phenomena get equal probabilities, though this is actually not a problem when there are only two phenomena (i.e., a bad thing happened or didn't happen). We can use a reference model q_{ref} to discriminate unobserved phenomena. Some of the techniques explained below use a reference model.

- **Absolute Discounting** In this technique, a constant δ is subtracted from the counts of each phenomenon. The estimated probability with Absolute Discounting is

  $$q_{i,j}^{k} = \frac{max(r-\delta, 0) + \delta k_u q_{ref}}{k},$$

  where k_u is the number of unique phenomena in the repeated plays of the game.

- **Linear Interpolation, Jelinek-Mercer** This technique shrinks the probability uniformly toward q_{ref}. The estimated probability with Linear Interpolation (Jelinek-Mercer) is

  $$q_{i,j}^{k} = (1-\lambda)\frac{r}{k} + \lambda q_{ref},$$

  where λ is a parameter.

- **Dirichlet Prior/Bayesian** This technique assumes a pseudocount μq_{ref}. A pseudocount is an amount added to the number of observed cases in order to change the expected probability in a model of those data, when not known to be zero. The estimated probability with Dirichlet Prior (Bayesian) is

$$q_{i,j}^{k} = \frac{r + \mu q_{ref}}{k + \mu} = \frac{k}{k + \mu}\frac{r}{k} + \frac{\mu}{k + \mu}q_{ref},$$

  where μ is a parameter.

- **Good Turing** This technique assumes the total number of unobserved phenomena to be n_1 (the number of phenomena observed exactly once), and adjusts the observed phenomena in the same way. For each count r, we compute an adjusted count r^*:

$$r^* = (r + 1)\frac{n_{r+1}}{n_r},$$

  where n_r is the number of phenomena observed exactly r times. Then, the estimated probability with Good Turing is

$$q_{i,j}^{k} = \frac{r^*}{k}.$$

4.3 Simulation Experiments

In this section, we use discrete event simulations to confirm that the theory presented in the previous section correctly predicts the likely outcome of insured access in a simulated example scenario, and to understand the effect of different parameters on the outcome. Our results show that on average, the expected capital of each VO member, the insurer, and the VO as a whole does grow over time. We also examine the insurer's probability of ruin as a function of its degree of risk aversion. Testing the techniques for estimation of distributions from claims data is beyond the scope of this thesis, as real claims data is not available to us.

We have open-sourced the simulator and it is available on the author's GitHub repository under the name *IASimulator* [17]. Our simulator is written in C++, and uses the Boost

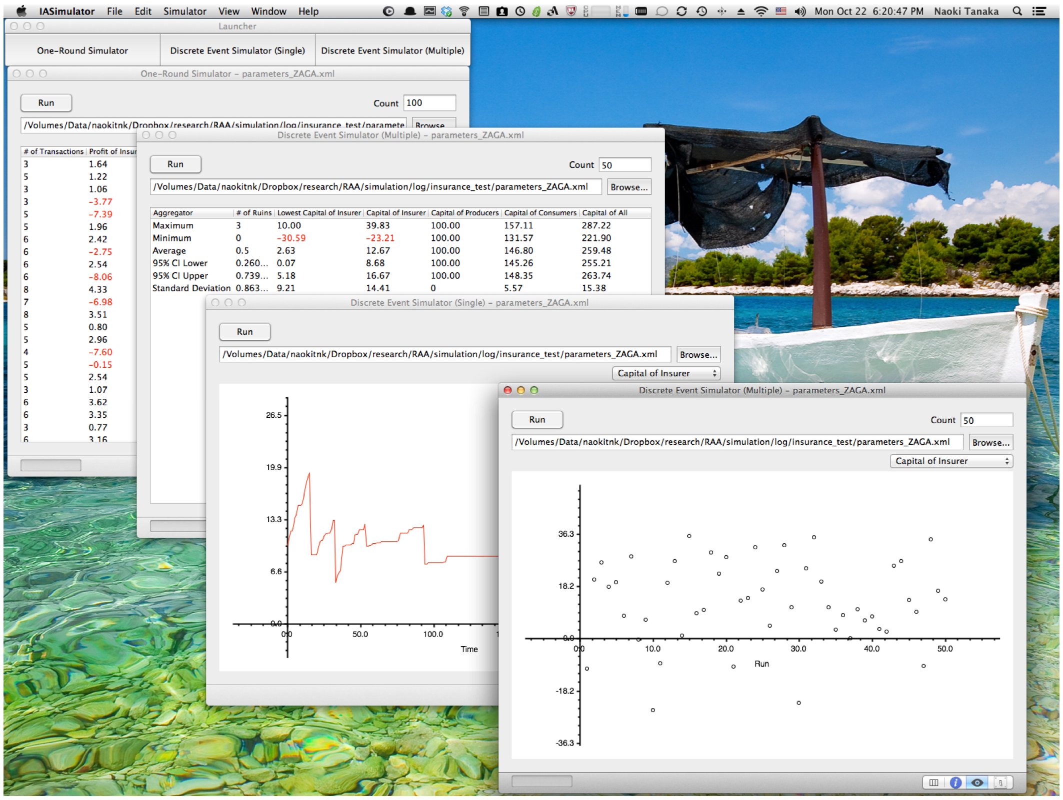


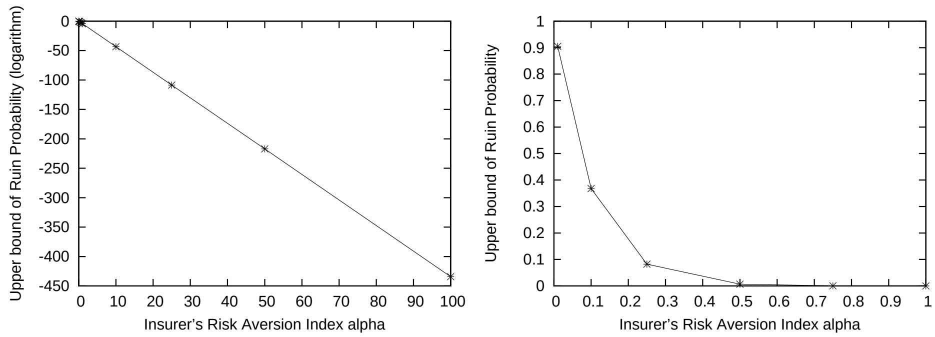
Figure 4.2: Screenshot of GUI for IASimulator

C++ Libraries [7] and their Math Toolkit. The simulator is a command line interface tool, but we have also developed a graphical user interface (GUI) for it so that we can quickly examine how simulation results change with various parameter values. This GUI tool still uses the same C++ logic, but its GUI parts are written in Objective-C and it can run on OS X. Figure 4.2 shows a screenshot of this GUI tool.

4.3.1 Experimental Setup

The simulations model a scenario where the VO is partitioned into producers and consumers. Each producer produces one map, which is unique across all producers. Some maps are more sensitive than others, depending on who the consumer is. This sensitivity is reflected in the parameters of their distributions of claim sizes, and thus in their premiums.

We model each insured access as an independent (uncorrelated) risk. We model the



(a) Logarithm of Probability ($0 < \alpha \leq 100$) (b) Probability ($0 < \alpha \leq 1$)

Figure 4.3: Upper Bounds on Ruin Probabilities: An insurer with a large risk aversion index has only a small probability of eventual ruin.

arrival of requests for insured access using a separate Poisson process for each consumer. For purchased policies, we model the arrival of claims using one Poisson process for each issued policy. The process starts only after the policy is issued, and we limit it to at most one claim per policy. From a global perspective, several claims and access requests may arrive at the same time step of the simulation.

For the reasons discussed earlier, we price policies using the Exponential Principle, and use Formula 4.5 for consumers' policy purchase decisions. We consider a range of values for the Exponential Principle's parameter α, which is the insurer's degree of risk aversion. Figure 4.3 shows the upper bound ε on the probability of eventual ruin from Formula 4.3, with a range of risk aversion indexes for the insurer and a fixed initial capital, with and without a log scale. This figure is intended to help the reader visualize the impact of risk aversion on the chance of ruin; note, however, that Formula 4.3's assumptions do not quite hold in our simulations, as the insurer's income from premiums will not be constant at each unit of time and the insured accesses will not all have identical claim size distributions.

For each potential insured access, the consumer expects a certain benefit, modeled as a random variable, against which the premium must be weighed. The consumer's *profit* is its actual benefit minus the premium it paid. Real consumers could benefit from collecting data on their actual benefits and determining their distribution for each class, e.g., using the maximum likelihood estimation suggested earlier for modeling historical claim size data.

Absent data on actual benefits, the exponential family of distributions is widely used in many disciplines for modeling outcomes of various kinds of transactions. Among the many members of the exponential family, normal distributions are common and easy to visualize, so we use normally distributed benefits. We model the receipt of benefit from insured accesses with a separate Poisson process for each access. The benefit's process starts only after the policy is issued, and the benefit arrives at most once per policy. For any Poisson process, the time between each pair of consecutive events is exponentially distributed.

For total claim sizes, which we refer to simply as *claims*, we adopt two distributions from the exponential family, which is widely used for modeling claims. The first is a normal distribution. However, a normal distribution can overestimate the total claims on a policy, because if one waits long enough when using a Poisson process for claim arrival, a claim will eventually arrive for any given policy. In contrast, in real life many policies never have any claims at all. The zero-adjusted gamma distribution (ZAGA) [84] is very effective at modeling this situation, because it explicitly models the chance of there being no claim at all. Thus when a claim arrives under the Poisson process, the ZAGA distribution explicitly models the chance that the "claim" is for \$0. The ZAGA distributions we used have fatter tails than the normal distributions, thus illustrating the impact of rarer events.

The probability density function of a Zero-Adjusted Gamma distribution (ZAGA) is defined by

$$f(x;k,\theta,q) = \begin{cases} q\frac{1}{\theta^k}\frac{1}{\Gamma(k)}x^{k-1}e^{-\frac{x}{\theta}} & \text{if } x > 0 \\ 1-q & \text{if } x = 0 \end{cases}$$

for $x \geq 0$, $0 \leq q \leq 1$ and $k, \theta > 0$. Here, k and θ are shape and scale of the usual gamma distribution, respectively, and q is the probability of the random variable x being positive.[1] Informally, we can say that ZAGA is derived by putting probability mass $1-q$ at $x = 0$ on the usual gamma distribution. This is a helpful property to represent claim sizes because policy holders don't file claims so often (no claims with probability $1-q$), but when they do (positive size claims with probability q), the claim sizes can be very large, which can be well

[1] Alternatively, q can be defined the other way around, i.e., as the probability of the random variable x being 0, as done in [84].

Table 4.4: Mean, Variance and Moment Generating Function of ZAGA

Mean	$qk\theta$
Variance	$q(k\theta)^2(1-q+\frac{1}{k})$
Moment Generating Function	$(1-q)+q(1-\theta t)^{-k}$ for $t<\frac{1}{\theta}$

Table 4.5: Parameters Used in Simulation: Parameters are set so that benefits tend to be larger than claims.

Parameter	Value
# of runs	1,000
# of producers	10
# of consumers	10
distributions for benefits	Normal
mean of Normal distribution for benefits	random sample from Uniform(1.0, 1.5)
standard deviation of Normal distribution for benefits	random sample from Uniform(0.25, 0.5)
consumers' utility function	Exponential Utility
parameter α_c for consumers' exponential utility	random sample from Uniform(0.1, 10)
insurer's initial capital w_I	10
consumers' initial capital w	10
distributions for claims	Normal
mean of Normal distribution for claims	random sample from Uniform(0.5, 1.0)
standard deviation of Normal distribution for claims	random sample from Uniform(0.1, 0.35)
insurer's utility function	Exponential Utility
parameter α for insurer's exponential utility	0.01, 0.1, 1, 10, 25, 50, 100
distributions for claims	Zero-Adjusted Gamma (ZAGA)
probability of positive size claims happening	random sample from Uniform(0, 0.2)
shape k of ZAGA for claims	random sample from Uniform(5, 10)
scale θ of ZAGA for claims	0.999
insurer's utility function	Exponential Utility
parameter α for insurer's exponential utility	0.01, 0.1, 0.25, 0.5, 0.75, 1

modeled by the gamma distribution. The mean, variance and moment generating function of ZAGA can be calculated as shown in Table 4.4.

When a consumer requests an insured access, IASimulator chooses its producer uniformly at random from the producers it has not purchased from previously. The rationale is that once a consumer has purchased a particular map, it does not need to purchase that map again.

The simulation parameters are shown in Table 4.5. Global parameters, parameters for the normal claim distribution, and those for the ZAGA claim distribution are shown at the top, middle, and bottom of the table, respectively. When a consumer requests a policy, the insurer uses the Exponential Principle to set the premium for the policy. The consumer computes its expected benefit from the insured access, then uses Formula 4.5 to determine the maximum premium it is willing to pay. If this is less than the quoted premium, the consumer buys the policy. Each consumer has its own parameter α_c for risk aversion, chosen uniformly at random from [.1, 10], and hence its own maximum premium for a particular map.

The simulation has to use concrete numbers for the benefit of accessing maps and for claim sizes. For each map, we choose an average benefit uniformly at random in [1.0, 1.5], with its average claims drawn uniformly at random from [.5, 1] for the normal distribution. For the normal distributions, the range of possible means of claims and benefits is narrow and relatively close, as otherwise the outcome of the simulation will be dominated by the larger values. For the normal distributions, standard deviations are set so that three standard deviations from the mean (a reasonable threshold for tail events) is at most twice the mean, so that the tail starts by 3 for benefits and by 2 for claims. The ZAGA distributions are chosen so ZAGA claims have the same average amount as normally distributed claims. This means that when there is a non-zero ZAGA claim with probability 0.1 on average, its average amount is ten times higher than the average value under the normal distributions. To avoid dull simulations where the quoted premiums are usually larger than the maximum premium the consumer will pay, the mean of the distributions for benefits is generally larger than that for claims. The average claim size is equal to the premium under the Net Principle, which in turn is less than the premium under the Exponential Principle, which governs what the consumer will pay. In the simulation, premiums do not include a fee for the insurer or producer, and we do not share profits, so producers break even. We simulate the behavior of 10 consumers and 10 providers and track the wealth (capital) of each consumer plus the insurer, each of whom has an initial capital of 10 for sharing. The benefit from sharing comes from outside the VO, i.e., it is not taken from other VO members' capital. The insurer's initial capital is rather low, a deliberate choice to allow us to investigate ruin empirically.

The simulation needs concrete λ parameters for the Poisson processes' exponentially-

Table 4.6: Parameter λ of Exponential Distribution: λ is set so that claims are fixed usually later than benefits being fixed and the inter-arrival time of consumers is generally longer than the delays of claims/benefits being fixed.

Time	Parameter λ of Exponential Distribution
Inter-arrival time of consumers	random sample from Normal(0.2, 0.01^2)
Delay of claims being fixed	random sample from Normal(0.5, 0.01^2)
Delay of benefits being fixed	random sample from Normal(1.0, 0.01^2)

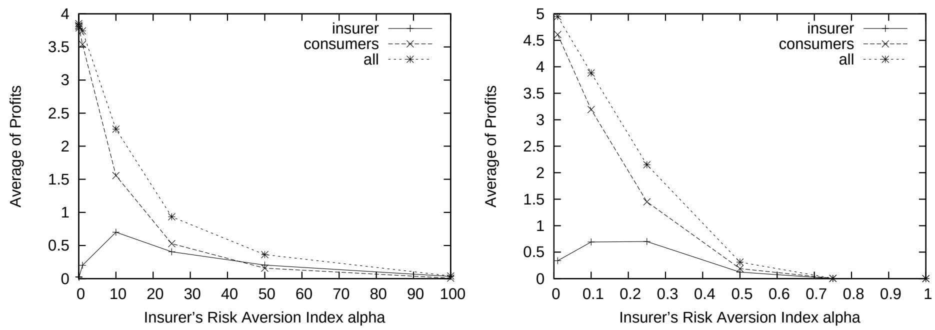
distributed inter-arrival times. As shown in Table 4.6, we choose the λ parameters randomly from a normal distribution, with the means of the distributions chosen so that benefits typically arrive before claims, and consumers usually make their next insured access request after the previous request's benefits and claims are known. That translates to an average of five time steps between requests from the same consumer, two time steps for the claims on a new policy to arrive, and one time step to learn the benefits of an insured access. We run the simulation 100 time steps, which is about twice as much time as consumers usually need to get a chance to buy all 10 maps, with this range of Poisson parameters. We repeat the simulation 1,000 times and report the averages across all simulations. We computed the standard error for each reported average, as the standard deviation divided by the square root of the number of runs. The resulting error bars were too small to be observed, so we do not include them in the figures.

In addition to the capital of the insurer and the VO consumers, we present the *ruin ratio*, which gives the chance of insurer insolvency for a given level of insurer risk aversion. The ruin ratio is the fraction of runs where the insurer' capital became negative. In runs where ruin occurs, we assume that the VO loans the insurer enough funds to continue to pay claims until it is back in the black. Thus the simulation continues even after ruin, on borrowed funds.

We also evaluate how bonus-malus systems affect the capital of principals, using the Dutch system explained earlier. The steps of consumers are updated every five time units according to the transition table, based on their number of claims in the previous period. To differentiate between good and bad risk takers, we set the probability of the ith consumer causing a claim to $i/10$ as shown in Table 4.7. When we want to model situations where

Table 4.7: Probability of Causing Claims: Probabilities are set so that consumers with larger IDs tend to cause more claims.

Consumer ID	1	2	3	$\cdots$	10
Probability of causing claims	0.1	0.2	0.3	$\cdots$	1.0



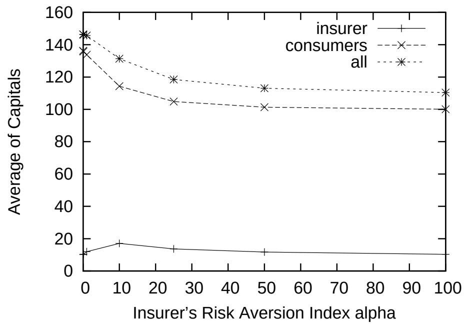
(a) Average Profits (Normal Claim Distribution)(b) Average Profits (ZAGA Claim Distribution)

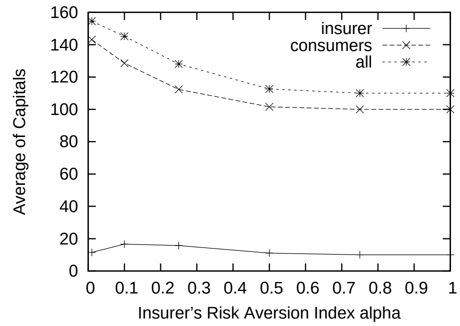
Figure 4.4: One-Round Simulation Results: The x-axis is the insurer's risk aversion index and the y-axis is average profits. When the insurer is more risk averse (i.e., a larger value for the risk aversion index α), fewer transactions take place and profits are smaller.

consumers don't cause claims based on these probabilities, our simulator doesn't make events in which corresponding damages materialize. Other parameters for this experiment are the same as those used in the discrete event simulations explained in the previous subsection.

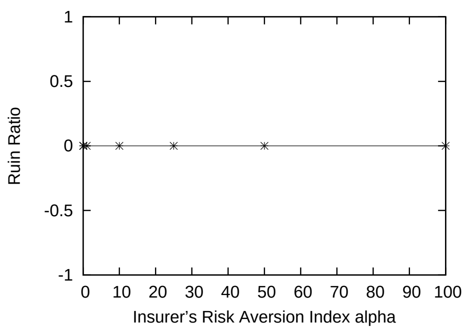
4.3.2 Experimental Results

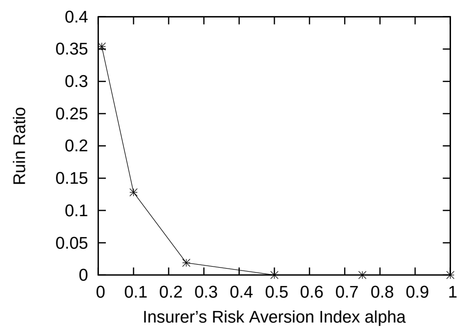
Figure 4.4 shows the result of the one-round simulations, i.e., the simulation of a single timestep. The three lines in each graph are for the average profit of the insurer, the average of the sum of profits of consumers, and for the average of the sum of profits of all principals (i.e., the insurer and the consumers). Note that the profits of producers are not shown here because they don't get any profits (but don't incur any loss either) in this set of simulations. These figures show that a more risk averse insurer (i.e., larger risk aversion index α value) tends to have smaller profits because the number of transactions decreases as α gets larger. An exception is that the insurer's profit is smaller with very small α values. This is because the risk seeking insurer allows many risky transactions and as a result it has to pay the





(a) Average Capital (Normal Claim Distribution) (b) Average Capital (ZAGA Claim Distribution)



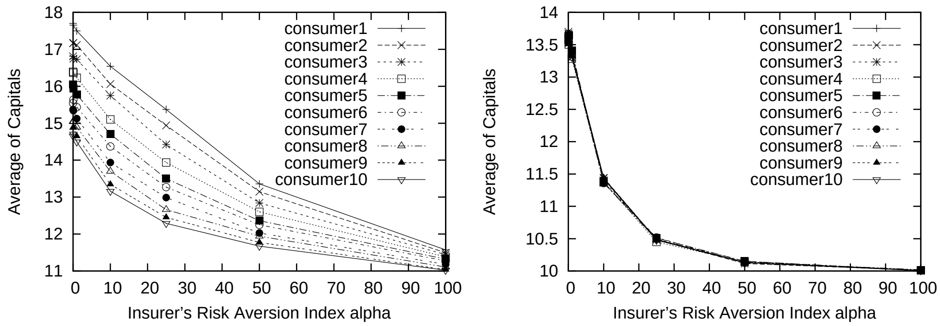


(c) Ruin Ratio (Normal Claim Distribution) (d) Ruin Ratio (ZAGA Claim Distribution)

Figure 4.5: Discrete Event Simulation Results: The x-axis is the insurer's risk aversion index and the y-axis shows the average capital at the end of the run, as well as the ruin ratio (fraction of runs that led to ruin). More risk averse insurers have less capital but a smaller chance of ruin, and sell fewer policies.

claims caused by such risky transactions. We also confirmed that the standard deviation of profits and that of the number of transactions also decrease as α gets larger. These results confirm that the risk aversion index is working as expected in our approach.

Now, we show the results of the multiple-round simulations, i.e., the simulation of multiple time steps. Figures 4.5a and 4.5b present the average capital at the end of the discrete-event simulation. The three lines in these graphs give the average capital of the insurer, the average sum of capital across all consumers, and the average sum of the capital of the insurer and the consumers. Figures 4.5c and 4.5d present the ruin ratio, which is rather large when the risk aversion index α is small and claims follow ZAGA distributions, even though the ruin ratio is always 0 when claims follow normal distributions. This is because claim sizes can be



(a) Average Capital with Bonus-Malus (b) Average Capital without Bonus-Malus

Figure 4.6: Bonus-Malus System Simulation Results: The x-axis is the insurer's risk aversion index and the y-axis is average capital at the end of the run. Consumers who cause fewer claims have more capital, if the bonus-malus system is enforced.

quite large under a ZAGA distribution, even though the average claim is generally smaller than the average benefit. More concretely, if we ignore the chance of a \$0 claim with ZAGA and instead assume that claims are always positive, then for ZAGA's parent family of gamma distributions, the average claim size is $k\theta = (5+10)/2 \times 0.99 \simeq 7.5$, while the mean claim size once \$0 claims are taken into account as in ZAGA is just $(0+0.2)/2 \times 7.5 = 0.75$. In contrast, the average claim size when claims follow a normal distribution is always $(0.5+1.0)/2 = 0.75$.

These figures show that the more risk averse insurer (i.e., larger α) has less capital but a lower ruin ratio, because the number of insured accesses decreases as α gets larger. Although the upper bounds on ruin probability in Figure 4.3 rely on assumptions not satisfied by our experiments, Figure 4.5d shows that these upper bounds are actually very good approximations. The ruin ratio for ZAGA claim distributions is rather large, due to the insurer's low initial capital and the high \$7.5 average claim size. In practice, as discussed earlier, the VO must start with sufficient capital for its planned portfolio size and take corrective action if the ruin probability approaches the VO's cap.

We also evaluated how bonus-malus systems affect the capital of principals, and Figure 4.6 shows the results with normal claim distributions, with and without a bonus-malus system. These figures show the average capital of each consumer at the end of the run. These graphs show that consumers who cause fewer claims (i.e., those who have smaller ID

Table 4.8: Parameters Used in Parameter Space Exploration Experiments.

Parameter	Values
# of producers	1, 5, 10, 25, 50, 100, 500, 750, 1000
# of consumers	1, 5, 10, 25, 50, 100, 500, 750, 1000
insurer's initial capital w_I	5, 10
parameter α for insurer's exponential utility	0.01, 0.1, 1, 10
duration of discrete event simulation	100, 250, 500

numbers) have more capital when the bonus-malus system is enforced[2]. Without bonus-malus, there are no such differences among consumers. These results confirm that the bonus-malus system can reward good risk takers and punish bad ones. This is a big step forward for access control because a bonus-malus enforced system can automatically adjust itself to users' recent activities so that it can encourage more beneficial information sharing with smaller probability of misuse of information while discouraging information accesses that are prone to result in bad outcomes.

4.3.3 Parameter Space Exploration

The simulation experiments so far examined the effectiveness of insured access, but they have not throughly examined the parameter space. In this section, we perform systematic large-scale simulations by varying parameter values to examine which parameters have the biggest impact on the system.

We focus on parameters that are not well examined in the previous section and vary parameters as shown in Table 4.8. The other parameters are the same as shown in Table 4.5. The results shown in this section are derived using a normal distribution for claims. We also tried ZAGA distribution, but we omit its results here because the use of ZAGA distribution didn't give us additional insights.

In many of the results, we can see two major phenomena: saturation and underutilitization. Saturation occurs when every consumer accesses every possible piece of information. In the situation of underutilization, on the other hands, relatively few accesses have been

[2]When we did the same experiment with ZAGA claim distributions, bonus-malus did not significantly impact capital, as ZAGA distributions already model the probability of filing claims.

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0.004	0.066	0.13	0.194	0.214
5	0	0	0.002	0.015	0.064	0.129	0.276	0.338	0.4	0.395
10	0	0.001	0.006	0.081	0.179	0.256	0.341	0.413	0.452	0.489
25	0	0.013	0.064	0.197	0.278	0.374	0.383	0.409	0.397	0.426
50	0.001	0.061	0.125	0.242	0.246	0.172	0.055	0.018	0.002	0.001
100	0	0.065	0.156	0.206	0.195	0.136	0.017	0.003	0	0
250	0	0.056	0.158	0.207	0.183	0.103	0.021	0	0	0
500	0	0.065	0.124	0.191	0.174	0.109	0.021	0	0	0
750	0	0.053	0.157	0.206	0.176	0.138	0.014	0	0	0
1000	0	0.058	0.133	0.206	0.178	0.119	0.024	0	0	0

(a) α = 0.01, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0.001	0.015	0.03	0.054
5	0	0	0	0	0.001	0.011	0.085	0.175	0.255	0.293
10	0	0	0	0.002	0.011	0.074	0.191	0.263	0.351	0.373
25	0	0	0.001	0.03	0.082	0.187	0.29	0.332	0.342	0.351
50	0	0	0.008	0.048	0.067	0.085	0.041	0.006	0.001	0
100	0	0	0.007	0.045	0.06	0.051	0.011	0.001	0	0
250	0	0	0.007	0.038	0.051	0.045	0.009	0	0	0
500	0	0	0.01	0.039	0.074	0.05	0.007	0	0	0
750	0	0	0.008	0.034	0.064	0.041	0.011	0.001	0	0
1000	0	0.001	0.01	0.054	0.071	0.049	0.008	0.001	0	0

(b) α = 0.01, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0.002	0.042	0.093	0.122	0.149
5	0	0	0	0.007	0.058	0.104	0.185	0.213	0.219	0.207
10	0	0	0.006	0.062	0.117	0.176	0.222	0.215	0.194	0.166
25	0	0.008	0.061	0.156	0.191	0.189	0.157	0.124	0.094	0.06
50	0	0.048	0.091	0.173	0.145	0.08	0.005	0	0	0
100	0.001	0.05	0.099	0.151	0.123	0.048	0.004	0	0	0
250	0.001	0.058	0.111	0.144	0.106	0.033	0.002	0	0	0
500	0.001	0.049	0.118	0.138	0.11	0.043	0.001	0	0	0
750	0.001	0.055	0.113	0.136	0.108	0.033	0	0	0	0
1000	0.001	0.046	0.135	0.149	0.095	0.035	0.004	0	0	0

(c) α = 0.1, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0	0.007	0.013	0.038
5	0	0	0	0	0.001	0.01	0.057	0.1	0.09	0.106
10	0	0	0	0	0.014	0.036	0.082	0.13	0.106	0.101
25	0	0	0.001	0.013	0.058	0.097	0.102	0.079	0.047	0.048
50	0	0	0.003	0.025	0.033	0.031	0.005	0	0	0
100	0	0	0.005	0.021	0.024	0.01	0	0	0	0
250	0	0	0.003	0.022	0.022	0.01	0	0	0	0
500	0	0	0.005	0.026	0.019	0.013	0	0	0	0
750	0	0	0.002	0.023	0.031	0.018	0.001	0	0	0
1000	0	0	0.002	0.036	0.032	0.008	0.001	0	0	0

(d) α = 0.1, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0.002	0	0	0.002	0
5	0	0	0	0	0.001	0.001	0	0	0	0
10	0	0	0.001	0.001	0	0	0	0	0	0
25	0	0	0.001	0.001	0	0	0	0	0	0
50	0	0.003	0	0	0	0	0	0	0	0
100	0	0.003	0.001	0.001	0	0	0	0	0	0
250	0	0.003	0.003	0	0	0	0	0	0	0
500	0	0.004	0	0	0	0	0	0	0	0
750	0	0.001	0.002	0	0	0	0	0	0	0
1000	0	0	0.002	0	0	0	0	0	0	0

(e) α = 1, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0
250	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0

(f) α = 1, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0
250	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0

(g) α = 10, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0
250	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0

(h) α = 10, wI = 10

Figure 4.7: Ruin Ratio: The row is the number of producers (np) and the column is that of consumers (nc). The duration of the simulation is 250 timesteps. The insurer's risk aversion index α and its initial capital w_I are varied as shown in the caption of each heat map.

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	15.37	56.87	108.8	264.5	524.1	1043	2599	5195	7790	10384
5	16.87	64.45	124	302.5	599.3	1196	2979	5954	8929	11904
10	18.79	73.73	143.1	349.7	694.1	1384	3451	6900	10348	13794
25	24.03	101.2	197.2	484.9	964.8	1922	4801	9602	14395	19193
50	29.97	29.97	251.4	619	1234	2460	6147	12284	18419	24584
100	30.94	135.1	266.1	656.3	1306	2611	6515	13032	19543	26050
250	32.06	139.5	272.6	671.3	1341	2678	6688	13371	20050	26738
500	32.03	139.7	274.9	677.9	1352	2699	6729	13472	20204	26938
750	31.98	140.2	274.5	678.6	1355	2703	6757	13508	20248	27008
1000	31.8	141.2	275.2	681.9	1357	2706	6763	13526	20275	27041

(a) α = 0.01, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.38	61.84	113.9	269.5	528.8	1048	2605	5200	7795	10389
5	21.89	69.55	129	307.1	604.6	1200	2984	5960	8932	11908
10	23.78	78.96	147.5	354.3	700	1389	3459	6907	10355	13802
25	21.89	106	202	490.3	970	1930	4803	9601	14402	19192
50	34.37	132.5	254.9	623.4	1237	2467	6155	12299	18440	24578
100	35.89	140.2	270.3	661.6	1314	2616	6524	13032	19557	26060
250	36.88	144.2	278	678.6	1348	2688	6694	13377	20057	26728
500	36.87	144.7	278.7	683.2	1354	2703	6746	13472	20207	26945
750	37.15	144.6	279.7	685.6	1359	2708	6763	13512	20263	27011
1000	36.79	144.7	280.3	684.7	1364	2712	6777	13529	20286	27045

(b) α = 0.01, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	15.4	56.92	108.9	264.4	524	1043	2600	5194	7788	10384
5	16.9	64.5	123.8	302.5	599.4	1195	2978	5951	8924	11898
10	18.73	74.07	143.1	350.1	694.6	1383	3450	6896	10340	13787
25	24.01	100.6	196.4	483.9	963.2	1922	4799	9580	14364	19166
50	29.48	127.5	250.5	617.4	1230	2459	6136	12258	18398	24520
100	31.23	135.3	265.4	655.4	1306	2602	6505	13004	19491	26007
250	31.81	138.4	271.6	671.9	1341	2673	6674	13353	20017	26680
500	31.82	139.2	274.4	678.2	1349	2693	6731	13447	20175	26898
750	31.98	139.6	274	679.5	1352	2700	6741	13481	20224	26962
1000	31.93	139.5	274.2	680	1356	2701	6749	13498	20249	27001

(c) α = 0.1, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.39	61.87	113.8	269.4	529.1	1048	2605	5199	7794	10388
5	21.88	69.4	129.1	307.3	604.5	1199	2983	5957	8930	11903
10	23.89	78.68	148	355.2	699	1388	3455	6898	10345	13770
25	29.18	105.7	202.2	488.7	967.4	1925	4800	9590	14385	19176
50	34.69	132.1	254.3	621.8	1235	2463	6138	12264	18394	24511
100	35.74	139.7	270.1	662.2	1308	2609	6510	13010	19511	26010
250	36.64	144.6	277.7	677.6	1344	2677	6679	13347	20018	26686
500	36.81	144.6	278.2	681.1	1354	2699	6732	13447	20165	26896
750	37.4	144.9	279.3	685.4	1360	2705	6748	13487	20225	26969
1000	37.11	144.4	279.6	684.5	1359	2709	6753	13498	20259	26999

(d) α = 0.1, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	15.38	56.85	108.7	264.2	523.3	1042	2597	5188	7780	10373
5	16.81	64.23	123.4	301	596.4	1189	2965	5924	8883	11844
10	18.71	73.27	142	346.8	688.3	1372	3425	6841	10258	13680
25	24.12	99.19	193.6	476.4	949.5	1893	4721	9441	14155	18878
50	28.92	124.4	244.9	603	1204	2403	6000	11996	17988	23979
100	30.3	132.8	259.4	640.7	1279	2552	6366	12744	19107	25472
250	31.36	135.2	266.7	658.9	1314	2621	6551	13092	19634	26171
500	31.49	136.3	268.5	666.6	1325	2643	6601	13207	19793	26388
750	31.19	137.3	268.6	666.6	1327	2650	6620	13236	19849	26467
1000	31.61	137.2	269.2	667.5	1330	2653	6623	13250	19874	26491

(e) α = 1, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.37	61.8	113.6	269.3	528.5	1047	2602	5194	7786	10377
5	21.86	69.32	128.5	305.7	601.6	1194	2968	5928	8889	11849
10	23.69	78.36	146.9	352.6	693.7	1378	3429	6848	10264	13682
25	28.73	104.3	199.1	481.3	953.6	1897	4728	9447	14164	18882
50	33.9	129.2	249.9	610.4	1209	2317	6007	11987	17990	23972
100	35.26	137.1	264.7	647	1284	2556	6378	12747	19113	25493
250	35.92	141.7	272.3	662.5	1317	2627	6550	13095	19640	26177
500	36.48	142.5	273.9	669.8	1327	2647	6607	13202	19796	26400
750	36.25	141.7	274	673.7	1332	2656	6620	13229	19853	26461
1000	36.84	142.7	275.5	672.2	1334	2657	6634	13255	19881	26497

(f) α = 1, wI = 10

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	15.22	56.15	107.3	260.8	516.5	1028	2562	5120	7677	10234
5	16.16	60.87	116.4	283.9	561.8	1119	2792	5577	8365	11148
10	17.31	66.48	128.1	312.6	618.8	1234	3077	6147	9216	12288
25	20.34	81.59	157.5	386.9	768.2	1532	3827	7649	11470	15296
50	22.44	94.05	182.9	449.9	892.6	1782	4446	8889	13322	17766
100	24.02	100.5	196.2	481	955.1	1910	4775	9540	14303	19080
250	24.98	103.9	203.5	500.5	997.8	1990	4963	9916	14880	19830
500	25.24	105.7	205.3	508.3	1009	2015	5025	10051	15072	20087
750	25.63	106.1	206	507	1013	2018	5042	10082	15128	20160
1000	25.38	105.4	207.2	510.3	1014	2026	5053	10111	15154	20198

(g) α = 10, wI = 5

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.22	61.13	112.3	265.6	521.5	1033	2567	5124	7682	10239
5	21.13	65.84	121.6	288.5	567.1	1125	2796	5582	8369	11154
10	22.31	71.44	132.9	317.4	624.2	1238	3082	6151	9222	12295
25	25.37	86.65	163.1	392	774.9	1539	3831	7654	11470	15297
50	27.84	98.95	188.1	454.2	896.8	1785	4454	8890	13328	17771
100	28.96	104.4	200.4	488.3	964	1921	4779	9546	14321	19085
250	29.83	109.7	207.3	505.9	1003	1995	4971	9929	14886	19841
500	30	110.9	210.5	512.2	1014	2018	5029	10050	15075	20085
750	30.26	110.4	210.9	514.1	1018	2027	5050	10089	15132	20165
1000	29.9	111.4	212.6	514.5	1018	2031	5061	10112	15165	20200

(h) α = 10, wI = 10

Figure 4.8: Sum of Capital: The row is the number of producers (np) and the column is that of consumers (nc). The duration of the simulation is 250 timesteps. The insurer's risk aversion index α and its initial capital w_I are varied as shown in the caption of each heat map.

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0	0.021	0.037	0.049
5	0	0	0	0	0	0.012	0.09	0.186	0.218	0.27
10	0	0	0	0.001	0.01	0.046	0.102	0.141	0.142	0.141
25	0	0	0	0.003	0.004	0.001	0	0	0	0
50	0	0	0	0	0.001	0	0	0	0	0
100	0	0	0	0.004	0.001	0.002	0	0	0	0
250	0	0	0	0	0.002	0	0.001	0	0	0
500	0	0	0	0	0.001	0.001	0	0	0	0
750	0	0	0	0.001	0.001	0.001	0	0	0	0
1000	0	0	0	0.003	0.001	0.001	0	0	0	0

(a) ruin ratio with duration 100

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.38	61.89	113.9	269.4	529	1048	2605	5200	7794	10390
5	21.9	69.47	129.3	307.1	604.6	1199	2983	5956	8929	11904
10	23.63	78.24	146.7	351.6	692.5	1376	3425	6838	10252	13662
25	26.12	90.43	171.4	411.6	814.7	1617	4033	8059	12074	16098
50	26.64	92.76	175	423.5	834.9	1659	4135	8262	12386	16512
100	26.57	93.14	177.1	427.3	845	1677	4178	8352	12521	16685
250	26.8	94	177.6	428.8	848.6	1687	4206	8400	12595	16792
500	26.81	93.8	178	429.7	849.9	1692	4212	8415	12621	16822
750	26.88	94.32	178.2	430.2	850.1	1694	4212	8415	12626	16829
1000	26.82	93.95	178	430.8	851.7	1693	4216	8421	12624	16832

(b) capital with duration 100

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0.001	0.015	0.03	0.054
5	0	0	0	0	0.001	0.011	0.085	0.175	0.255	0.293
10	0	0	0	0.002	0.011	0.074	0.191	0.263	0.351	0.373
25	0	0	0.001	0.03	0.082	0.187	0.29	0.332	0.342	0.351
50	0	0	0.008	0.048	0.067	0.085	0.041	0.006	0.001	0
100	0	0	0.007	0.045	0.06	0.051	0.011	0.001	0	0
250	0	0	0.007	0.038	0.051	0.045	0.009	0	0	0
500	0	0	0.01	0.039	0.074	0.05	0.007	0	0	0
750	0	0	0.008	0.034	0.064	0.041	0.011	0.001	0	0
1000	0	0.001	0.01	0.054	0.071	0.049	0.008	0.001	0	0

(c) ruin ratio with duration 250

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.38	61.84	113.9	269.5	528.8	1048	2605	5200	7795	10389
5	21.89	69.55	129	307.1	604.6	1200	2984	5960	8932	11908
10	23.78	78.96	147.5	354.3	700	1389	3459	6907	10355	13802
25	21.89	106	202	490.3	970	1930	4803	9601	14402	19192
50	34.37	132.5	254.9	623.4	1237	2467	6155	12299	18440	24578
100	35.89	140.2	270.3	661.6	1314	2616	6524	13032	19557	26060
250	36.88	144.2	278	678.6	1348	2688	6694	13377	20057	26728
500	36.87	144.7	278.7	683.2	1354	2703	6746	13472	20207	26945
750	37.15	144.6	279.7	685.6	1359	2708	6763	13512	20263	27011
1000	36.79	144.7	280.3	684.7	1364	2712	6777	13529	20286	27045

(d) capital with duration 250

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	0	0	0	0	0	0	0.001	0.011	0.027	0.044
5	0	0	0	0	0.001	0.016	0.092	0.172	0.232	0.285
10	0	0	0	0.001	0.012	0.07	0.214	0.287	0.354	0.383
25	0	0	0.001	0.027	0.097	0.212	0.323	0.414	0.454	0.502
50	0	0.001	0.016	0.108	0.213	0.326	0.393	0.445	0.481	0.513
100	0	0.007	0.053	0.159	0.218	0.228	0.14	0.076	0.048	0.015
250	0	0.01	0.059	0.166	0.189	0.176	0.086	0.029	0.009	0.003
500	0	0.008	0.065	0.179	0.21	0.175	0.087	0.025	0.007	0.003
750	0	0.012	0.056	0.155	0.206	0.152	0.087	0.019	0.005	0.001
1000	0	0.019	0.074	0.158	0.197	0.184	0.08	0.021	0.004	0.002

(e) ruin ratio with duration 500

np \ nc	1	5	10	25	50	100	250	500	750	1000
1	20.37	61.93	113.8	269.6	529	1048	2605	5200	7795	10389
5	21.96	69.65	129.2	307	604.5	1200	2984	5960	8932	11907
10	23.71	78.88	148.3	354.9	699.8	1390	3458	6905	10356	13806
25	29.6	106.7	204.6	496.8	983.8	1958	4876	9745	14616	19483
50	38.42	151.6	294.5	720.3	1429	2849	7107	14211	21306	28416
100	49.24	207.1	403.8	994.1	1973	3930	9820	19642	29422	39241
250	52.37	222.8	437.4	1075	2138	4272	10657	21307	31931	42561
500	53.49	229.5	444.6	1098	2182	4354	10889	21754	32613	43454
750	53.56	228.8	449.1	1104	2196	4385	10951	21880	32833	43758
1000	53.93	230.1	447.2	1106	2209	4395	10985	21960	32940	43882

(f) capital with duration 500

Figure 4.9: Effects of Duration: The row is the number of producers (np) and the column is that of consumers (nc). The insurer's risk aversion index α is 0.01 and its initial capital w_I is 10. The duration of the simulation is varied as shown in the caption of each heat map.

allowed. How quickly saturation occurs depends on the insurer's risk aversion index α and on the number of potential policy purchases. The latter is the product of the number of consumers and the number of producers.

Figure 4.7 shows ruin ratios with various parameter values, in the form of a heat map. The row of each heat map is the number of producers (np) and the column is that of consumers (nc). The duration of the simulation is 250 timesteps. The insurer's risk aversion index α and its initial capital w_I are varied as shown in the caption of each heat map. From these heat maps, we can observe the following phenomena. We explain conceivable reasons for each phenomenon.

The insurer with large α has a low ruin ratio. The insurer with large α is risk averse, and such an insurer doesn't allow risky transactions.

The insurer with large initial capital w_I has a low ruin ratio. Large initial capital provides large buffer to avoid ruin.

Few producers and few consumers result in a low ruin ratio. With few producers and few consumers, there are few transactions, lowering the possibility of risky transactions being allowed.

Many producers and many consumers result in a low ruin ratio. With many producers and many consumers, there are many transactions, and many transactions give the insurer sufficient capital from premiums to avoid ruin.

5-50 producers and many consumers result in a high ruin ratios Chances are relatively high that many consumers try to access information with high cost early in simulations because not so wide a variety of information (i.e., producers) is available.

Figure 4.8 shows the sum of capital, i.e., the sum of the insurer's and consumers' capital, at the end of the simulation with various parameter values, in the form of a heat map. The row of each heat map is the number of producers (np) and the column is that of consumers (nc). The duration of the simulation is 250 timesteps. We vary the insurer's risk aversion index α and its initial capital w_I as shown in the caption of each heat map. From these heat maps, we can observe the following phenomena.

- Many producers and many consumers result in large capital
- Capital doesn't change much when there are sufficiently many producers. (saturation)
- The insurer with small α has large capital.

So far, the duration of the simulation is 250 timesteps. We now examine how this duration affects results. Figure 4.9 shows the ruin ratio and the sum of capital at the end of simulation with various parameter values, in the form of a heat map. The row of each heat map is the number of producers (np) and the column is that of consumers (nc). The insurer's risk aversion index α is 0.01 and its initial capital w_I is 10. The duration of the simulation is varied as shown in the caption of each heat map. From these heat maps, we can observe a phenomenon that a long duration of simulation results in large capital and a high ruin ratio. This is reasonable because with longer duration, there are more chances that more transactions occur. If the run is long enough, saturation eventually occurs, i.e., every consumer accesses every possible piece of information. In Figure 4.9, saturation occurs at duration 100 (resp. 250, 500) when there are at least 25 (resp. 50, 100) producers.

Chapter 5

Crowdsourcing Experiments for Insured Access

In the previous chapter, we proposed insured access to reimburse information producers for damage they incur due to sharing their information, and examined its effectiveness through simulation experiments. Although experiments that simulate human decision making are helpful to examine how insured sharing works without real claim data, only experiments with real organizations and humans can tell us whether insured access would ever work in the real world. Unfortunately, real claim data is not available to us, so using such data is not an option. Without real claim data, however, we can still obtain experimental results produced by humans by making use of crowdsourcing services such as Amazon Mechanical Turk [2] or CrowdFlower [12]. Crowdsourcing services are often used for tasks that are simple for people but difficult for computers, e.g., image tagging and data cleansing.

Crowdsourcing does not allow us to test insured access in real-world scenarios, but it does allow us to examine real human behaviors in experiments. Then, we can test the following questions:

- Would humans be willing to use insured access?
- If so, how far will their behavior deviate from the ultra-rational model specified in the previous chapter?

For human consumers to make a rational decision on whether to use insured access to obtain a piece of information, they need to know the expected value of the information for them. In some VO scenarios, this value will be hard for the consumer to estimate. For example, how much is patient care improved if a hospital nurse shares information with a home nurse, without written authorization from the patient? In some scenarios, the consumer will have a clear understanding of the information's value, but not in financial terms. For example, the humans in a military unit might know exactly how helpful a map would be in completing their mission. When consumers have a clear understanding of the value of a piece of information, they still might not be in a position to decide whether that value exceeds the premium required to obtain it. In such scenarios, the consumers' parent organization may choose to establish a policy covering both valuation and acceptable premiums, so that the policy directly dictates consumers' access decisions.

When humans clearly understand the dollar value of a potential access, their behavior may still diverge from that of the ultra-rational consumers modeled in Chapter 4. Kahneman and Tversky [59] give two human tendencies: the certainty effect and the isolation effect. Under the certainty effect, people are less likely to choose options whose outcomes are uncertain, compared to outcomes that are certain though somewhat less favorable. For example, many humans would prefer to be guaranteed to receive $75, rather than to have a 75% chance of receiving $150. The certainty effect is one factor in humans' risk aversion in choices that include a guaranteed positive outcome, and to risk seeking in choices involving guaranteed losses. However, the certainty effect is inconsistent with the expected utility theory that underlies Chapters 3 and 4. Under the isolation effect, people generally pay little attention to factors that are shared by all prospects under consideration. This effect leads to inconsistent preferences when the same choice is presented in different forms.

Kahneman [58] also explains that humans have two modes of thought: *System 1* and *System 2*. System 1 is fast, instinctive and emotional. In contrast, System 2 is slower, more deliberative, and more logical. Decision theory is all about System 2, and the whole setup of insured access is based on System 2 principles. However, people normally just operate in their daily lives by using System 1. Thus, it's not clear that people would behave as the model predicts with insured access. We need to consider these psychological aspects of humans in our crowdsourcing experiments.

In this chapter, we first explain the insured access scenario used in our experiments. Then, we show how we set up a web service for people to participate in our study. After that, we explain the test procedures. Finally, we show experimental results and discuss how the results can validate the models underlying insured access.

5.1 A Supply Chain Simulation Game

For the crowdsourcing experiments, we need a scenario that meets the following criteria:

- The scenario needs to model a situation involving a VO.
- The scenario needs to model a situation where information sharing can benefit a VO and its information consumers, and can harm its information producers.
- The benefits and harm in the scenario need to be easily reduced to cash equivalents.
- The effect of information sharing must be clearly observable in the scenario, both for those conducting the experiments and those participating in them.
- The scenario needs to allow us to test the two questions listed in the beginning of this chapter.
- The scenario needs to be reasonably quick, as otherwise the experiments would be prohibitively costly.
- The scenario should not require a lot of domain expertise, as we are unlikely to find that expertise through crowdsourcing, and lengthy training of crowdsourcing workers would be too expensive.
- The scenario must be reasonably easy to implement, or have source code already available to us.

We searched for the potential candidates for the crowdsourcing scenario. All candidates we found were simulations of real-world situation, and most of them were business simulation games. For example, we considered FORESTIA [16], where players manage a virtual forest in such a way as to sustain the economy, protect biodiversity and satisfy the needs of multiple users such as hunters, fishers and hikers; Simunomics [32], where players create their own business and set their own strategies to conquer a vast, changing economy; and

Virtonomics [34], where players develop the virtual company according to their own unique scenarios. Information sharing could play a role in many of the business simulation games we considered. However, many took too long to play. Others required business expertise, were too complex for easy implementation, or made it too hard to separate out the financial effects of accessing a particular piece of shared information.

We settled on a well-known simulation game called the Beer Distribution Game. The Beer Distribution Game, or Beer Game in short, is a business simulation game developed by MIT Sloan School of Management in early 1960s. It has been widely used to illustrate human decision making and the concepts of supply chain management [80, 62]. The Beer Game is a classic supply chain optimization problem, and it has been widely used in training students in the supply chain management domain [55, 51, 42]. The game can be played in person as a board game, or over the internet [22, 30, 14].

The Beer Game matches the list of desiderata as explained below:

- The whole supply chain is a VO.
- Each player independently decides how much product to order, but a player quickly learns that this decision is easier if they know about other players' orders. This corresponds with the real-world understanding that supply chain management is easier and more effective when organizations along the chain share certain information with each other. However, organizations along a supply chain usually do not want to share their information freely with each other, as information about their orders and inventory is very sensitive and could be misused by their partners.
- Each unit of the product has an equivalent cash value.
- We can observe the effects of information sharing from the cash balances of the VO members.
- We can make a version of the Beer Game in which insured access is optional, to test whether human consumers make the same information access decisions as the insured access model predicts. We can also test whether human information producers want

to enable insured access in the game. To verify that insured access can produce better outcomes for a VO, we can run the game with and without insured access and compare the final cash balances.

- The typical total playing time is 30-60 minutes, which is reasonably quick for our experiments.
- The game is developed to educate students, and it doesn't require a lot of domain expertise.
- The rules of the game are simple enough to be implemented in a reasonable amount of time.

In this seemingly simple game, players' decisions often are far from optimal [72], frequently resulting in oscillations between major product shortages and oversupply, accompanied by player frustration and confusion. Even when a very simple and deterministic end-customer demand pattern is used, the results of the game are quite unpredictable, as they are strongly affected by players' product order decisions. An especially well-known phenomenon we often see in the game is the **bullwhip effect**, where product demand?information is grossly distorted as it moves from the end customer to the product manufacturer. The bullwhip effect occurs because each player tends to order more (or less) than actually needed, due to (and exacerbating) instability in the supply chain [41].

5.1.1 Original Rules

The Beer Game simulates a four-stage product supply chain: retailer, distributor, wholesaler, and producer, as shown in Figure 5.1. Four people take part in a game and they play the roles of the four companies that are each other's customers or suppliers in a supply chain. Players sell only one product, beer, and one unit of product is one crate of beer.

Each (simulated) week during a game, each player is responsible for determining how much product to order from their supplier, based on the stock levels in their warehouse, their customer's order, and their previous orders. Each week, the following happens:

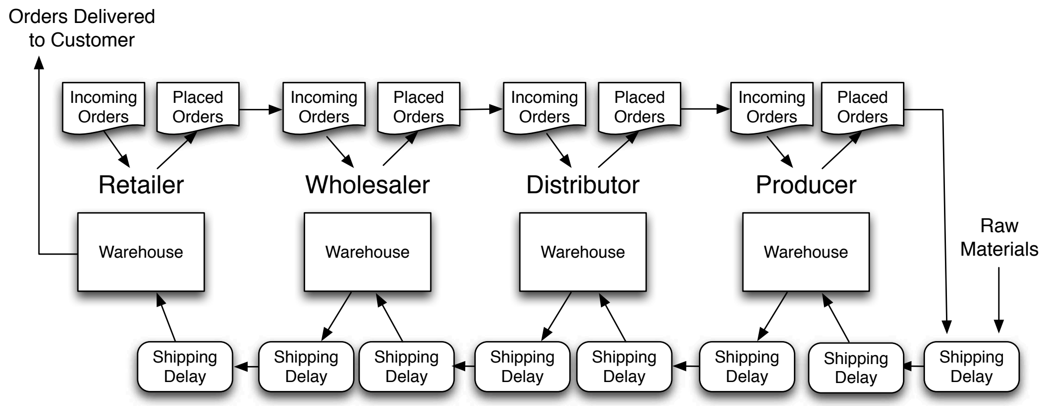


Figure 5.1: Beer Distribution Game Board

- A player receives product from their supplier.
- They receive orders from their customer.
- They ship product to their customer as ordered.
- They order additional product.

If a player cannot ship the full amount of product ordered by their customer, the shortfall is noted as backlog, and the player must clear the backlog by shipping additional product once it has enough in stock. The goal for players of the traditional Beer Game is to minimize the total cost for everyone in the supply chain by maintaining low stocks but nevertheless managing to fulfill all orders immediately.

There are two costs involved in the traditional Beer Game: inventory carrying costs and backlog costs.

- The **inventory carrying cost** per unit in stock is $1 per week.
- The **backlog cost** per unit that a player fails to ship is $2 per week.

The game is made more challenging by built-in time delays between placement and receipt of an order. In particular, when a player places an order, the order is processed at their supplier two weeks later. When a supplier fulfills an order, it takes two weeks for the shipment to arrive at their customer. Thus, it takes a minimum of four weeks for a newly placed order to be fulfilled. The exception to this rule is that it takes three weeks for a producer's newly ordered supplies to arrive.

While playing the traditional Beer Game, players must not exchange any information other than that constituted by the order itself. In other words, they cannot jointly plan a strategy, even though the objective of the four players is to minimize total costs.

A game lasts 52 rounds and each round represents a week, so that a game simulates a year. A player would need 30-60 seconds each round and thus the total playing time would be 30-60 minutes.

When a game is over, a debriefing session usually follows for educational purposes to review the results of the game and discuss the lessons learned.

The game sounds easy, but it actually is very hard because the delays and the lack of communications among players often cause the bullwhip effect.

5.1.2 Customized Rules

As stated above, there are two costs involved in the traditional Beer Game: inventory carrying costs and backlog costs. The version developed by the MA-system company, however, also includes a sales income when a player ships a crate of beer to its customer, and a purchase cost when a crate is delivered to a player [6]. For simplicity, the purchase cost and sales income are the same in all supply chain positions.

- The **purchase cost** for each unit that is delivered to a player is $1.
- The **sales income** for each unit a player shipped is $4.

Because of these additions, the overall goal of this version is to make as much revenue as possible for the supply chain. We also use the purchase cost and sales income in the versions used in our experiments because expanding the goal from cost minimization to profit maximization makes the game a little more realistic and a lot more appealing. It is more satisfying for human players to see a profit than to keep costs as close as possible to zero. Also, the goal for a real organization would obviously be to make a profit.

The game starts in the state of equilibrium where each player has 12 crates in stock, 4 crates arriving from supplier, and 4 crates being shipped to their customer. This results in $0 revenue because the costs and incomes offset each other exactly. The initial cash balance of each player is set to $0.

Another thing we do slightly differently from the existing versions regarding the costs and incomes is that we use larger dollar amounts so that players behave more responsibly for their decisions on their order amounts. Specifically, we use 100 times larger amounts, as listed below. Even though this change is superficial, people tend to be psychologically affected by the magnitude of dollar amounts.

- The **inventory carrying cost** per unit in stock is $100 per week.
- The **backlog cost** per unit that a player fail to ship is $200 per week.
- The **purchase cost** for each unit that is delivered to a player is $100.
- The **sales income** for each unit a player shipped is $400.

In the original Beer Game, players are not supposed to share their information with each other. This is because one objective of the game is to teach people how difficult it is to manage a supply chain without such information. In fact, many players have reached the conclusion that with complete sharing of information about players' status and orders, players would be able to avoid the bullwhip effect. While it is true that information sharing will reduce the severity of the bullwhip effect, it does not eliminate it. To drive this point home, the Near Beer Game was developed [8]. In the Near Beer Game, players can see the other players' status freely, and they can decide their amount of order by making use of all available information. Although this problem setting does make the game easier, it is still difficult because even with perfect information, i.e., even when there are no breakdowns in communication, the supply chain is still subject to the bullwhip effect due to procurement and manufacturing delays.

Although the Near Beer Game has its own lesson that it tries to teach people, in reality, it is unlikely for all the members of a supply chain to share their information freely. This is because detailed information about their orders and inventory must be very sensitive and they could be misused by their partners. For example, if a company is developing a backlog, then its best customer might decide to order from one of its competitors who promises faster delivery. If a company has too much stock in its warehouse, its customer might threaten to order from its competitor unless it cuts prices. A company's competitors might learn about its ordering strategy and copy it. Thus, although everyone would like to see detailed information from their partners, no one likes to give out much information about their own company, even though everyone recognizes that sharing information usually leads to better supply chain management overall.

To accommodate this situation, we introduced new versions of the game that use insured access. In our first version, the supply chain partners have agreed to share information with each other, in spite of the dangers outlined above. However, information sharing is not free. If a player wants to look at their partners' information, they have to pay for an insurance policy that covers the potential damage to their partners if their company misuses the shared information. Similarly, if their partners look at their information and abuse that privilege in some way, and they lose business as a result, then they can file a claim with the insurer for reimbursement of the damage to them. Thus, in this version of the game, players have to decide whether they want to pay the cost of the insurance policy, as well as the amount of orders.

As mentioned above, this version assumes that the supply chain partners have agreed to share information with each other under our scheme of insured access. However, we are also interested in examining an organization's willingness to adopt insured access in the first place. In a real deployment, the decision to share information and adopt insured access would be taken at the VO level. More precisely, managers from the organizations in the VO would get together and decide what information is to be shared and whether insured access is to be used. Then professionals with the appropriate business and actuarial knowledge would work out the details of the insurer's policies for different types of shared information. Upper management and actuarial professionals are likely to have a very different perspective on financial matters than do rank-and-file employees. For example, humans often are reluctant to spend money to hedge against harmful rare events, which is one reason that insurance is often required by laws and regulations, rather than being left to the individual's discretion. Thus neither the high-level nor actuarial decisions relating to insured access would be left to rank-and-file employees.

Unfortunately, in a crowd-sourced game, we cannot evaluate top management's or financial professionals' willingness to adopt insured access. And when we put those decisions in the hands of ordinary human players, we expect the classical psychological factors discussed later in this chapter to hold center stage, including loss aversion and probability discounting. Nonetheless, we introduced other versions that let the human player decide whether to adopt insured access, in which we consider the following three options:

(a) Players deny status information access requests.

(b) Players receive money from information requesters in exchange for sharing their status information.

(c) Players adopt insured access, i.e., they make information requesters purchase policies before they share their status information.

To make experimental results easier to interpret, we give players two of the above three options in a version of the game, rather than all three at once. Case (i) below is used to test players' willingness to share their status information, and case (ii) is used to test players' risk aversion.

(i) Players can choose either (a) or (c).

(ii) Players can choose either (b) or (c).

In both cases, if human players choose option (c), i.e., they opt into the insured access program, then they will be reimbursed for their losses. At the end of the game, the insurer's final balance will be split among the players who opted in, as a form of profit sharing. Without profit-sharing, a player might rationally decide to opt out because the chance of a major loss occurring in any particular game is small. Nonetheless, profit-sharing is delayed gratification, compared to cash in the pocket right now. Delaying gratification is not easy for humans in general. Even economic models discount the value of uncertain future profits, compared to known profits that are immediately available.

5.2 Experimental Setup

5.2.1 IRB Clearance

Because our crowdsourcing experiments involve people, we applied for UIUC Institutional Review Board (IRB) review of research involving human subjects. The project name for our IRB application is "Incentives to Encourage the Sharing of Information Across Organizational Boundaries", and its IRB Protocol Number is 14755. Its expiration date is 05/06/2015. The risk designation applied to our project is *no more than minimal risk*. We attach our online consent document and recruiting text approved by the IRB in Appendix A.

5.2.2 Web Application Implementation & Deployment

In order to collect experimental results, we developed a web app for people to play our versions of the Beer Game and hosted it at `https://oak.cs.illinois.edu/`. Players' actions during games are recorded in databases. We describe the game's implementation details in this section.

One of the requirements to realize Beer Game as a web app is real-time communications between a server and multiple clients. There are several Beer Game implementations available online [22, 30, 14] and they use Java to meet this requirement. This means that users need to install Java and enable its browser plug-ins to play games on these sites. This system requirement is rather cumbersome for users. There are also relatively frequent reports for Java vulnerabilities these days, and people would rather not enable Java on their systems. Because we try to reach the general public in the crowdsourcing experiments, we would like to eliminate this requirement of Java installation. Thus, we use HTML5 and Javascript for both client-side and server-side instead of using Java in our web app. With these technologies, all users need is just a modern web browser, and no special plug-ins are

Table 5.1: Model-View-Controller Components

Component	Technologies
Model	MongoDB
View	HTML5, CSS3, JQuery, JQueryUI, D3.js, Jade
Controller	Node.js, Socket.IO, Express, etc.

necessary. There is a recent trend in using server-side Javascript as well. One advantage of doing so is that we can use the same language, i.e., Javascript, top to bottom.

We use technologies in each of the Model-View-Controller (MVC) components as shown in Table 5.1 to realize our web app.

The core technology we rely on is Node.js [26]. Node.js is an open-source, server-side Javascript platform that provides an event-driven, non-blocking I/O web server. It is getting much attention these days and there are many modules for it. The key module in our web app is Socket.IO, which realizes real-time communications between a server and clients [33]. Socket.IO primarily uses the WebSocket protocol, but it falls back on multiple other methods, such as Adobe Flash sockets, JSONP polling, and AJAX long polling when needed while providing the same interface. We also use the Express module, which provides a web application framework that enables a RESTful API [15]. Other Node modules we use are: Connect middleware [10], Connect Flash messaging middleware [9], Passport [27], Passport-local [28], Connect Roles [11], and Bcrypt [4] for authentication and authorization, Mongoose [24] for MongoDB object modeling, and jStat [21] for statistical calculations. With Node.js, we use the Jade template engine to generate HTML [18].

To store experimental data, we use MongoDB, which is a leading NoSQL database [23]. MongoDB is a light-weight, document-oriented database that is said to match very well with Node.js. It is relatively easy to change database structures, or schemas, with NoSQL databases. This is convenient for us because we needed to change specifications from time to time as we progressed in our experiments.

For the client side, we use the jQuery [19] for HTML document traversal/manipulation, event handling, etc., and the jQuery UI [20] that provides a curated set of user interface interactions, effects, widgets, and themes. After each game is over, we need to show graphs

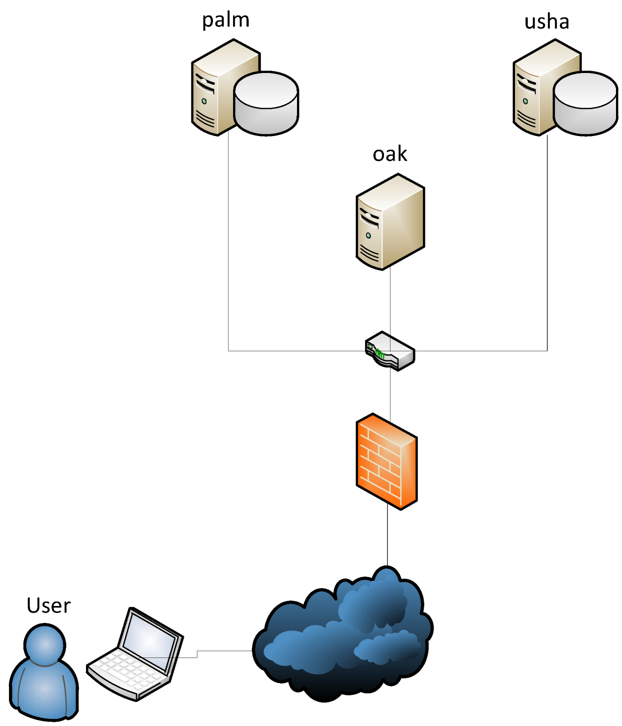


Figure 5.2: System Architecture

explaining game results. For that data visualization, we use D3.js [13], which uses Scalable Vector Graphics (SVG) [31] to draw graphs.

Figure 5.2 shows the system architecture of our Beer Game web app. The machines `oak.cs.illinois.edu`, `palm.cs.illinois.edu`, and `usha.cs.illinois.edu` are our server machines in the `cs.illinois.edu` network, which are protected by a firewall. Each of these server machines serves the processes as shown in Table 5.2.

The machine `oak` is a main server machine that serves our Node.js Beer Game Web App process.

We duplicate database instances, and `palm` and `usha` serve as a primary and secondary MongoDB servers, respectively. In addition, `oak` serves as a MongoDB arbiter to coordinate the primary and secondary MongoDB processes. This database duplication improves

Table 5.2: Servers for Beer Game Web Application

Host Name	Server Processes
`oak.cs.illinois.edu`	Node.js Beer Game Web App Process MongoDB Arbiter Nginx Web Server (Reverse Proxy)
`palm.cs.illinois.edu`	MongoDB Primary Server
`usha.cs.illinois.edu`	MongoDB Secondary Server

availability and also makes it easier to take backups.

The Node.js process is run by a normal user without root privileges, to reduce risks if the process is compromised. Because of that, the process cannot use port number 80 (without SSL/TLS) or 443 (with SSL/TLS). It uses port 3000 instead. This means users need to specify the port number when they access our web app, which is cumbersome for users. To avoid this extra work, we set up a Nginx [25] web server and use it as a reverse proxy.

Our experiments don't collect sensitive information from participants, but to secure data as much as possible, we encrypt all network communications with SSL/TLS. We installed a certificate on `oak` that has been signed by a UIUC authority. When players use our system for the first time, they need to sign up to create accounts. After logging in to our system, a player can access only their own information, e.g., past game history. We set up the MongoDB instances so that only an administrator who knows its password can access the instances directly.

Figures 5.3 to 5.5 and Figure B.1 to B.9 in Appendix B are screenshots of the web app we developed. Figure 5.3 and Figure B.1 to B.9 in Appendix B show screenshots while a user is playing our game. Figures 5.4 and 5.5 show screenshots of game results after the game is over.

We plan to opensource the code for our web app and make it available on the author's GitHub repository [3].

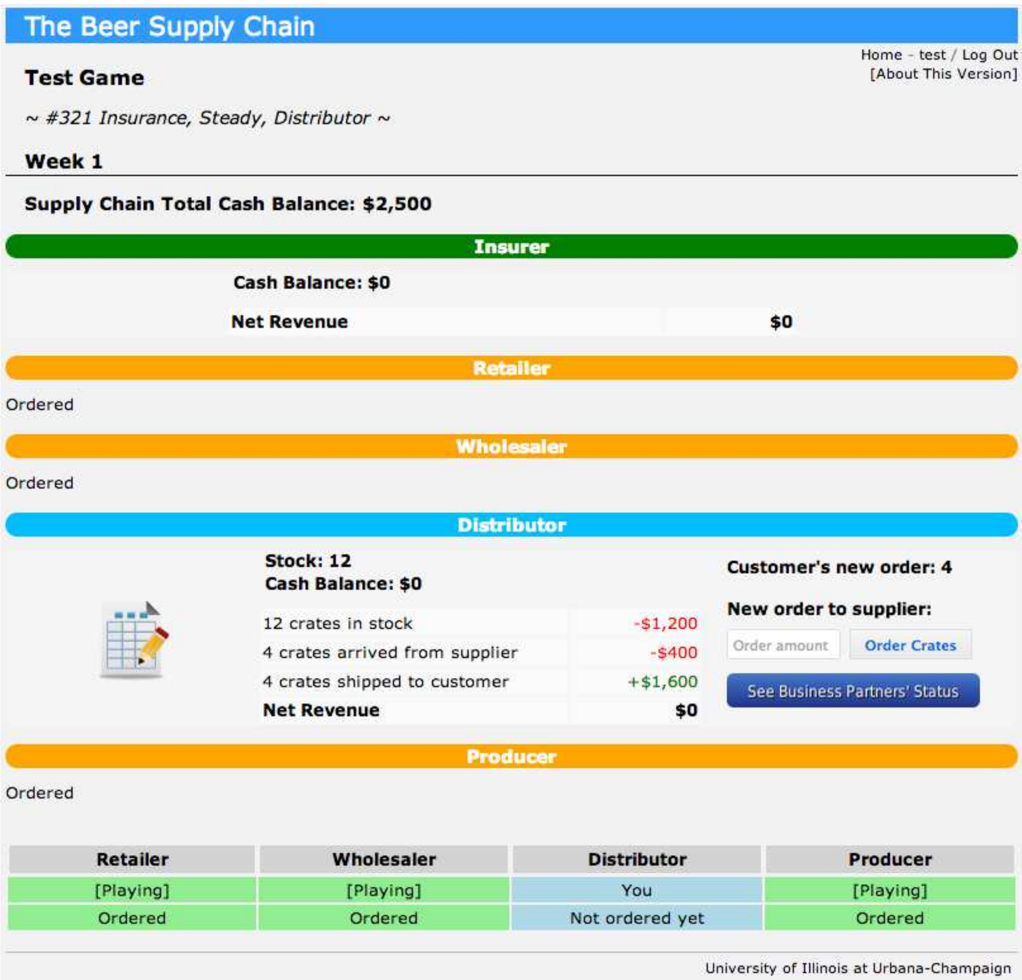


Figure 5.3: Screenshot of Beer Game Web App (playing)

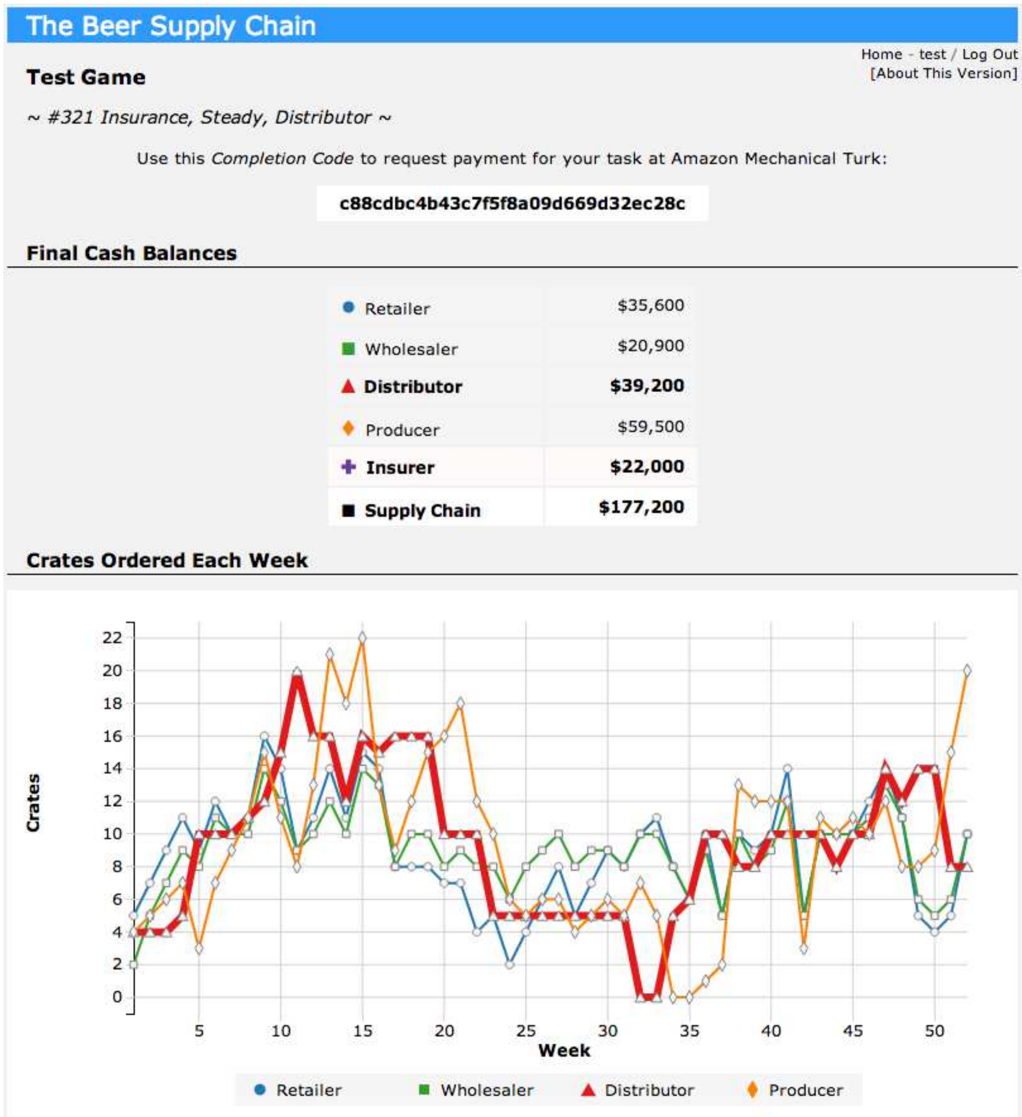


Figure 5.4: Screenshot of Beer Game Web App (result 1)

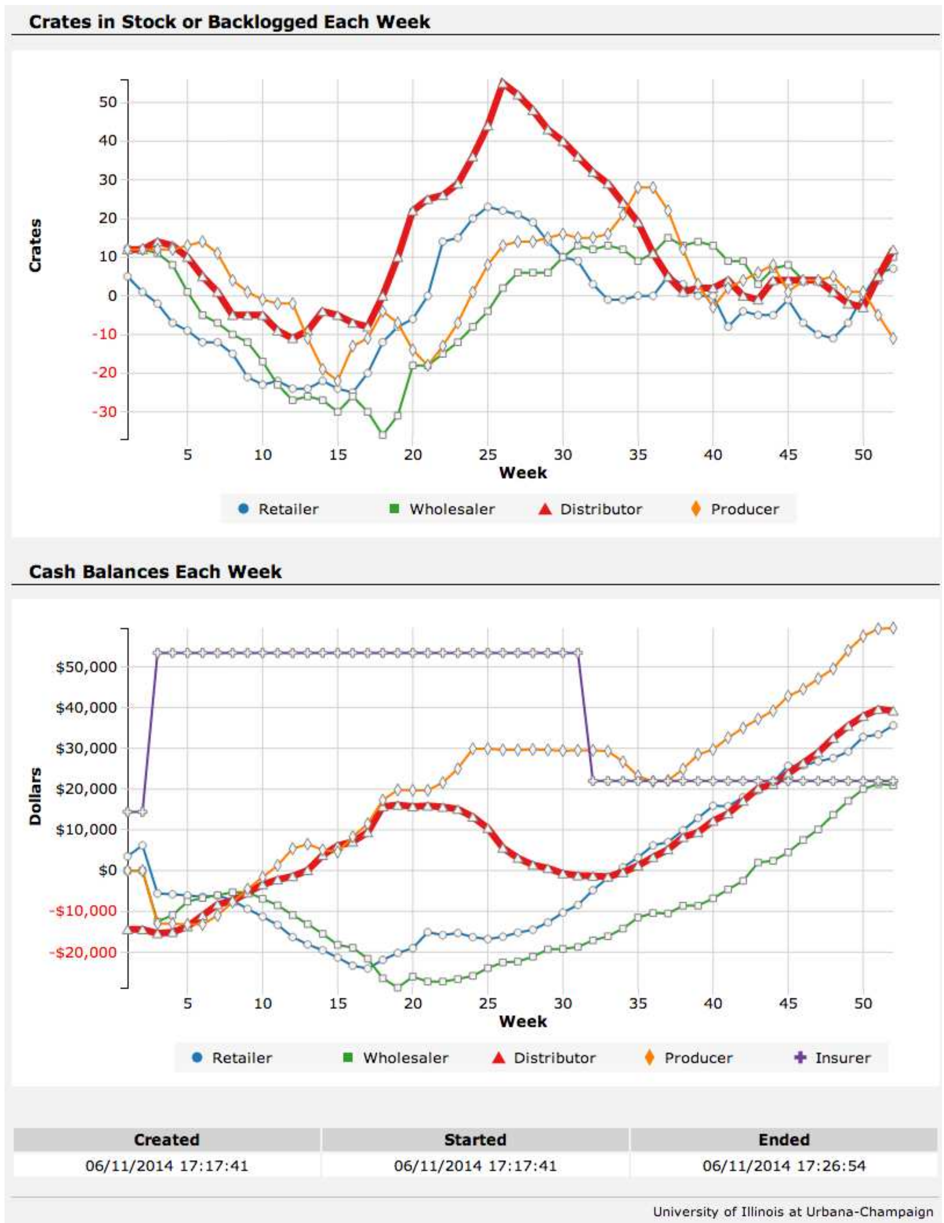


Figure 5.5: Screenshot of Beer Game Web App (result 2)

5.2.3 Automated Ordering Strategy

The Beer Game is usually played by four people. Human players often feel frustrated during a game because they are not achieving the results they want. In the debriefing time after a game, it is explained that these feelings are common and that reactions based on these feelings would contribute to the bullwhip effect. In our experiments, however, it is convenient if we can automate the play for some of the four positions, even though, generally, the more human players a game has, the better the learning opportunities because the human players can discuss how they reacted as the game progressed and how they decided their order amounts, during the game debriefing session.

Instead of using four human players, it is better for our game implementation to support automated play for selected positions in the supply chain. This is because our human players are crowdsourcing workers, each with their own schedule, making it very difficult to coordinate multiple humans to play together at the same time, without introducing annoying delays and artifacts that would make it difficult to interpret the game results. For example, one player might need to break to take a phone call, leaving the other three hanging. The extra waiting time might make one player go past his time limit at Amazon Mechanical Turk for completing the game. When he realized that, he might quit in frustration, ruining the game for the other players. Or, he might continue playing half-heartedly, making it hard to interpret the results of the game. More generally, suppose a game produces very little profit for the VO. Which of the four people played badly, and which are innocent victims of their bad partners? Or, do the four people just play badly with each other?

We needed a setup that would make it easy to compare results and players across multiple games. Thus, we decided to automate three out of the four players. In explaining the game to the human players, we called these automated players "computer players". In order to implement computer players, we need high-quality ordering strategies, i.e., algorithms to determine how many crates of beer a computer player should order each round. We need high-quality strategies so that the computer players will not inadvertently sabotage the results of games, which would frustrate the human players and make it hard for us to judge how well the humans are playing. A supply chain is no stronger than its weakest link, and if possible, we would like the human player to be the weakest link.

Interestingly, optimal ordering strategies for versions of the Beer Game are a research topic in their own right [80, 81, 65]. One very simple strategy is a customer order strategy used by the implementation by the MA-system company, under which a computer player's outgoing order equals its incoming customer order [6]. We refer to this as the "Pass-through" strategy in this thesis. In the implementation by the MA-system company, this very simple strategy works reasonably well because the end-customer demand (the incoming order to the retailer) pattern in this implementation is always One Step Demand (OSD), in which the customer demand is four crates of beer per week until week 4 and then steps to eight crates of beer per week for the rest of the game. However, if the pattern of demand is more complex, the Pass-through strategy works quite poorly, producing marked bullwhip effects.

Strozzi et al. [81] presents an algorithm for the optimal Beer Game order policy using a genetic algorithms (GA) [69] technique, but it is primarily based on very simple customer demand patterns. In our experiments, we need to use more complex demand patterns because if we use just simple patterns such as One Step Demand, human players learn the optimal ordering strategy after a few games and we cannot examine the effects of information sharing and insured access. If more complex customer demand patterns are used, the Pass-through strategy or the algorithm by Strozzi et al. [81] obviously do not always work well.

Liu et al. [65], however, proposes an effective algorithm that works well even with such complex demand patterns. Their approach is based on the three ordering principles proposed by Sterman [80]. According to the principles, players should place sufficient orders to:

- satisfy expected demand
- adjust inventory levels
- adjust for orders currently in the supply chain

Based on these principles, Liu et al. [65] proposes the following equations. We refer to this as the "Stock Management Structure" (SMS) strategy in this thesis, as they do in their paper.

First, orders must be nonnegative:

$$OP_t = max(0, OP_t^*) \tag{5.1}$$

In Equation 5.1, OP_t represents the actual orders placed at week t, whereas OP_t^* represents the orders calculated through the *ordering heuristic* at week t defined as follows:

$$OP_t^* = ED_t + \gamma(INV_t^* - \beta \times OSL) \tag{5.2}$$

In Equation 5.2, ED_t represents the expected demand at week t (see Equation 5.3), INV_t^* represents the discrepancy between desired and actual inventory at week t (see Equation 5.4), and OSL represents the orders in the supply chain. γ and β represent the adjustment parameters for the inventory and the supply chain, respectively. Equation 5.2 clearly takes into account the three factors by Sterman [80]: expected demand, inventory level, and orders in the supply chain. The expected demand (ED_t) in Equation 5.2 is expressed as:

$$ED_t = \theta \times CD_{t-1} + (1 - \theta) \times ED_{t-1} \tag{5.3}$$

In Equation 5.3, ED_t and ED_{t-1} are the expected demand at week t and $t - 1$; CD_{t-1} is the customer demand at week $t - 1$, and θ describes the rate at which the demand expectations are updated. The parameter θ is typically set to 0.25 [80], and we use that value in our experiments. The discrepancy between desired and actual inventory (INV_t^*) in Equation 5.2 is formulated as follow:

$$INV_t^* = Q - INV_t + BL_t \tag{5.4}$$

In Equation 5.4, Q represents the desired inventory level, INV_t represents the current inventory and BL_t represents the backlogs at week t. The parameter Q is set to 17 in the paper by Liu et al. [65], and we use that value in our experiments.

The two adjustment parameters γ and β are used by the game participants to determine their orders [83]. Intuitively, these two factors weight the importance given to how far the warehouse inventory stock level is from the player's preferred level, and the orders the player has placed in the past but that have not been delivered. More precisely:

(γ) This represents the discrepancy between desired and actual inventory ordered. This parameter is usually represented in the range ($0 \leq \gamma \leq 1$).

(β) This represents the fraction of the supply chain taken into account. This parameter is usually represented in the range ($0 \leq \beta \leq 1$). If $\beta = 1$, the participant factors in all of their orders in the supply chain or conversely, if $\beta = 0$, the participant factors in no orders in the supply chain.

The combination of the adjustment parameters (γ, β) corresponds to a set of behaviors for a given participant, and Liu et al. [65] provides optimal values for them by minimizing total costs using a particle swarm optimization (PSO) [60, 46] and a genetic algorithms (GA) [69] approach.

The paper by Liu et al. [65] uses the traditional Beer Game in which players don't know the other players' status. In our experiments, however, we also use versions in which players do have access to such information. In such versions, the computer players use just the actual customer demand at week t, i.e., CD_t, instead of ED_t in Equation 5.2.

Another difference between the traditional Beer Game and our versions is that the goal of our versions is to maximize revenues, as already mentioned, while that of the traditional Beer Game is to minimize costs. Thus, the sets of optimal value pairs for the two adjustment parameters provided by Liu et al. [65] (one set includes four (γ, β) pairs for the four positions) are not necessarily the best values in our versions. Table 5.3 shows the two sets from Liu et al. [65]. The two sets are obtained with PSO and GA, respectively, and they are for the Steady demand pattern, which we use extensively in our experiments. In this demand pattern, the mean demand remains constant over time. The distribution of the demand conforms to a normal distribution. We use a mean of 8 and the standard deviation of 2.

We conducted a Beer Game tournament that pitted four configurations against each other. The first configuration consisted of four computer players using the SMS algorithm with the PSO γ and β values from Table 5.3. The second configuration was identical, except that the computer players used the GA values from Table 5.3. The remaining two

Table 5.3: Best γ and β for Stock Management Structure (SMS) Ordering Strategy, from Liu et al. [65]

Position	Parameter	PSO	GA
Retailer	γ	0.2215	0.0267
	β	0.7832	0.2753
Wholesaler	γ	0.0823	0.6076
	β	0.7577	0.7578
Distributor	γ	0.1602	0.1693
	β	0.7342	0.6992
Producer	γ	0.8437	0.9726
	β	0.7167	0.7391

Table 5.4: Supply Chain Balance with Stock Management Structure (SMS) Ordering Strategy

Final Cash Balance	Expected (PSO)	Expected (GA)	Actual (PSO)	Actual (GA)
Average	$99,600.10	$244,840.60	$397,801.40	$316,440.20
Standard Deviation	$102,497.54	$59,433.25	$29,537.13	$29,469.88

configurations differed from the first two in that we replaced the expected demand ED_t used in the SMS algorithm with the actual customer demand CD_t. The tournament consisted of 1,000 games in each of these four configurations.

Figure 5.6 shows average supply chain balances with the γ and β in Table 5.3. The length of the error bars is set to 1 standard deviation. Table 5.4 gives average and standard deviation of supply chain balances. "Expected" means the expected demand at week t, i.e., ED_t is used in Equation 5.2. This is the case where a player cannot see the other players' status information. "Actual" means the actual demand at week t, i.e., CD_t is used instead of ED_t in Equation 5.2. This is the case where a player can see the other players' status information.

All four configurations achieve a positive supply chain balance on average, so they all satisfy the criteria we discuss later for reasonable human play. However, the configurations that use actual demand rather than expected demand achieve much higher final cash balances. Also, when using expected demand in SMS, the GA configuration's final balance is more than twice as high as the PSO configuration's final balance. However, when using

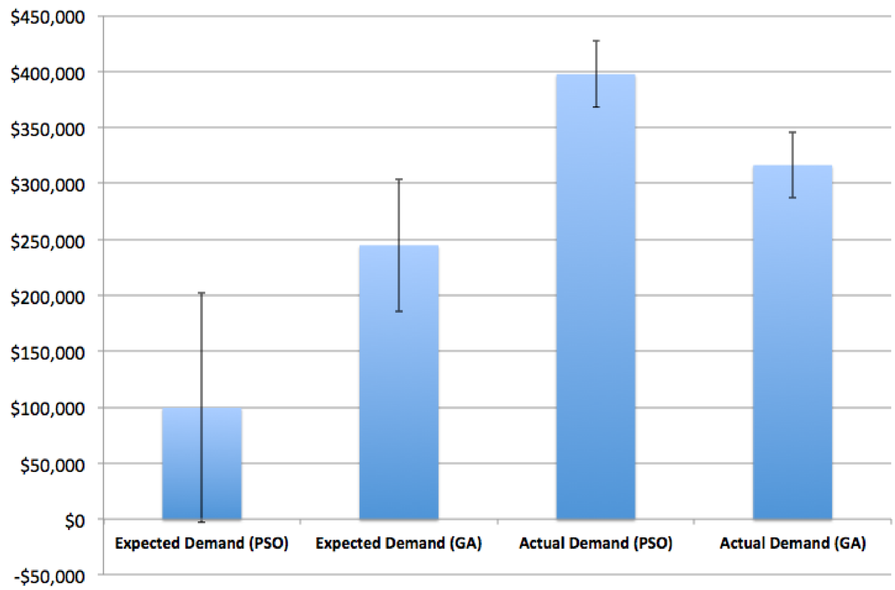


Figure 5.6: Average Supply Chain Balance with Stock Management Structure (SMS) Ordering Strategy

actual demand in SMS, the final cash balance is approximately 20% higher for PSO than for GA.

Although the γ and β values in Table 5.3 are unlikely to be optimal in our versions, we can see that these values are working reasonably well in our settings, too. Because the focus of our experiments is not to find the best algorithms or parameter values, it suffices to use reasonably good combinations of the parameter values. Based on the results in Figure 5.6 and Table 5.4, we chose the γ and β values for the computer players to give the best possible final cash balances possible, subject to the visibility of the information needed for SMS. Specifically, we use the values by GA when the expected demand must be used because a player cannot see the others' status and those by PSO when the actual demand can be used because a player has access to the others' status.

Note that when players cannot see the others' status, we use the expected demand for the retailer too even though the retailer always knows the actual demand. This may sound irrational, but we found that if we use the actual demand for the retailer and the expected demand for the other players, the outcomes become worse than those when the expected demand is used for all of the four players including the retailer. This is likely because the four players' orders don't synchronize well if one of them uses the actual demand and the others use the expected demand, which becomes a factor in causing the bullwhip effect.

5.2.4 Amazon Mechanical Turk

Amazon Mechanical Turk [2] is the most popular crowdsourcing service, and we use it in our experiments. Templates for simple, well-defined tasks such as tagging and data cleansing are available on Amazon Mechanical Turk. Our experiments, however, don't fall in such categories, and we host our own web app on our servers. Because of this separation, we need ways to connect Amazon Mechanical Turk and our web app. For that purpose, we first asked each player to use their Amazon Mechanical Turk Worker ID as their username when they made an account on our web site. This allows us to know which worker played which game on our system. In addition, we made our web app issue a completion code every time a player finishes a game, and asked the worker to fill this code in the task completion form on Amazon Mechanical Turk. This completion code is generated as a hash value from the combination of the player's username (Amazon Worker ID), a game ID, the position of the player, and the timestamp when the game is ended. We collect optional comments from human players when they submit their tasks.

On Amazon Mechanical Turk, individual tasks that requesters ask workers to complete are called Human Intelligence Tasks, or HITs. We used the recruiting document approved by the IRB (see Appendix A.2) when we posted our HITs. As already explained, workers need to create accounts on our system to participate in our study, and we asked them to read and agree to the online consent document approved by the IRB (see Appendix A.1) when they sign up.

5.2.5 Versions of Games

The major goal of our crowdsourcing experiments is to show that the introduction of insured access encourages information sharing, which in turn gives the VO and its members better outcomes. Thus, we want to see whether players choose to use insured access when it is available to them. We are also interested in the difference between outcomes in the traditional Beer Game in which no information is shared and those in our insured access version in which a player can see the others' status if the player purchases an insurance policy.

There are three parameters that we can change to make different versions of the Beer Game: the visibility of status information, the end-customer demand patterns, and the position of players.

Regarding the visibility of status information, we prepared the following five configurations:

See All A player can see the other players' status freely during the game, as in the Near Beer Game.

See Only Yours A player cannot see the other players' status. This is the usual setting in the traditional Beer Game.

Insurance A player can see the other players' status only if the player purchases an insurance policy. This is our insured access scheme.

Share & Keep Money / Share & Require Policy A player can choose either to receive money from the other players in exchange for sharing the player's status information, or can make the requesters use that money to purchase insurance policies before they see the information. This is used to test players' risk aversion.

Deny / Share & Require Policy A player can choose either to deny requests to see the player's status information, or else make the requesters purchase policies before seeing the player's status information. This is used to test players' willingness to share their status information.

As for the end-customer demand patterns, Liu et al. [65] modeled the following four patterns based on the outlines by Yan and Woo [88]:

One Step Demand This demand changes only once in the period of this simulation. It is typically set that the customer demand is four crates of beer per week until week

4 and then steps to eight crates of beer per week. This is the standard pattern of demand used when people play the Beer Game, and because of that, this is also the most widely analyzed pattern in the literature.

Steady Demand (This is called "Stationary Demand" by Liu et al.) The mean demand remains constant over time. The distribution of the demand conforms to a normal distribution. We use a mean of 8 and the standard deviation of 2.

Uniform Demand This demand fluctuates randomly and is generated using a uniform distribution in the range of [0,16].

Cyclic Demand This pattern varies cyclically over time to model the demand pattern for seasonal merchandise such as snow shovels, Easter egg dye, Halloween costumes, or fireworks. The mean value of demand changes periodically. The cycle of the demand is a year. In every cycle of the first half, the distribution conforms to a normal distribution with the mean of 10 and the deviation of 2; in the second half, it conforms to a normal distribution with the mean of 6 and the deviation of 2.

The One Step Demand pattern is the most commonly used demand pattern when analyzing the Beer Game. However, it is only useful for training players in our experiments, because good players quickly learn the optimal ordering strategy for this demand pattern, and information sharing is irrelevant thereafter. Thus after the initial training, we use the Steady Demand pattern, as it makes games reasonably interesting and challenging for players. The Uniform Demand pattern seems unrealistic and unnecessarily challenging, while the Cyclic Demand pattern is an interesting option for future work.

Note that each game has its own unique end-customer demand, generated from a unique random seed. Thus, each game will be different. As Figure 5.6 and Table 5.4 show, the final cash balances vary game to game even when computer players take all of the four positions. Thus, while the final cash balance achieved in a game does serve as a rough indicator of a human player's skill, the balance is not a precise measure of skill. As an extreme example, if the end-customer demand is for zero crates every week, the supply chain will not have a high final cash balance, no matter how skilled the players are.

The last thing to consider is which position in the supply chain should be played by the human. The retailer position is not a good position for the human player, because the retailer knows the end-customer demand throughout the game. End-customer demand is the single most useful piece of information to share with others, so if a rational human player already knows this value, the player will have little motivation to buy a policy so that they can see other players' status information. We exclude the producer position next because it is the hardest position to play, and so it might require quite a lot of training before humans could play reasonably well. This leaves us with the wholesaler and distributor positions. We have the human play the distributor position in our experiments, because the distributor position is further from the retailer than the wholesaler position. This location makes the position more difficult to play and makes information sharing more valuable. The other three positions are played by computer players.

Based on these considerations, we prepared the following six versions for the crowdsourcing experiments:

1. #111 See All, One-Step, Distributor
2. #121 See All, Steady, Distributor
3. #221 See Only Yours, Steady, Distributor
4. #321 Insurance, Steady, Distributor
5. #421 Share & Keep Money / Share & Require Policy, Steady, Distributor
6. #521 Deny / Share & Require Policy, Steady, Distributor

Regarding the ordering strategy for computer players, we use the Pass-through strategy in version #111, but in the other five versions, we use the SMS strategy. For the SMS strategy, the expected demand is used in versions #221 through #521 before a player purchases an insurance policy, because the player doesn't see the other players' status. The actual demand is used in versions #121 and #321 through #521 after a player purchases an insurance policy, because the player can see the others' status.

In order to let human players know about the characteristics of each version, we show the texts in Appendix C at the beginning of games. Along with the texts, we also show a video tutorial we made to teach players the basic features of our games. The video is available at the author's YouTube account [5].

5.2.6 Versions with Insurance

Figures 5.7, 5.8, and 5.9 are flowcharts for #321 Insurance, Steady, Distributor, for #421 Share & Keep Money / Share & Require Policy, Steady, Distributor, and for #521 Deny / Share & Require Policy, Steady, Distributor, respectively. The flowcharts are from the human player's viewpoint.

In all of these versions, human players can click a button to request accesses to computer players' status information before they make their own orders, if the computer players' statuses are not visible already. Because computer players are rational and because premiums are calculated using premium principles, the computer players always share their status information if the human player buys a policy from the insurer. If a player gets access to the status information once, it is visible until the end of the game.

In versions #421 and #521, human players get status information access requests from computer players in the first week that the premiums are lower than what the computer players consider to be the value of the status information. In version #421, human players can either keep the money offered by the computer players, or require them to use that amount of money to purchase policies before they can see the status information. In version #521, human players have the option to grant (with insured access) or deny the computers players' requests to share status information. If they deny the requests, the computer players will not ask them again in the remaining weeks.

If a player shares status information, the game determines whether there will be monetary damages as explained in detail later in this chapter. When there should be monetary damages by chance, we choose a week from the remaining weeks uniformly randomly. When that week starts, the game handles claim filing and reimbursement if the player who requested the access has purchased a policy. If the player who requested the access has paid

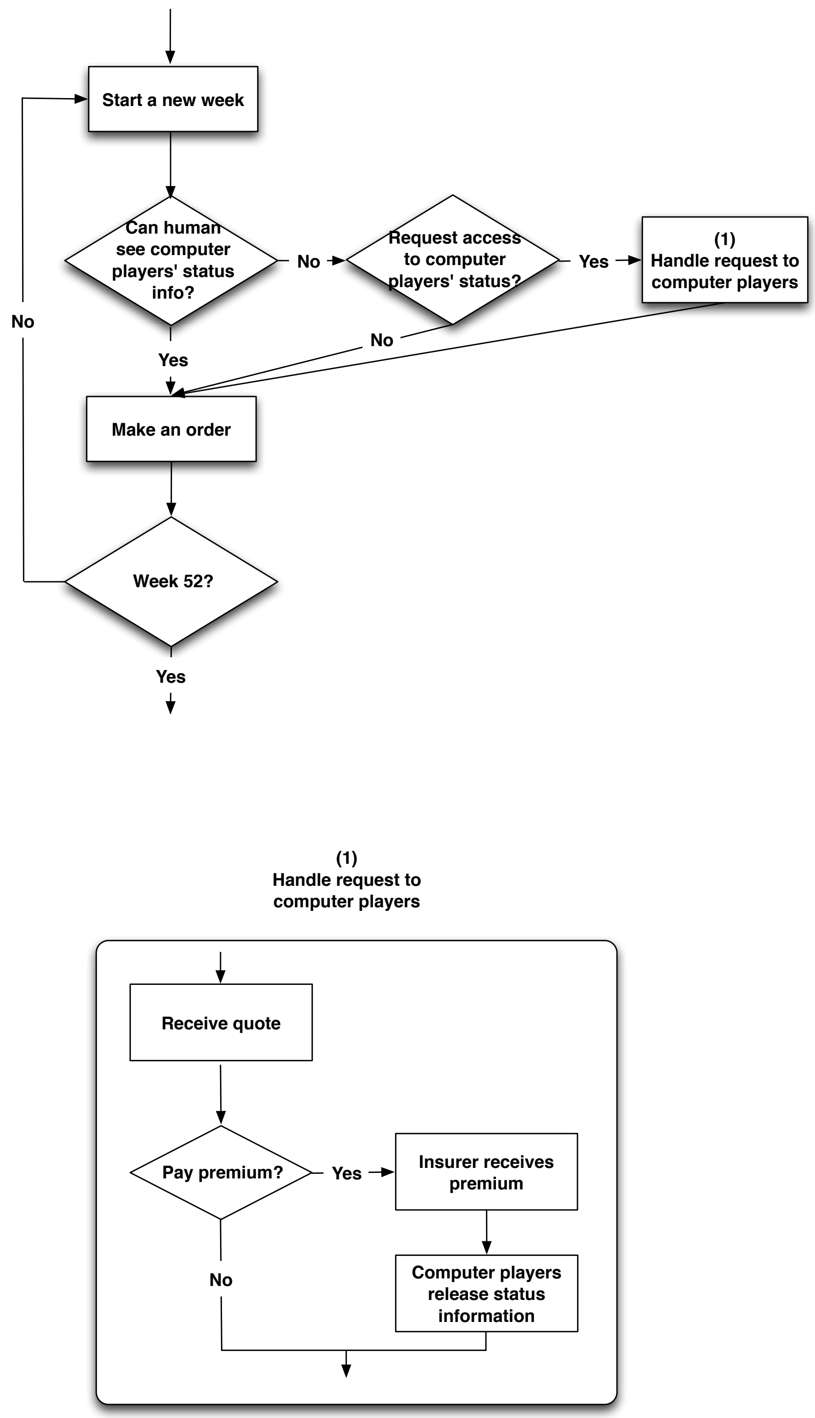


Figure 5.7: Flowchart for Human Player in #321 Insurance, Steady, Distributor

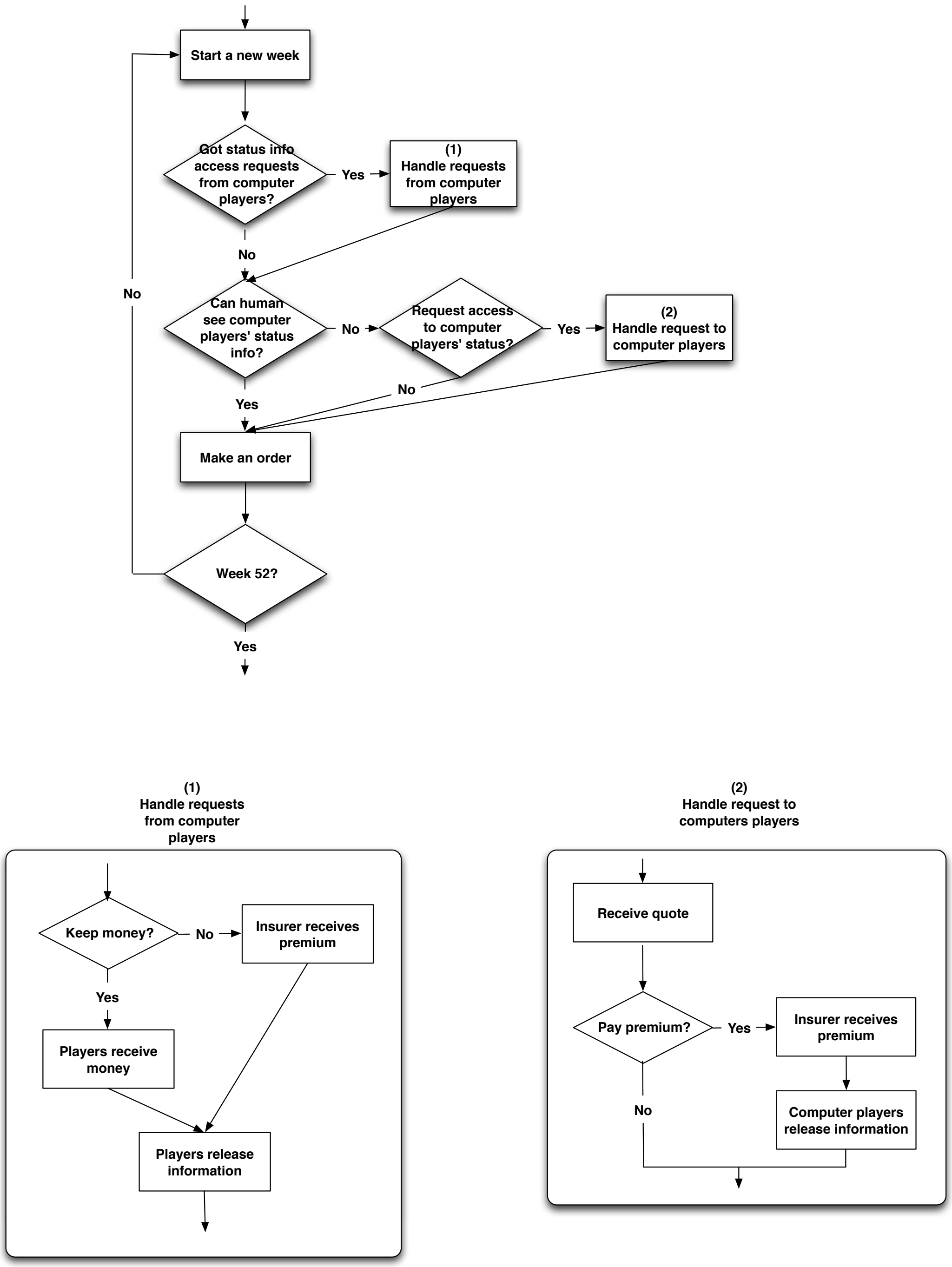


Figure 5.8: Flowchart for Human Player in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

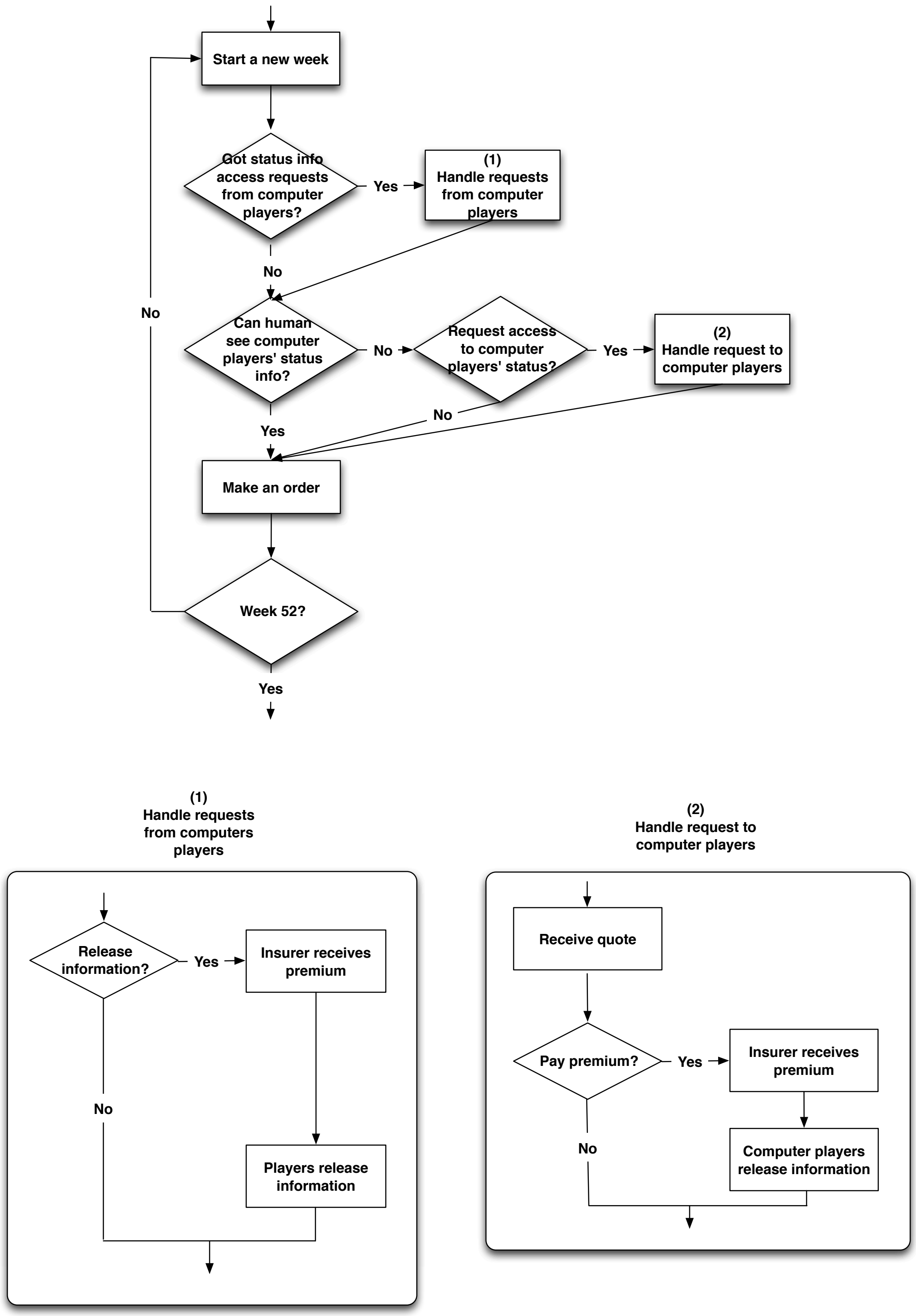


Figure 5.9: Flowchart for Human Player in #521 Deny / Share & Require Policy, Steady, Distributor

cash to players who shared status information instead of buying a policy, the game selects one of the players who shared information randomly and that player incurs the monetary damages. These events are handled automatically by the game software.

The game is set up so that once a player pays a premium, that player can see all the partners' status information. In other words, insurance policies cover all of the partners and premiums are calculated accordingly. Thus, when three of the partners are asked to share their status information with the fourth, the three partners need to agree on whether they are willing to share with the fourth. When the fourth player is the human player, then the other three players are computer players. Because the computer players are rational and reason as outlined above, they always agree to share their status information with the human, and require the human to purchase a policy. When one of the computer players asks the other three to share their status information, then the human and two of the computer players must agree on their decision. In this case, the two computer players always follow the human player's decision.

In version #421, a human player can choose to keep the money offered by a computer player when status information is shared. If the human player chooses to keep the money, then the human player and the other two computer players share their status information with the computer player who requested access. Each of these three players who share information receive one-third of the amount that would have been paid to the insurer if the human player had chosen to make the requester purchase a policy. This is because insurance policies cover all of the partners and premiums are calculated accordingly, as explained above.

At the end of the game in versions #421 and #521, the insurer's final balance is split among the players who opted to use insured access, as a form of profit sharing. The more a player contributes to the insurer's capital by making partners purchase policies when information is shared, the more the player receives from the insurer.

5.2.7 Recruitment, Selection, and Training of Human Players

The Beer Game is hard, even when players can see each others' status. We want the game to be as realistic as possible, within its limitations. Not everyone has what it takes to manage a supply chain reasonably well. If a human player is using a poor strategy to decide how much beer to order, then it is likely that learning the status of other players would not help the human to make better ordering decisions. As an extreme example, if a human randomly picks how much beer to order, then additional information will not affect that process. In real-world supply chain management, however, information about end-customer demand is extremely helpful. Thus, in our experiments, we need to train human players with a very simple version of the Beer Game first, and select reasonably good players who can make use of the other players' status. Then, we can ask such players to play the versions that interest us most.

To exclude not so good players, we used the following qualifications to limit crowdsourcing workers who can accept our HITs. In the list below, "Required for preview" means that only those who meet the qualification can see the details of the HITs we post. In our case, only workers who meet the qualifications can see the recruiting document approved by the IRB (see Appendix A.2).

- HIT Approval Rate (%) for all Requesters' HITs greater than or equal to 95 (Required for preview)
- Number of HITs Approved greater than or equal to 1000 (Required for preview)
- Location is UNITED STATES (Required for preview)

We first posted a test HIT allowing only two workers to submit their assignments for our first version of the game, in order to get to know how Amazon Mechanical Turk works and to make sure that our web app worked as expected. The reward was set to $2 for this initial HIT. After we confirmed that our system worked fine, we posted more HITs with $1 reward. After examining the time players needed to complete a game, we found that they needed around 15 minutes to finish a game on average. With this fact, we decided that a $2 reward is more appropriate for our tasks because it corresponds to $8 per hour rate, which is more than the Amazon Mechanical Turk standard and also matches our online consent document (see Appendix A.1). Complying to this reward rate, we gave $1 bonuses to all workers who played the game with a $1 reward so that they get $2 for one game. We use a $2 reward for the other three versions as well.

Because there are many HITs available on Amazon Mechanical Turk and also because requesters, i.e., people who post HITs on Amazon Mechanical Turk, cannot send workers messages directly, it is not easy for requesters to have target workers complete their tasks. Requesters can, however, give bonuses to workers who have completed tasks, and they can add notes when they do so. We used this feature of granting bonuses to notify workers of the availability of new versions.

We also granted bonuses to workers who achieved especially good results, to encourage workers to play better. We can say that generally time is money for Amazon Mechanical Turk workers, because their base rate of pay is per game, regardless of the outcome. This gives them an incentive to play as fast as possible, regardless of the outcome. However, if the workers make random choices, or don't think hard about their choices, then the results will not be reflective of real-world workplace situations where employees are trying to optimize the outcome for their organization or for the entire VO. We gave bonuses for good performance to try to overcome this tendency.

We also did the following to encourage workers to play better:

- We told them that the best players would be invited back to play additional versions of the game.

- We tried to make the game interesting and engaging to play.

- Because the game can be frustrating and confusing for beginners, who don't understand why the results are so bad, we gave them graphs and statistics at the end of the game that could help them understand the outcomes.

- Because poor partners almost guarantee poor results and player frustration, we implemented reasonably good ordering algorithms for the computer players. As a result, the computer players play better than we do, though probably not as well as the best human players that we found on Amazon Mechanical Turk.

5.2.8 Parameters for Insured Access

There are several parameters specific to version #321 Insurance, Steady, Distributor and the subsequent versions that involve insurance policies. Each week during a game of version #321, players have the option of purchasing an insurance policy that will allow them to see the current status of all players.

Psychological Effects

To determine whether insured access might ever work in the real world, we need to consider some psychological factors when we choose parameter values related to insured access.

The purpose of insurance is to reduce the risk associated with unlikely but very expensive events, such as scandals arising from abuse of shared information. For that reason, potential damages need to be large. There are strong psychological reactions to damage, e.g., having one's car totaled or house burned down, and thus there is the chance that even with insurance, a producer who has experienced these events might want to opt out of the system. In order to make players aware of the large potential damages, we show players the size of damages in our system if a game encounters such events.

A related factor is that humans tend to attach disproportionately large importance to gains and losses rather than to focusing on the final value of their wealth [58]. Thus, the value of shared information should be set at a level that can have a significant impact on consumers' capital. Otherwise, consumers will not be very interested in accessing shared information, with or without insurance. Similarly, damages should be reasonably large compared with producers' and consumers' capital. If damages are very small compared to producers' and consumers' capital, then producers will not be very concerned about being repaid, and consumers will be able to afford to use their own funds to repay producers. Insurance exists to spread risk, so that producers and consumers do not have to fear being wiped out by unexpected calamities.

Table 5.5: Human Decision Reweighting: behavioral economists have found that humans discount the chance of highly likely events and overweight the chance of rare events.

Probability (%)	0	1	2	5	10	20	50	80	90	95	98	99	100
Decision Weight	0	5.5	8.1	13.2	18.6	26.1	42.1	60.1	71.2	79.3	87.1	91.2	100

Table 5.6: Example of Fourfold Pattern of Risk Attitudes

	Gains	**Losses**
High Probability (Certainty Effect)	95% chance to win \$10,000 *Fear of disappointment* RISK AVERSE	95% chance to lose \$10,000 *Hope to avoid loss* RISK SEEKING
Low Probability (Possibility Effect)	5% chance to win \$10,000 *Hope of large gain* RISK SEEKING	5% chance to lose \$10,000 *Fear of large loss* RISK AVERSE

In addition, psychological experiments show that when humans know the probabilities of different outcomes, then they behave as though the probabilities were rather different, as shown in Table 5.5 [58]. Thus, we cannot hope that human players will behave exactly like the ultra-rational decision makers modeled in Chapters 3 and 4.

Our insurance setting falls in one of the fourfold pattern of risk attitudes as shown in Table 5.6 [59]. Specifically, under a low probability of large damages, humans tend to be risk averse in fear of a large loss.

Psychological studies have shown that people are influenced more by exact counts than by percentages. For example, most people find the statement *3,000 people die of foodborne illnesses in the US each year* more compelling than *.001% of the US population dies of foodborne illness each year.* For this reason, we give human players specific numbers for damages and average value, and do not talk about percentages and probability distributions.

There are other psychological factors such as the annoyance of filing claims, not getting all one's damages reimbursed, or the belief that one's business is suffering because of information sharing, even though it has not. Our experiments consider some but not all of these factors, and we will explain how we set the experiments' parameters to reflect these factors.

Scales of Parameter Values

In order to realize insured access in the game, the game needs to use probability distributions internally to represent values of information and claim sizes (risks) involved in the information access. There is one thing we need to be careful about when we decide the parameter values of distributions. As we mentioned earlier in this chapter, we use 100 times larger amounts for costs and incomes in our experiments than those used traditionally in the Beer Game, e.g., the inventory carrying cost per unit in stock in our game is $100 per week instead of $1, because people tend to be psychologically affected by the magnitude of the dollar amounts. The probability distributions used inside the game need to match this scale.

We accomplish this rescaling by using such parameter values for distributions that match the traditional beer costs and income internally, then multiplying the sample values from the distributions by 100 before presenting them to the user. As explained below, technical issues dictate that we take this approach instead of revising the parameter values for distributions used internally to match the larger scale.

As explained in detail later, we scale down the parameter values for the damage distributions linearly as the game proceeds, so that the premium calculated from the damage distributions becomes smaller as the game proceeds. As also explained later, we use Zero-Adjusted Gamma (ZAGA) distributions for claims. With ZAGA distributions, we found that the premium doesn't change much as the game proceeds if we use parameter values that match the 100 times larger cost/income amounts than traditionally used. However, we want to make the premium more reasonable as the game proceeds so that a human player may decide to purchase a policy sometime during the game with a reasonable price, even if they don't want to pay the high price for the policy in the first few weeks of the game. For this reason, we use such parameter values for distributions that match the traditional amounts, and calculate premiums from distributions with these parameter values. Then, we multiply the premium by 100 to match the scales of the costs and incomes in our games. We found that the price declines satisfactorily over time if we set parameters this way.

Table 5.7: Parameters Specific to Versions with Insurance

Parameter	Value
insurer's utility function	Exponential Utility
parameter α for insurer's exponential utility	0.01
initial maximum premium a player is willing to pay	\$14,000
distributions for claims	Zero-Adjusted Gamma (ZAGA)
initial probability of positive size claims happening	0.1
initial shape k of ZAGA for claims	360
scale θ of ZAGA for claims	0.99

Parameter Values Used in Experiments

Table 5.7 shows the values of the parameters specific to game versions with insurance, i.e., #321, #421, and #521. We explain each parameter value below.

First, we need to introduce another principal: the insurer. Because the insurer needs to behave rationally to quote premiums, it is played by the computer. We use an exponential utility function for the insurer so that we use the exponential premium principle that is widely used and known to have favorable properties, as explained in Chapter 4. Regarding the risk aversion index α for the insurer, we want to use a relatively small value because we want to have a reasonable policy price so that rational players would be willing to purchase policies to make use of status information. We found 0.01 is a reasonable value to be used with the other parameter values for this reason. We will explain how to set parameter values related to premiums in detail later.

We set the initial capital of the insurer to \$0 as we do for the other players in the game, because investigating ruin is not the focus of the crowdsourcing experiments. Because of this setting, the insurer may be ruined in some games. In real applications of insured access, as we discussed in detail in the previous chapter, VOs must start with sufficient capital for their planned portfolio size and take corrective action if the ruin probability approaches the VO's cap. In the crowdsourcing experiments, games continue even if the insurer is ruined, i.e., its cash balance becomes below \$0.

As shown in Table 5.4, computer players can achieve much better outcomes if they know the actual end-customer demand and use it in their formulas, instead of calculating an expected demand and using that to decide how much to order. Specifically, we can say that the information is worth \$152,960.80 = \$397,801.40 - \$244,840.60 on average for the four computer players. Because the optimal values for the two adjustment parameters (γ, β) in the SMS ordering strategy are derived considering the supply chain as a whole, we can assume that the status information is worth \$38,240.20 = \$152,960.80 / 4 per player on average. Technically, though, this is not necessarily true in our case because this difference comes from the fact that the computer players can use the actual customer demand instead of the expected demand in the SMS strategy, and because the retailer always knows the actual customer demand. In other words, there is no new information for the retailer, and thus the information is worth \$0 for the retailer and it's worth \$50,986.93 = \$152,960.80 / 3 on average for the other three players. In general, however, ordering strategies can take into account the status information of all players. Thus, we assume that we can equally divide \$152,960.80 by the four computer players, i.e., the status information is worth \$38,240.20 per player on average.

Then, we can use Formula 4.5 from Chapter 4 to calculate the maximum premium a rational player will be willing to pay. For Y in Formula 4.5, which is a random variable representing the consumer's expected additional value (or revenue) from accessing the information, we use a normal distribution with a mean of \$382.402 and the standard deviation of \$108.375, which are calculated from the results of the tournament with four computer players. Note that the scales of these parameter values for the distribution match the traditional cost/income amounts, and we multiply the maximum premium calculated by Formula 4.5 with these parameters by 100, as we explained earlier. We use an exponential utility function for a player, i.e., an information consumer, because of its favorable properties. Regarding its α_c, we want to use a relatively small value because the games will be uninteresting if computer players never choose to purchase policies, and we found 0.01 is a reasonable value to be used with the other parameter values. With these parameters, Formula 4.5 gives \$14,057.06 as the maximum premium, and we use a rounded value \$14,000 for the maximum premium a computer player is willing to pay in our games.

Now, we discuss the risk side. As already explained, detailed information about players' orders and inventory must be very sensitive and it could be misused by players' partners. For example, if a company is developing a backlog, then its best customer might decide to order from one of its competitors who promises faster delivery. If a company has too much stock in its warehouse, its customer might threaten to order from its competitor unless it cuts prices. A company's competitors might learn about its ordering strategy and imitate it. To model these situations, we must select a probability distribution to represent the monetary damages caused by these kinds of misuse of information.

As we discussed earlier, damages should be large, both because that's what people worry about for information sharing in the real world, and because insurance only makes sense in that setting. Two scenarios are of potential interest: one where damages are so high on average and thus the premium for accessing information that causes such damages is so high that a rational human should not buy the policy, and one where they would. Because we are more interested in scenarios where ultra-rational people would buy policies, we try to set parameter values such that they would do so. To keep damages high while trying to make human players purchase policies, we want to set a premium close to the maximum premium a computer player is willing to pay, which is $14,000 as we explained above. Another reason why a premium close to $14,000 is favorable is that $14,000 sounds very large to human players, which is important for psychological reasons. Because the player's initial capital is $0, losing $14,000 to purchase a policy at the beginning of the game is psychologically painful. Considering these factors, we set parameter values so that the premium becomes close to $14,000.

For the distribution for monetary damages, which will be claim sizes, we use the Zero-Adjusted Gamma (ZAGA) distribution because it can incorporate the probability of no claims well and also because it belongs to the exponential family, which is widely used for modeling claims. The chance of damages might be extremely low in the real world, although the consequences could be catastrophic if damage does occur. However, this situation is tricky to model realistically in games with humans. If most humans never encounter damages, then that aspect of information sharing will have no psychological impact on them. Thus, we set the claim probability to 0.1, which is not extremely low, but reasonably low to represent rare events. In addition, humans tend to assign relatively large decision weights to outcomes that have this probability, as shown in Table 5.5, so that a player would think that insurance policies are worth purchasing.

When a player shares status information, the game determines whether there will be monetary damages, based on the claim probability. When there should be monetary damages by chance based on the claim probability, we choose a week from the remaining weeks uniformly randomly and handle the event in that week. These events are handled automatically by the game software.

The shape k and scale θ of the distribution are set to 360 and 0.99, respectively. Note that the scales of the parameter values for the ZAGA distribution shown in Table 5.7 match the traditional cost/income amounts, and we multiply the premium calculated from the distribution with these parameters by 100, as we explained earlier. These parameter values are chosen so that the premium becomes close to $14,000.

Because a policy covers until the end of the game, we can intuitively say that the earlier a policy is purchased during a game, the more helpful the information is. As for the risk involved in the information access, we have two options. The first is to make the risk associated with an access apply to a certain interval of time after the access (e.g., one year), even though the abuse of that information might take place after the one-year scope of the game. This accurately models the potential for abuse of information in many real-world settings. However, it is a poor match for a game setting, where players will be judged by their performance at the end of a fixed fiscal period. The players have to pay the policy premium up front in a single installment, even though much of the period covered by the policy falls into the next fiscal period, i.e., extends beyond the end of the game, and might more appropriately be payable in a second installment in that next fiscal period. For this reason, we scale down the parameter values for the damage distributions linearly as the game proceeds. More precisely, we multiply some of the parameter values in Table 5.7 by (the number of remaining weeks / the duration of the game, which is 52). The parameters to which we apply this rule are the maximum premium a player is willing to pay ($14,000), the probability of positive size claims happening (0.1), and the shape k of ZAGA for claims

Table 5.8: Final Supply Chain Cash Balances in #111 See All, One-Step, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	180	-$420,796.67	$56,500.00	$2,401,660.79
	Best	168	-$436,609.52	$60,900.00	$2,483,979.51
Excluded	All	179	-$250,762.01	$58,600.00	$753,008.04
	Best	167	-$254,451.50	$61,700.00	$774,107.64

(360).

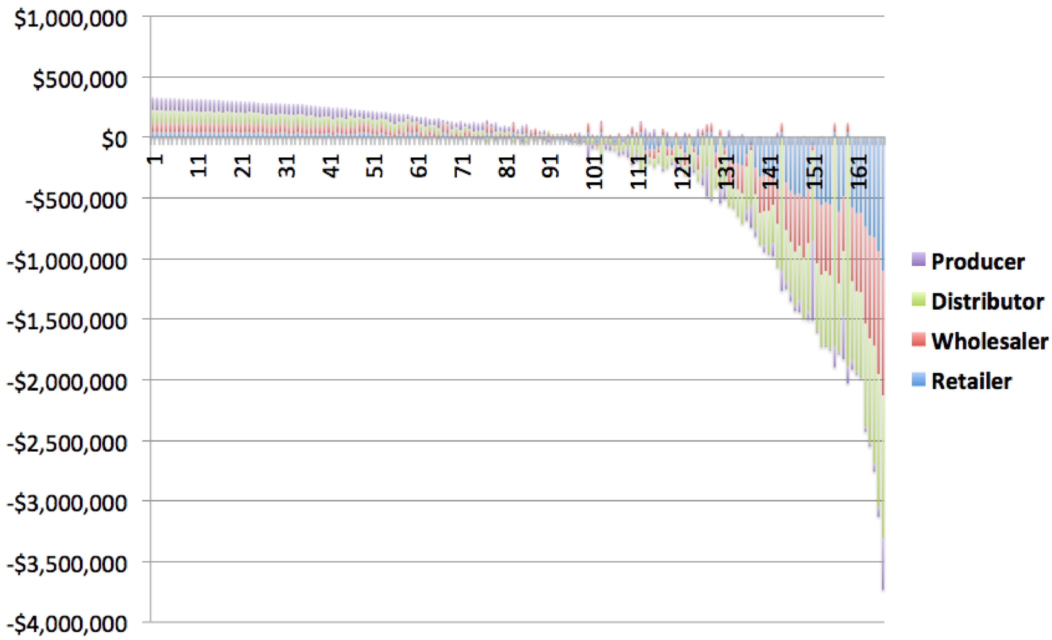
Using these parameter values, the premium in week 1 is $14,400, and computer players purchase policies in week 3 when the premium becomes $13,000.

5.3 Experimental Results

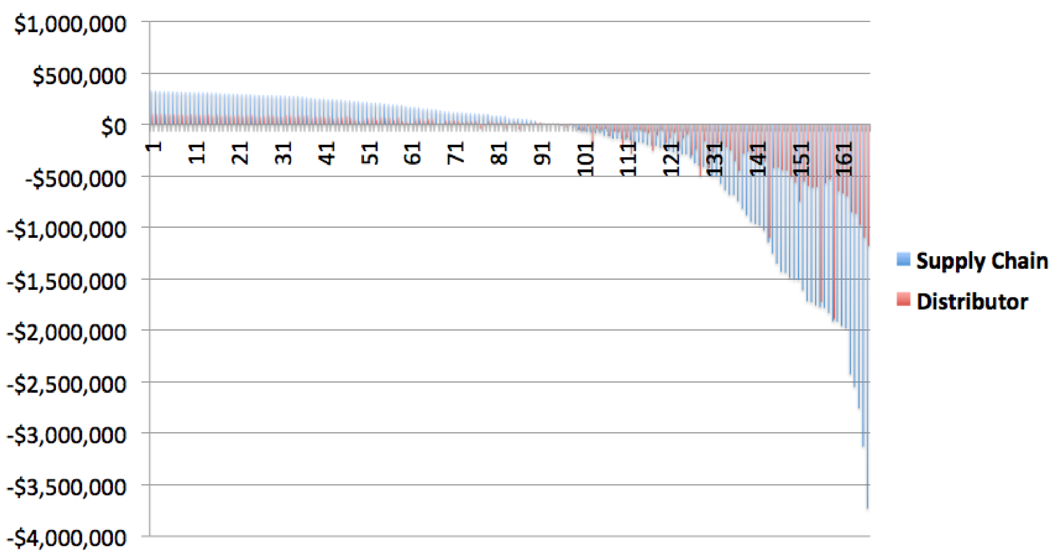
5.3.1 #111 See All, One-Step, Distributor

We use this version solely for recruiting the initial set of human players and training them. In this version, a player can see the status of the others freely. The end-customer demand pattern is One Step Demand, in which the demand is four crates until week 4 and then steps to eight crates after that. For this version only, we use the Pass-through ordering strategy. We showed the players the text in Appendix C.1 before they played this version of the game. We got the comments for this version shown in Appendix D.1.

We collected 180 data points, i.e., completed games from paid Amazon Mechanical Turk workers, for this version. Some players played this version multiple times, and if we retain only the best result for each player, we collected 168 data points. Figure 5.10 shows final cash balances of the four positions in a stacked column chart and and those of the supply chain and distributor in a clustered column chart. The supply chain's balance is just the sum of the four positions' balances. The data points are sorted by the supply chain balance in Figure 5.10. The charts omit an outlier whose final supply chain cash balance was -$30,857,000. For players who played this version multiple times, the charts show only the best result, i.e., the highest supply chain balance. Thus, the charts have 167 data points. Figure 5.11 shows the same results, but the data points are sorted by the distributor balance.

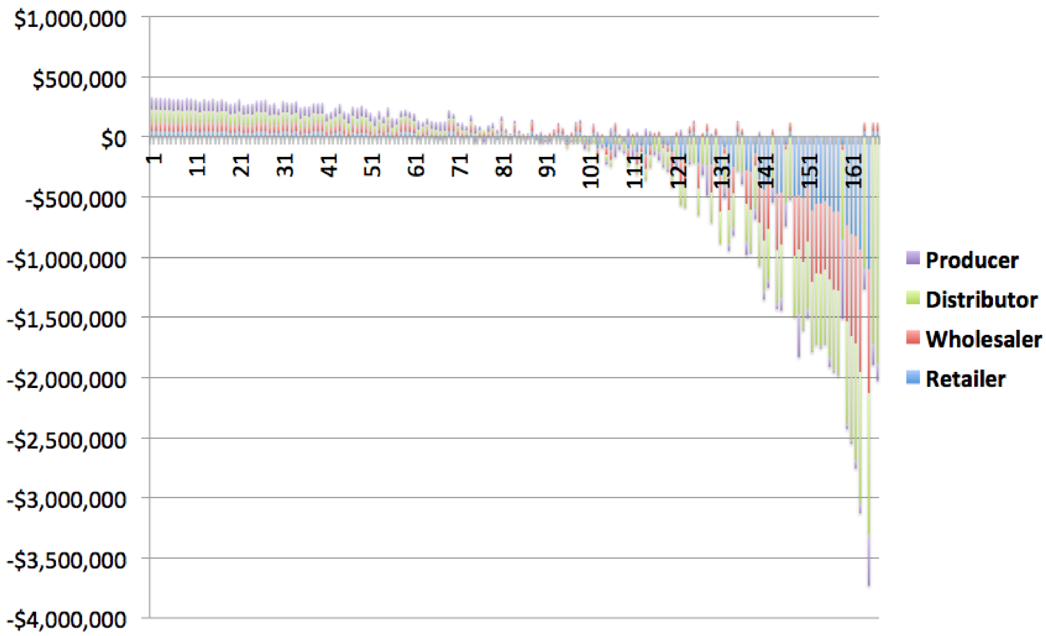


(a) Final Cash Balance of Each Position

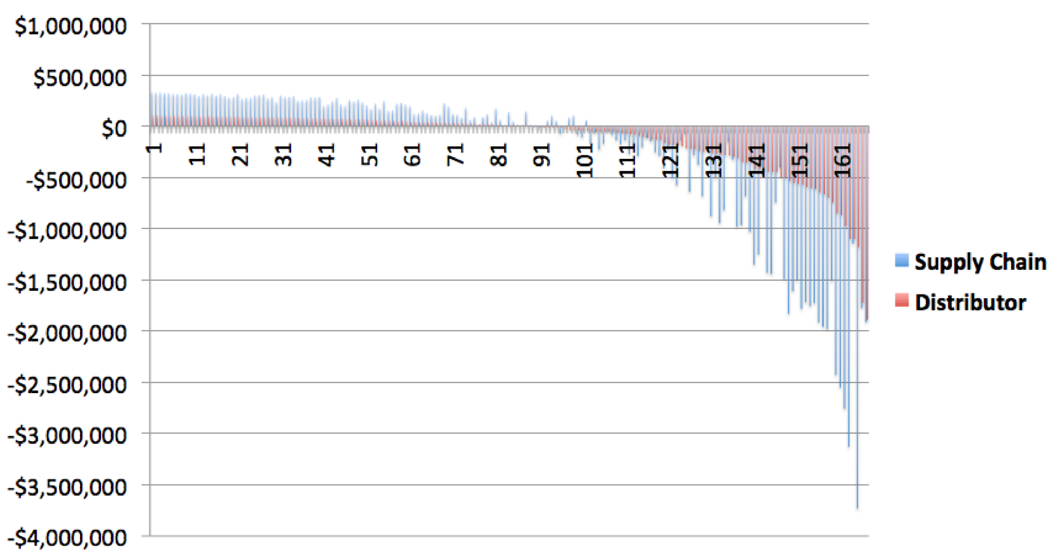


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.10: Final Cash Balances for 167 Players in #111 See All, One-Step, Distributor [Sorted by Supply Chain Balance]



(a) Final Cash Balance of Each Position



(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.11: Final Cash Balances for 167 Players in #111 See All, One-Step, Distributor [Sorted by Distributor Balance]

Table 5.9: Final Distributor Cash Balances in #111 See All, One-Step, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	180	-\$273,072.22	\$7,950.00	\$2,018,386.76
	Best	168	-\$287,507.14	\$10,050.00	\$2,088,686.88
Excluded	All	179	-\$124,510.06	\$8,900.00	\$318,942.48
	Best	167	-\$128,356.29	\$10,600.00	\$328,615.90

Table 5.10: Bonus Granted in #111 See All, One-Step, Distributor

Range of Supply Chain Balance (B_{sc})	% Workers	# Workers	Bonus Granted
$B_{sc} \geq \$300,000$	Top 11%	19	\$2.00
$\$300,000 > B_{sc} \geq \$200,000$	Top 11 - 33%	36	\$1.50
$\$200,000 > B_{sc} \geq \$0$	Top 33 - 57%	40	\$1.00

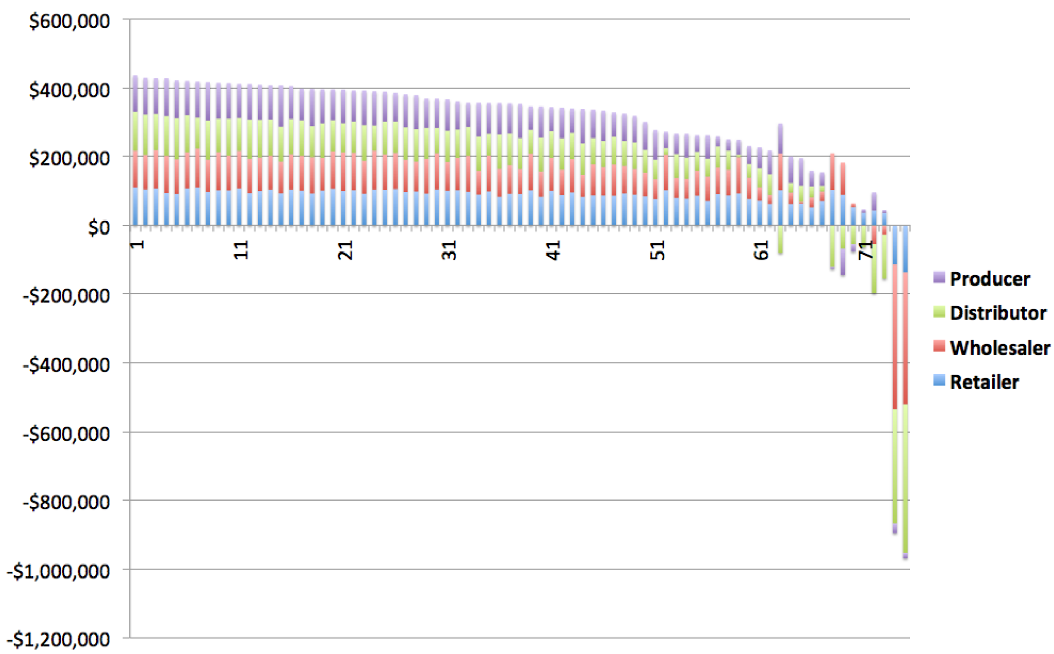
Table 5.8 summarizes statistical information about the final supply chain cash balances for this version. The negative average values illustrate the difficulty of the Beer Game, but over a hundred players achieved positive supply chain balances. Because we need reasonably good players for the remaining versions of our experiments, we allowed only those who achieved positive supply chain balances in this version to play the next version. Table 5.9 summarizes statistical information about the final distributor cash balances.

Table 5.10 summarizes the bonus amounts we granted in this version.

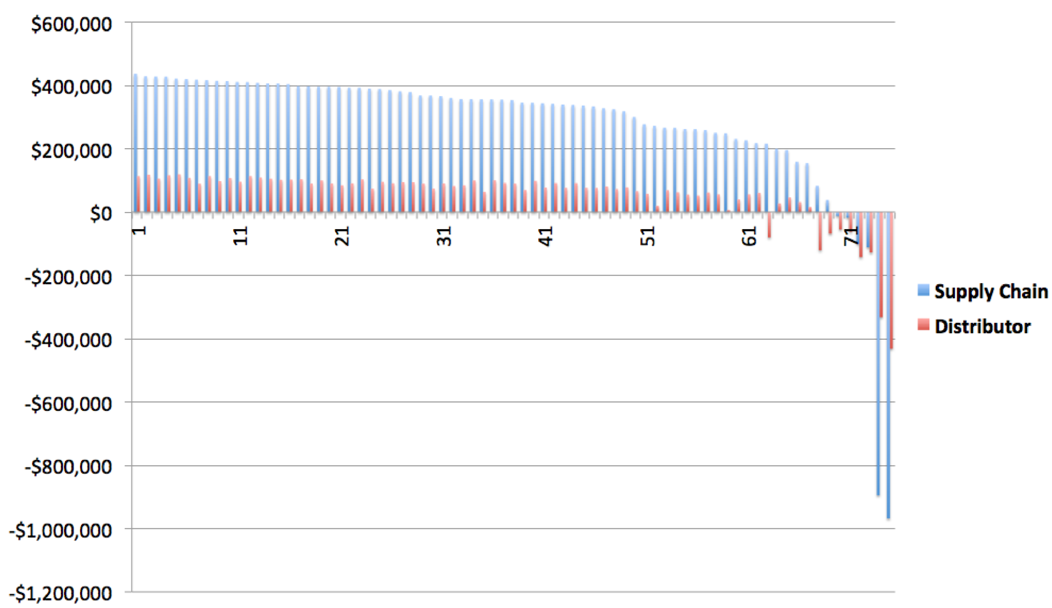
5.3.2 #121 See All, Steady, Distributor

We use this version for training the human players with the Steady demand pattern. We are also interested in the differences between the results of this version and those of the next See Only Yours version, because we would like to examine how valuable the status information is. Only those who received bonuses for the previous version can accept our HITs for this version.

In this version, a player can see the status of the others. The end-customer demand pattern is Steady, in which the mean demand remains constant over time. We use the γ and β values shown in Table 5.11. Before their game begins, we show players the tutorial text in Appendix C.2. We got the comments for this version shown in Appendix D.2.

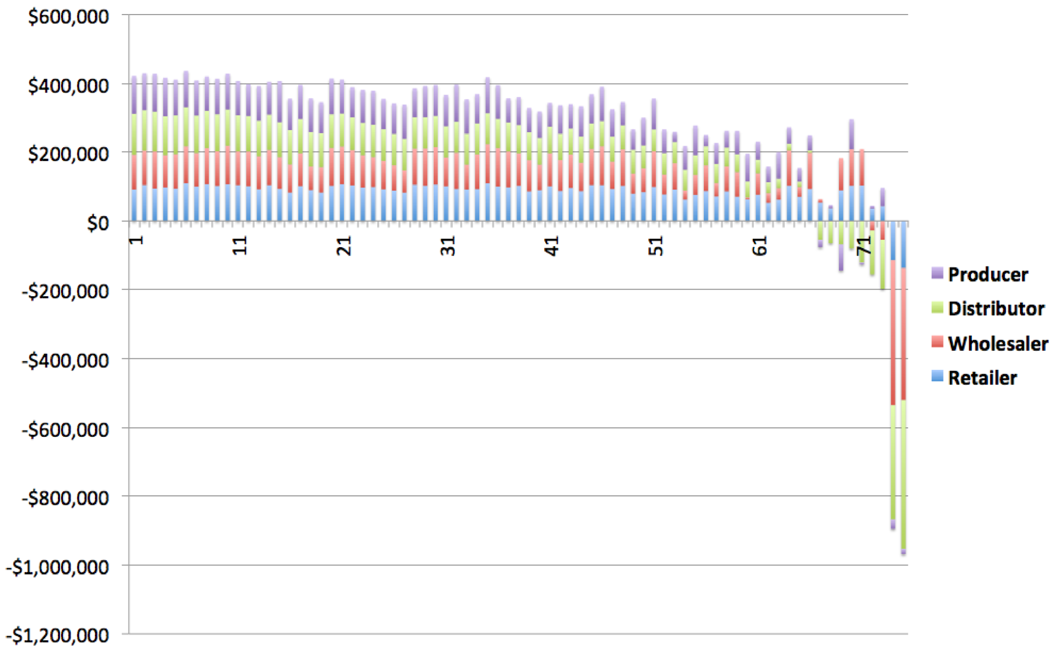


(a) Final Cash Balance of Each Position

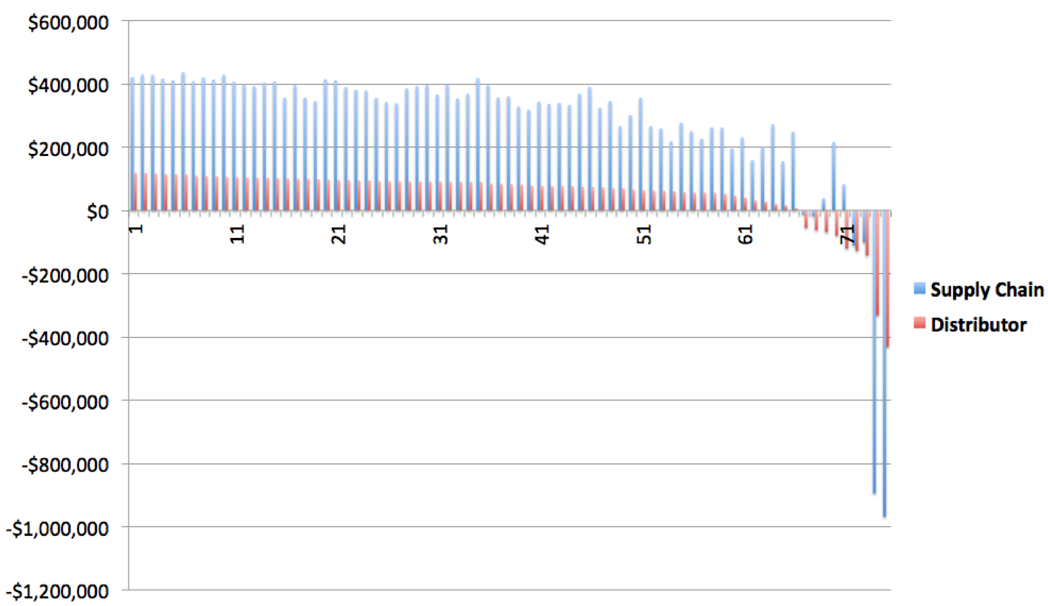


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.12: Final Cash Balances for 75 Players in #121 See All, Steady, Distributor [Sorted by Supply Chain Balance]



(a) Final Cash Balance of Each Position



(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.13: Final Cash Balances for 75 Players in #121 See All, Steady, Distributor [Sorted by Distributor Balance]

Table 5.11: γ and β in #121 See All, Steady, Distributor

Parameter	Retailer	Wholesaler	Distributor	Producer
γ	0.2215	0.0823	0.1602	0.8437
β	0.7832	0.7577	0.7342	0.7167

Table 5.12: Final Supply Chain Cash Balances in #121 See All, Steady, Distributor

Dataset	# Data Points	Average	Median	Standard Deviation
All	91	$254,470.33	$345,900.00	$299,269.83
Best	75	$279,192.00	$353,500.00	$235,852.33

We collected 91 data points for this version, representing 91 games played by 75 humans. Figure 5.12 shows final cash balances of the four positions in a stacked column chart and and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. For players who played this version multiple times, the charts show only the best result. Thus, the charts have 75 data points. Figure 5.13 shows the same results, but the data points are sorted by the distributor balance. Table 5.12 summarizes statistical information regarding the final supply chain cash balances for this version, and Table 5.13 summarizes statistical information regarding the final distributor cash balances.

Even with the harder demand pattern used in this version, most players achieved positive supply chain balances and the average is much better than that in the previous version. Our own experience with the Beer Game showed that practice pays off, and we expect that the human players' improvements are due to their increasing familiarity with the game. Because the remaining versions become even harder, we decided to exclude players who failed to achieved positive supply chain balances in this version.

Table 5.14 summarizes the bonus amounts we granted in this version. There was one

Table 5.13: Final Distributor Cash Balances in #121 See All, Steady, Distributor

Dataset	# Data Points	Average	Median	Standard Deviation
All	91	$35,730.77	$80,800.00	$168,033.14
Best	75	$52,668.00	$82,200.00	$92,823.043

Table 5.14: Bonus Granted in #121 See All, Steady, Distributor

Range of Supply Chain Balance (B_{sc})	% Workers	# Workers	Bonus Granted
$B_{sc} \geq \$400,000$	Top 21%	16	\$2.00
$\$400,000 > B_{sc} \geq \$350,000$	Top 21 - 51%	22	\$1.50
$\$350,000 > B_{sc} \geq \$0$	Top 51 - 91%	30	\$1.00

Table 5.15: γ and β in #221 See Only Yours, Steady, Distributor

Parameter	Retailer	Wholesaler	Distributor	Producer
γ	0.0267	0.6076	0.1693	0.9726
β	0.2753	0.7578	0.6992	0.7391

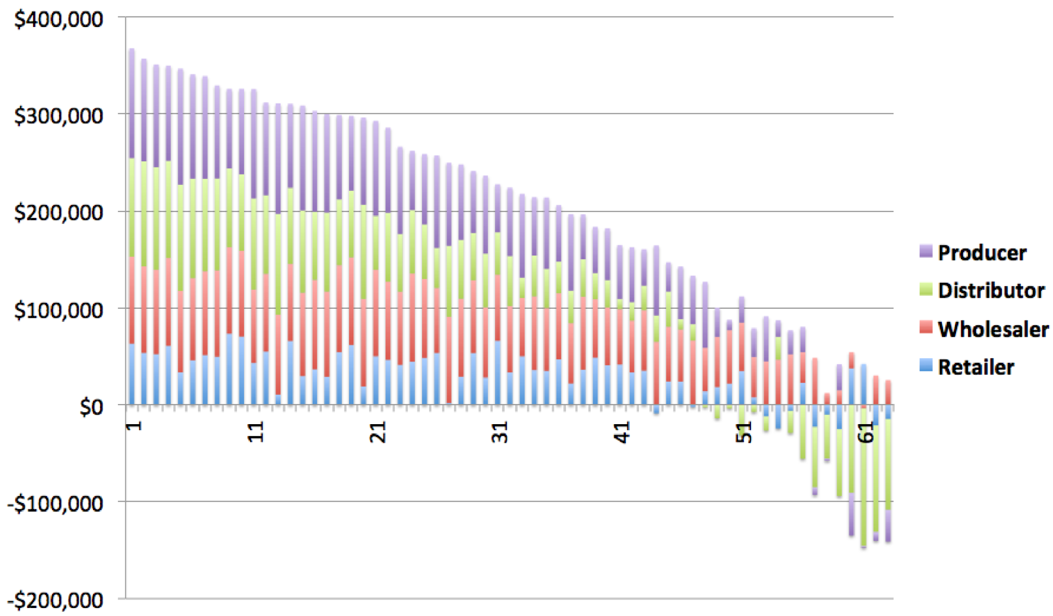
more worker in the range of $\$350,000 > B_{sc} \geq \$0$, but that worker didn't submit their assignment and we could not grant them a bonus.

5.3.3 #221 See Only Yours, Steady, Distributor

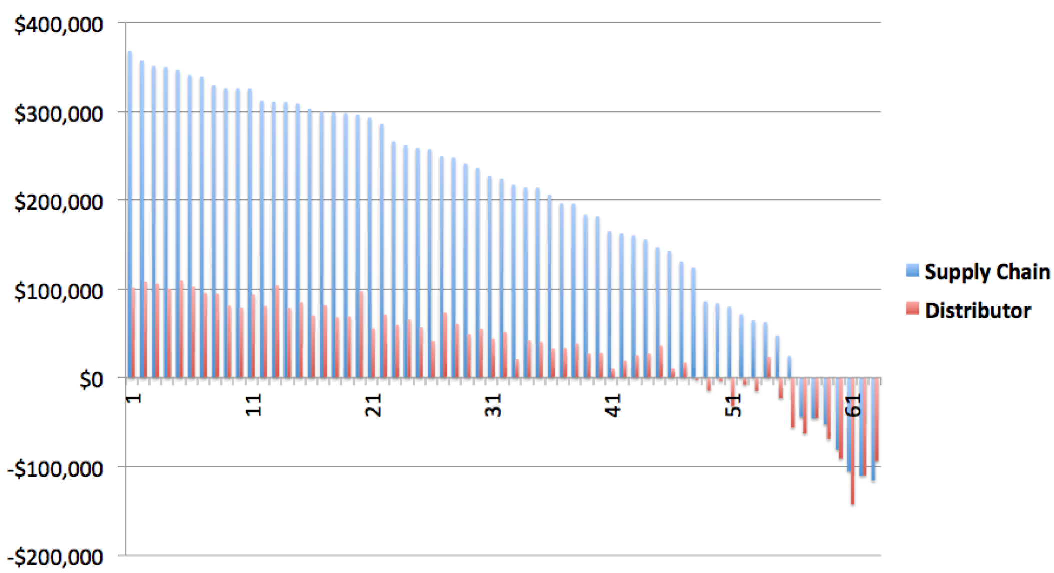
We use this version for training the human players without allowing them to see the status of the others. We are also interested in the differences between the results of this version and those of the previous See All version, because we would like to examine how valuable the status information is. Only those who received bonuses for the previous version can accept our HITs for this version.

In this version, a player can see only their own status. The end-customer demand pattern is Steady, in which the mean demand remains constant over time. We use the Stock Management Structure (SMS) ordering strategy. In this version, we use the γ and β values shown in Table 5.15. Before their game begins, we show players the tutorial text in Appendix C.3. We got the comments for this version shown in Appendix D.3.

We collected 88 data points for this version, representing 88 games played by 63 humans. Figure 5.14 shows final cash balances of the four positions in a stacked column chart and and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. For players who played this version multiple times, the charts show only the best result. Thus, the charts have 63 data points. Figure 5.15 shows the same results, but the data points are sorted by the distributor balance. Table 5.16

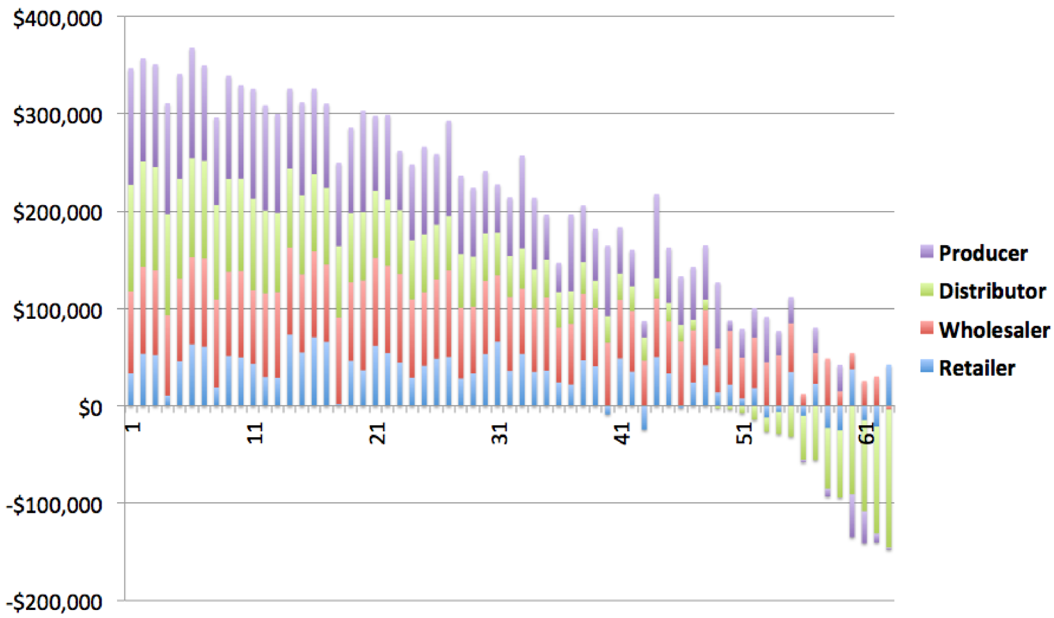


(a) Final Cash Balance of Each Position

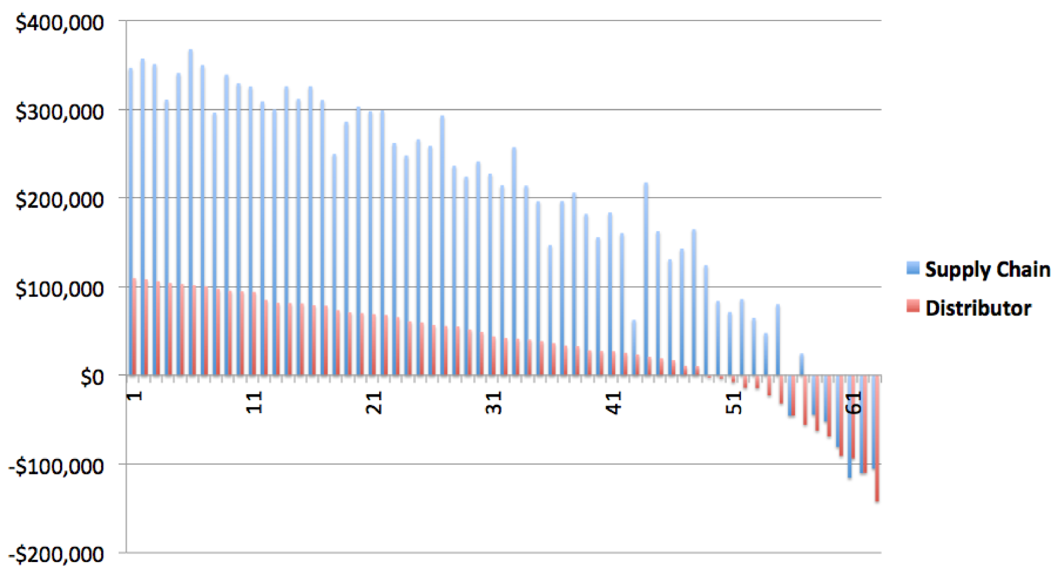


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.14: Final Cash Balances for 63 Players in #221 See Only Yours, Steady, Distributor [Sorted by Supply Chain Balance]



(a) Final Cash Balance of Each Position



(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.15: Final Cash Balances for 63 Players in #221 See Only Yours, Steady, Distributor [Sorted by Distributor Balance]

Table 5.16: Final Supply Chain Cash Balances in #221 See Only Yours, Steady, Distributor

Dataset	# Data Points	Average	Median	Standard Deviation
All	88	$130,361.36	$209,850.00	$288,590.15
Best	63	$195,368.25	$223,900.00	$132,153.27

Table 5.17: Final Distributor Cash Balances in #221 See Only Yours, Steady, Distributor

Dataset	# Data Points	Average	Median	Standard Deviation
All	88	-$244.32	$37,500.00	$165,272.89
Best	63	$34,282.54	$42,000.00	$58,373.34

summarizes statistical information regarding the final supply chain cash balances for this version, and Table 5.17 summarizes statistical information regarding the final distributor cash balances.

As expected, the games resulted in smaller final supply chain cash balances in this version than those in the previous version because the players could not see the status of the others. In order to examine this phenomenon more closely, we subtracted the final supply chain balance for this See Only Yours version from that for the previous See All version for each player. Figure 5.16 shows the differences for the 62 players who played both versions. The data points are sorted by the amount of the difference. The figure omits an outlier whose final supply chain balance in version #121 was only $83,000, resulting in a large negative difference value -$225,600.

We also performed a t-test and confirmed that the final supply chain balance in version #121 See All, Steady, Distributor is statistically significantly larger than that in version #221 See Only Yours, Steady, Distributor at the 1% significance level. We also examined the difference in the tournament with four computer players and observed similar results. Table 5.18 summarizes statistical information for the difference.

Because we want to see how the introduction of insured access affects the results, we decided to allow all players who played this version to play the next version, too.

Table 5.19 summarizes the bonus amounts we granted in this version.

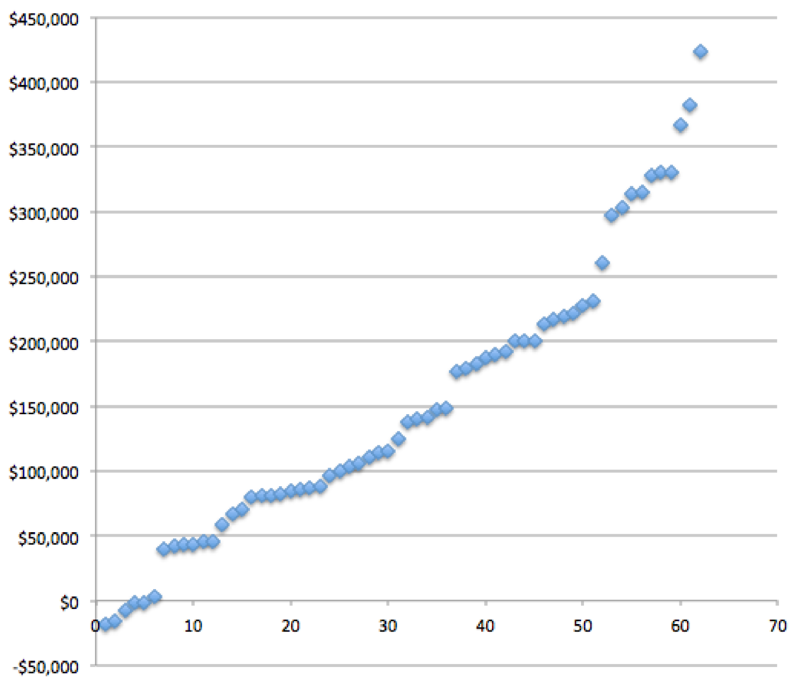


Figure 5.16: Difference of Final Supply Chain Balance Between Version #121 and #221 for Each Individual Human Player

Table 5.18: Difference of Final Supply Chain Balance Between Version #121 and #221

Dataset	# Data Points	Average	Median	Standard Deviation	P-value
Humans with Outliers	63	$144,930.16	$124,200.00	$117,577.36	1.7243E-14
Humans without Outliers	62	$150,906.45	$131,200.00	$108,462.27	2.4224E-16
Four Computer Players	1,000	$152,960.80	$158,300.00	$43,350.19	≈ 0

Table 5.19: Bonus Granted in #221 See Only Yours, Steady, Distributor

Range of Supply Chain Balance (B_{sc})	% Workers	# Workers	Bonus Granted
$B_{sc} \geq \$300,000$	Top 25%	16	$2.00
$\$300,000 > B_{sc} \geq \$200,000$	Top 25 - 57%	20	$1.50
$\$200,000 > B_{sc}$	Top 57 - 100%	27	$1.00

Table 5.20: γ and β in #321 Insurance, Steady, Distributor

Policy	Parameter	Retailer	Wholesaler	Distributor	Producer
Not Purchased	γ	0.0267	0.6076	0.1693	0.9726
	β	0.2753	0.7578	0.6992	0.7391
Purchased	γ	0.2215	0.0823	0.1602	0.8437
	β	0.7832	0.7577	0.7342	0.7167

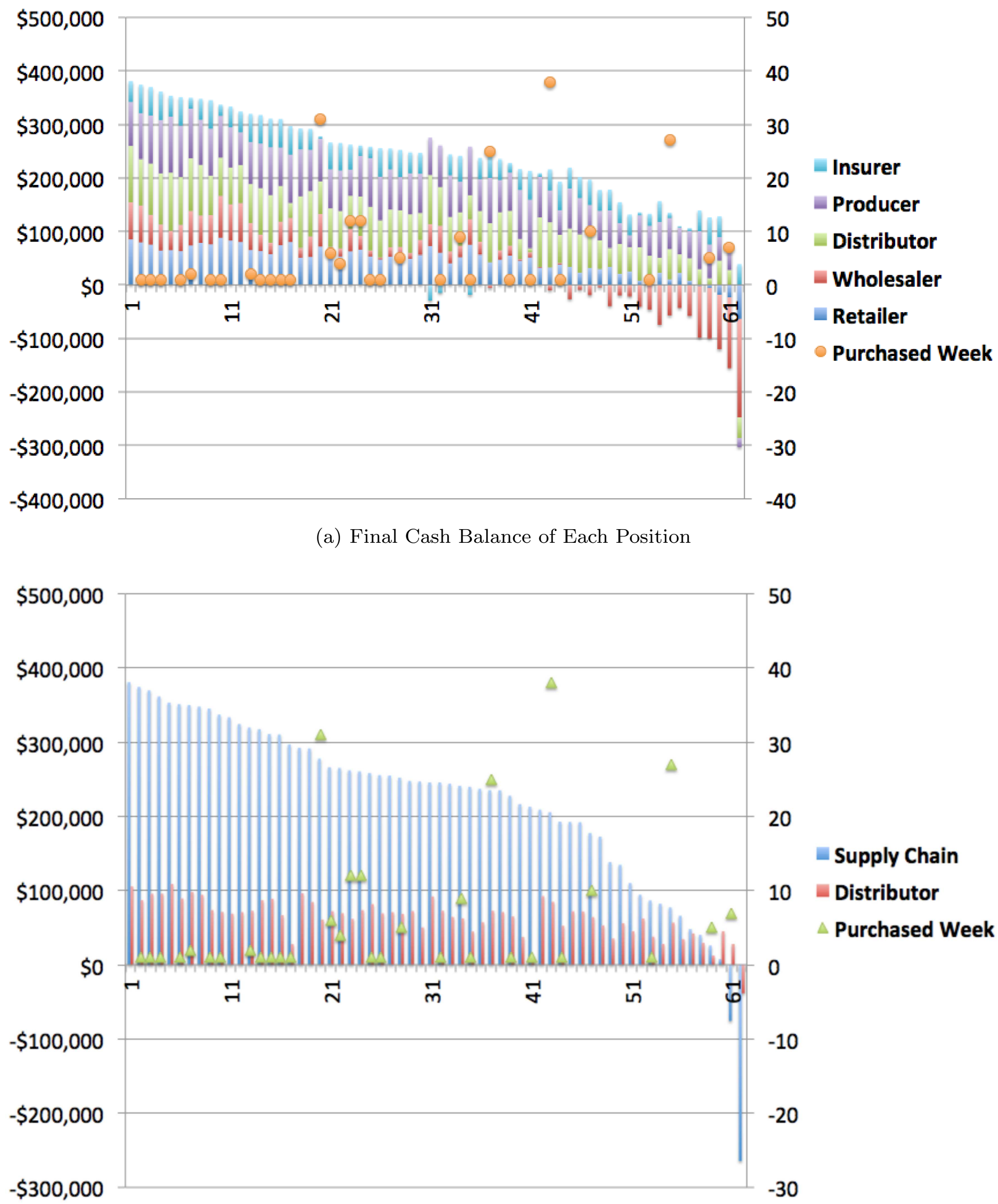
5.3.4 #321 Insurance, Steady, Distributor

This version is instrumental for determining whether human players behave like their ultra-rational counterparts, and use insured access to improve the outcome of their games. As we mentioned in the end of the previous section, we allowed all players who played version #221 to accept our HITs for this version.

In this version, players can see the status of the others freely only if they purchase an insurance policy. For the sake of simplicity, we set up this version so that an insurance policy covers access to other players' status information for the remainder of the game if a player purchases a policy once. The end-customer demand pattern is Steady, in which the mean demand remains constant over time. We use the Stock Management Structure (SMS) ordering strategy. In this version #321 Insurance, Steady, Distributor, we use the γ and β values shown in Table 5.20.

Before their game begins, we show players the tutorial text in Appendix C.4. Note that the text says that the supply chains made about \$115,000 more on average when players could see all information in the chain, but this was a temporary value calculated with about half of the data points. After collecting all data, this difference was \$150,906.45 on average without outliers as we showed earlier. We got the comments for this version shown in Appendix D.4.

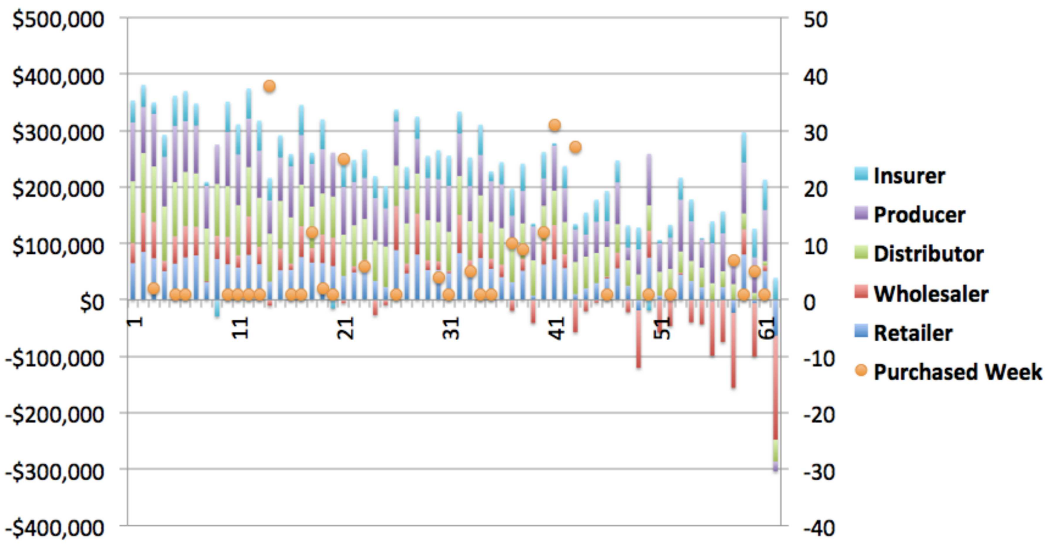
We collected 75 data points for this version. If we retain only the best result for each player, we collected 63 data points. Figure 5.17 shows final cash balances of the four positions and insurer in a stacked column chart and and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. The charts omit an outlier whose final supply chain cash balance was -\$1,805,500. For players



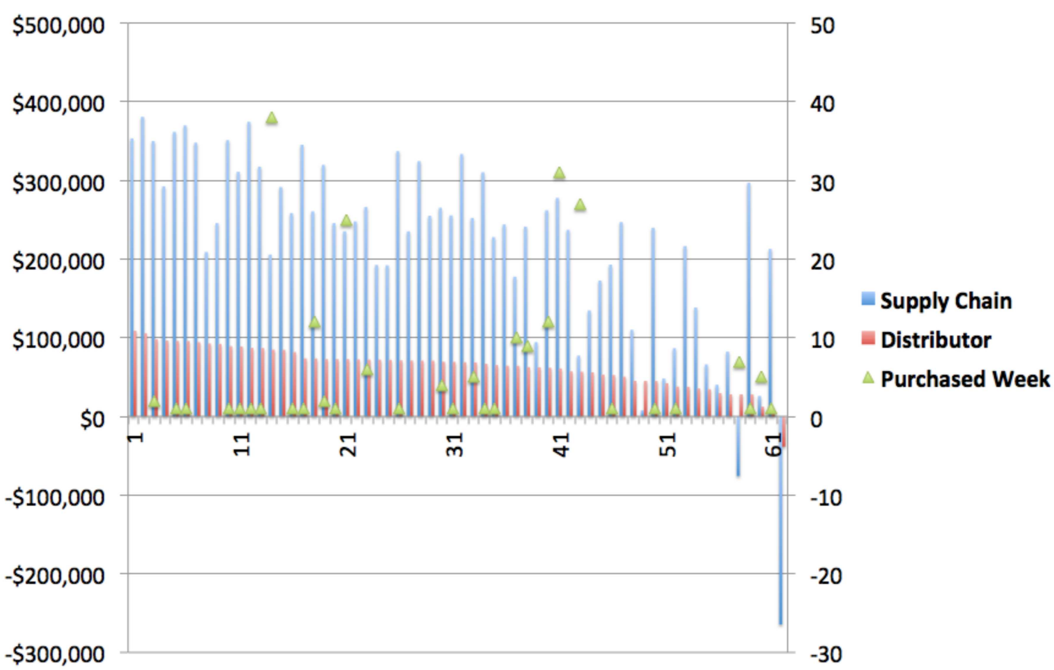
(a) Final Cash Balance of Each Position

(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.17: Final Cash Balances for 62 Players in #321 Insurance, Steady, Distributor [Sorted by Supply Chain Balance]

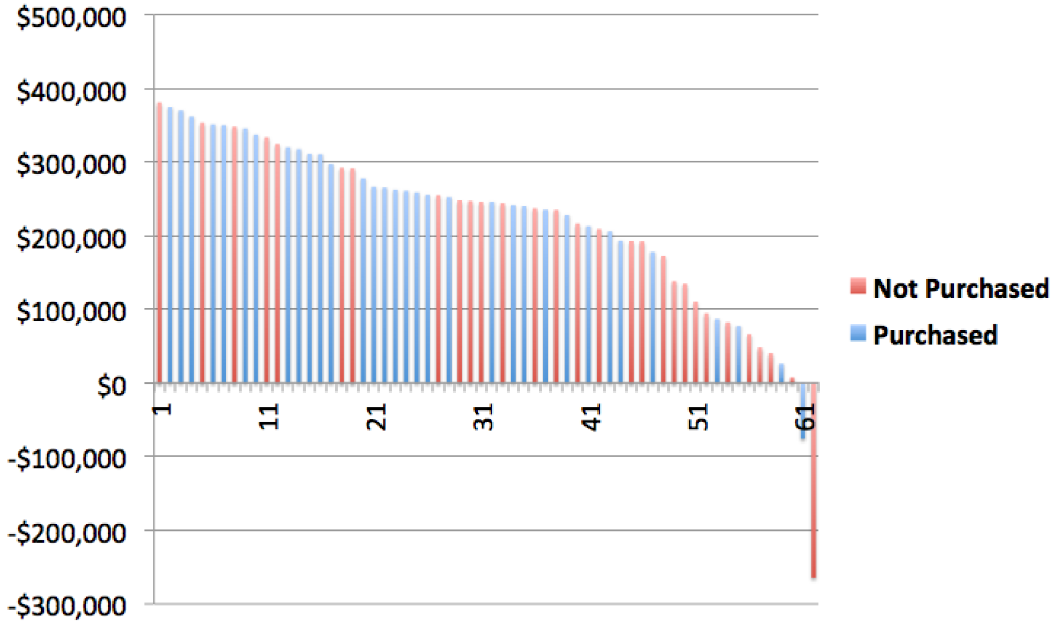


(a) Final Cash Balance of Each Position

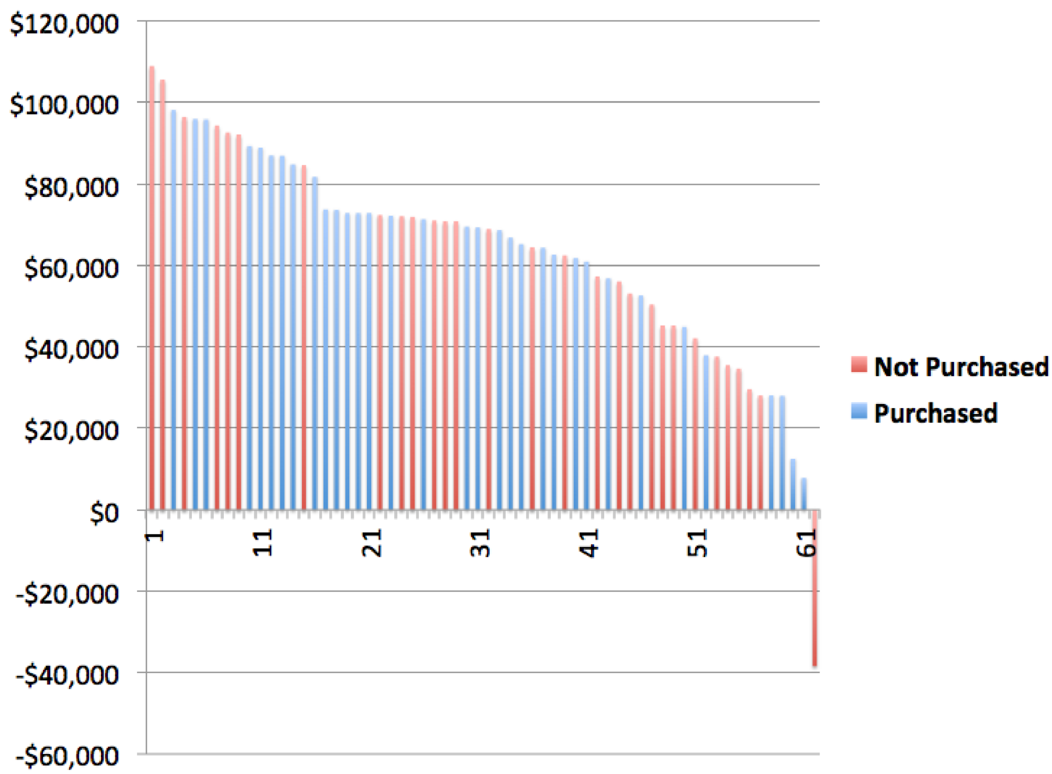


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.18: Final Cash Balances for 62 Players in #321 Insurance, Steady, Distributor [Sorted by Distributor Balance]



(a) Final Supply Chain Cash Balance



(b) Final Distributor Cash Balance

Figure 5.19: Final Cash Balances for 62 Players in #321 Insurance, Steady, Distributor (Policy Purchased vs. Not Purchased)

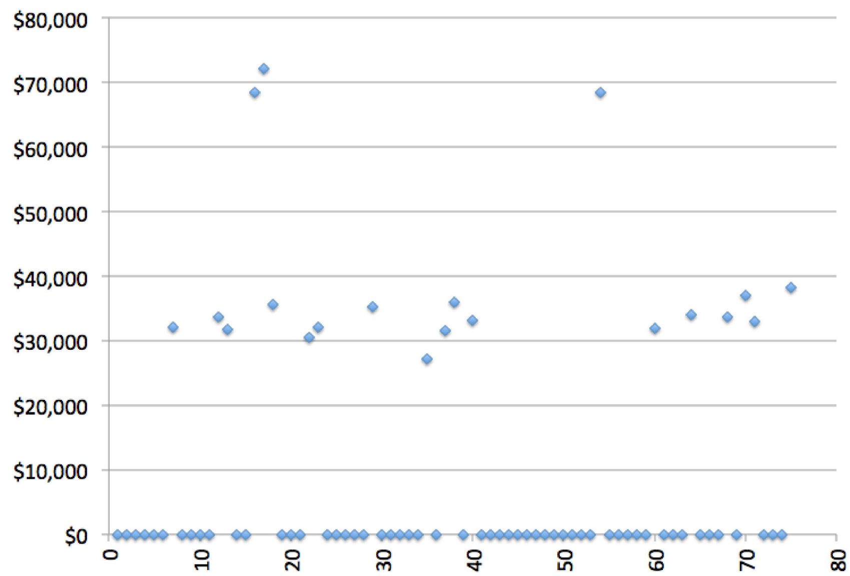


Figure 5.20: Total Claim Sizes in #321 Insurance, Steady, Distributor

Table 5.21: Frequencies of Claims in #321 Insurance, Steady, Distributor

# claims	0	1	2
# games	55	17	3

who played this version multiple times, the charts show only the best result. Thus, the charts have 62 data points. Figure 5.18 shows the same results, but the data points are sorted by the distributor balance.

Figure 5.19 also shows the sorted cash balances, but the data points are colored by the policy purchase decisions.

Figure 5.17 and Figure 5.18 show that there were a few cases where ruin occurred, i.e., the insurer's cash balance became below $0. In those cases, there were two claims in a single game. Ruin occurs because the insurer starts with no capital reserves, as do the other players. As explained earlier, ruin is acceptable in these experiments because our focus lies elsewhere. In the real world, VOs must start with sufficient capital for their planned portfolio size and take corrective action if the ruin probability approaches the VO's cap.

Figure 5.20 is a scatter plot showing the total claim sizes observed in version #321. The figure includes all of the 75 data points we got. Table 5.21 summaries the frequencies of

Table 5.22: Final Supply Chain Cash Balances in #321 Insurance, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	75	$191,445.33	$245,700.00	$264,205.18
	Best	63	$188,992.06	$245,700.00	$281,826.55
Excluded	All	74	$218,431.08	$245,700.00	$124,077.10
	Best	62	$221,161.29	$245,700.00	$120,263.69

Table 5.23: Final Distributor Cash Balances in #321 Insurance, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	75	$52,648.00	$66,900.00	$82,840.70
	Best	63	$52,958.73	$68,900.00	$89,259.79
Excluded	All	74	$61,660.81	$67,750.00	$27,943.26
	Best	62	$63,720.97	$69,100.00	$26,101.24

claims.

Table 5.22 summarizes statistical information for the final supply chain cash balances for this version, and Table 5.23 summarizes statistical information for the final distributor cash balances. Table 5.22 and Table 5.23 show the results for all players regardless of their decisions on policy purchases. We will discuss later the differences between those who purchased policies and those who didn't. As expected, the players had larger final supply chain cash balances in this version than those in the previous version because they could purchase policies to see the status of the others. In order to examine this phenomenon more closely, we subtracted the final supply chain balance for the previous See Only Yours version from that for this Insurance version for each player. Figure 5.21 shows the differences for the 62 players who played both versions. The data points are sorted by the amount of the difference and colored by the policy purchase decisions in version #321. The figure omits an outlier whose final supply chain balance in version #321 was -$1,805,500, resulting in a large negative difference value -$2,146,200.

We also performed a t-test and confirmed that the final supply chain balance in version #321 Insurance, Steady, Distributor is statistically significantly larger than that in version #221 See Only Yours, Steady, Distributor at the 5% significance level when the outlier is

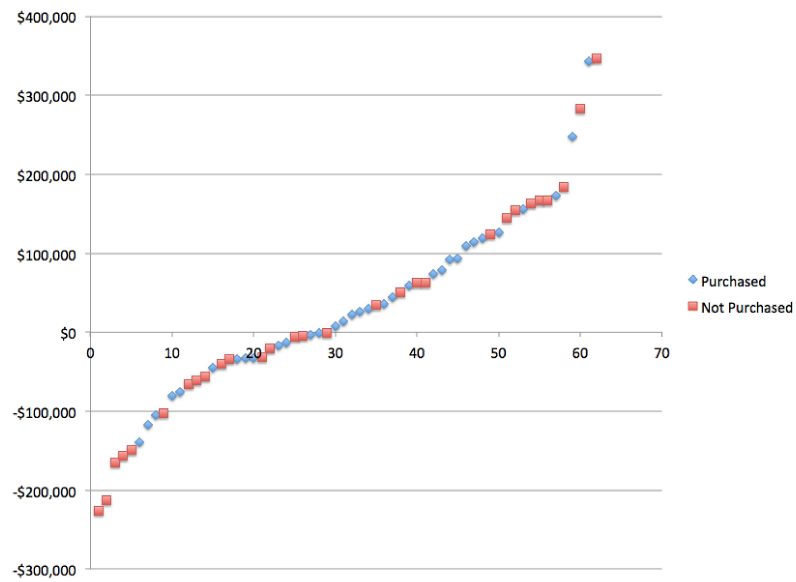


Figure 5.21: Difference of Final Supply Chain Balance Between Version #321 and #221

Table 5.24: Difference of Final Supply Chain Balance Between Version #321 and #221

Dataset	# Data Points	Average	Median	Standard Deviation	P-value
With Outliers	63	-$4,101.59	$13,800.00	$299,942.63	0.45696
Without Outliers	62	$30,448.39	$17,850.00	$122,492.63	0.027448

excluded. Table 5.24 summarizes statistical information about the difference.

Another thing we notice from the results of version #321 in Table 5.22 and those of version #221 in Table 5.16 is that variability (standard deviation) is reduced in version #321 if the poor-playing outlier player is excluded. This reduction occurs even though, as discussed later on, players who purchase policies tend to have significantly different final balances than players who do not. The reduction in variability might be because of the introduction of insured access, and might also be because players become more skilled over time. The results of the tournament with four computer players in Table 5.4 also show that the standard deviation of the results with information ("Actual" in the table) is smaller than that of the results without information ("Expected" in the table). This is interesting because variability indicates risks, and thus risk averse principals or organizations might naturally prefer to reduce variability by purchasing policies. Information is freely available

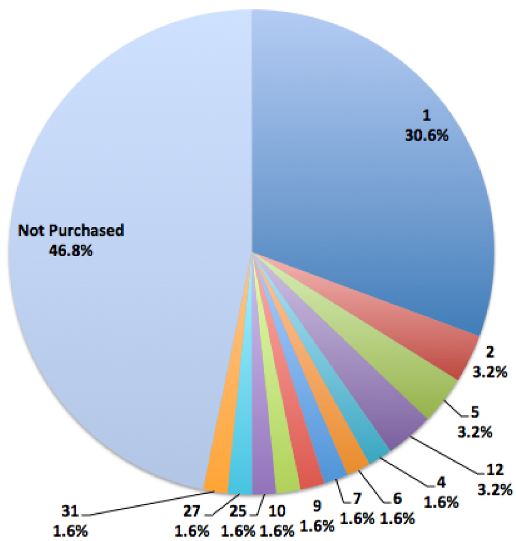


Figure 5.22: Week in Which Players Purchased Policies

in version #121, but the standard deviation of the results of that version in Table 5.12 is rather large. This may be because human players were not so familiar with the Beer Game with the Steady demand pattern when they played that version.

We are also interested in how many human players purchased policies in which week during games. Figure 5.22 illustrates the percentages of players who purchased policies in each week in the game version #321 Insurance, Steady, Distributor. The figure is for the 62 players excluding an outlier whose final supply chain balance in version #321 was -$1,805,500. For those who played this version multiple times, only data for the game they played the best is used. We can see that slightly more than half of the players purchased policies.

We now examine the differences between human players who purchased policies and those who didn't. Table 5.25 summarizes statistical information regarding the final supply chain cash balances for these two groups. We can see that players who purchased policies achieved better results than those who didn't. Players who purchased policies made about $65,000 more on average than those who didn't, when outliers are excluded. This is far higher than we would expect from the computer players. We also performed a t-test and confirmed that the final supply chain balance for the former is statistically significantly

Table 5.25: Final Supply Chain Cash Balances in #321 Insurance, Steady, Distributor (Policy Purchased vs. Not Purchased)

Dataset	# Data Points	Average	Median	Standard Deviation
Purchased	33	$253,854.55	$262,000.00	$101,868.93
Not Purchased with Outliers	30	$122,420.00	$212,750.00	$387,105.79
Not Purchased without Outliers	29	$188,900.00	$216,500.00	$133,715.09

Table 5.26: Final Distributor Cash Balances in #321 Insurance, Steady, Distributor (Policy Purchased vs. Not Purchased)

Dataset	# Data Points	Average	Median	Standard Deviation
Purchased	33	$65,915.15	$69,500.00	$22,636.92
Not Purchased with Outliers	30	$38,706.67	$63,400.00	$126,755.55
Not Purchased without Outliers	29	$61,224.14	$64,400.00	$29,773.15

larger than that for the latter at the 5% significance level. The P-value is 0.04025 if we include the outlier in the latter group and 0.01901 if we don't. Another thing we can see from the table is that standard deviation for those who purchased policies is smaller than that for those who didn't.

Table 5.26 summarizes statistical information regarding the final distributor cash balances for the two groups. We can see that players who purchased policies achieved better distributor balances than those who didn't. However, this difference is not statistically significant.

Perhaps the difference in final outcomes between the two groups (those who purchased policies and those who didn't) is largely attributable to skill, i.e., perhaps all the best players purchased. To examine this, we also analyze how the two groups performed in version #121 See All, Steady, Distributor and #221 See Only Yours, Steady, Distributor. Table 5.27 summarizes statistical information about the final supply chain cash balances for these two groups in the two versions. In the table, *#121 - #221* indicates the difference between the results of version #121 and those of version #221. If we look at the data more closely, we also notice that the purchasers played better than non-purchasers in the See All version

(#121), with approximately 10% better median and average final outcomes. Then with the See Only Yours version (#221), the medians for the two groups are almost the same. In other words, the purchasers are only better players when they can see the information that they purchase; without that information, they play no better than the others.

If we focus on the average final balance instead of the median in Table 5.27, purchasers do look better than non-purchasers for See Only Yours (#221). However, the averages are somewhat misleading because they are very affected by players' "wipeout" games, where one mistake by the distributor sends the supply chain spiraling toward a negative final balance. As shown in Figure 5.23, five of the seven wipeouts in See Only Yours were from non-purchasers. The typical (median) non-purchaser, however, played as well as the typical purchaser in See Only Yours. Wipeouts also contribute significantly to variability, and Table 5.27 shows that purchasers have much lower variability than non-purchasers.

The bottom pair of rows in Table 5.27 shows that on average, purchasers and non-purchasers' final balances dropped about the same amount when they could not see other players' status. Here again, however, the average is disproportionately affected by wipeouts. The median tells the story of the typical player: purchasers' balances dropped a third more than non-purchasers' balances when other players' statuses were no longer visible. This suggests that purchasers' ordering strategies were more dependent on information about other players than were non-purchasers', even before insured access was introduced.

Figure 5.23 is a scatter plot showing results of the two versions. The x-axis is for the results of version #121 and the y-axis is for those of version #221. Performance on the two versions is positively correlated; we computed a Pearson's correlation coefficient of 0.48 between the balances of players on the first version (1 to 63) and on the second version, and also computed a Kendall "/s Tau of 0.32, which is statistically significant with z-value of 3.70, between the rank order of players on the first version and on the second version As shown in Table 5.27, the non-purchasers are much more variable. We also see that the worst performers tend to be non-purchasers. Some non-purchasers did play extremely well in both games, compared to their peers. In fact, the players with the highest level of demonstrated skill, i.e., those who made around $300,000 and up in both games, are split almost equally

Table 5.27: Final Supply Chain Cash Balances in #121 and #221 (Policy Purchased vs. Not Purchased)

Dataset	Policy	# Data Points	Average	Median	Standard Deviation
#121 (See All)	Purchased	33	$356,472.73	$378,900.00	$63,995.67
	Not Purchased	30	$322,506.67	$355,400.00	$94,277.40
#221 (See Only Yours)	Purchased	33	$215,345.45	$223,900.00	$116,488.63
	Not Purchased	30	$173,393.33	$226,900.00	$146,324.40
#121 - #221	Purchased	33	$141,127.27	$140,300.00	$109,516.65
	Not Purchased	30	$149,113.33	$114,350.00	$127,618.51

between purchasers and non-purchasers. Even for the best players, however, final balances were much higher in See All than in See Only Yours.

Several factors affect an ultra-rational player's decision to purchase a policy, including their level of risk aversion, the value of the information to them, and the variability of potential outcomes of the purchase. We did not measure the human players' level of risk aversion in game play, or explain to them the variation of game results so far, so we cannot judge how those factors influenced their choices. However, we did tell the players how valuable it had been to see other players' statuses, based on the data points gathered so far, and that should have been sufficient to motivate a rational player to buy a policy for the premium we were offering. Confirming this statement, more than half of the players purchased them in version #321. On the other hand, however, 47% of the players didn't purchase policies.

We can think of several potential reasons why this happened in our experiments. First, players who did well without information in version #221 thought they could do well without purchasing policies. Players can tell where they rank relative to other players by their bonus, so the top players might have overlooked the fact that the median player with status information is as good as the best players without status information, as shown in Figure 5.23. A player who was about top 25% in both version #121 and #221 said in a comment field for version #321, "Interesting twist! Don't think the insurance was worth it though." This player probably overlooked the value of information. This player

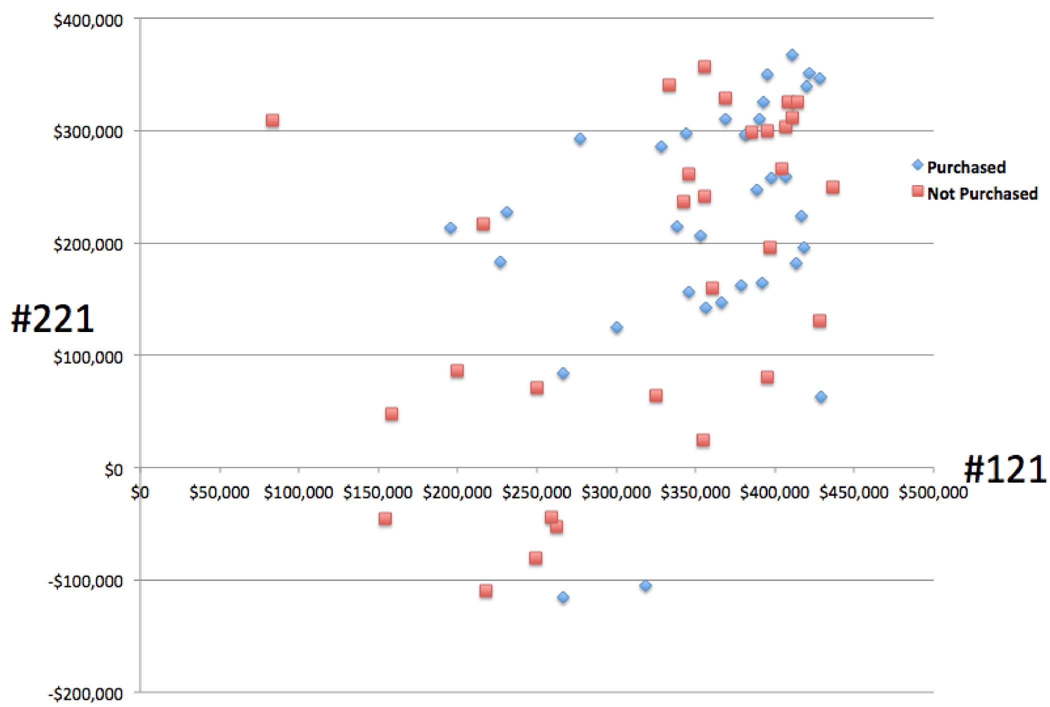


Figure 5.23: Final Supply Chain Cash Balances in #121 and #221 (Policy Purchased vs. Not Purchased)

ranked about top 50% in version #321, and they might have gotten better results if they had purchased a policy. Second, even though they knew policies were worth buying, they may have wanted to challenge themselves in harder games, by not purchasing access to the information that would make ordering easier. One player commented in version #321, "I took a chance and did not take the insurance and I paid for it. I got in a hole and worked my way out of it in the end, but that put me behind. I had a much better second half than first. I should have taken the insurance." This player challenged themselves in a harder game without a policy, but they regretted that decision. Third, their ordering strategy did not require them to consider the status of the others while playing games. The analysis of Table 5.27 lends support to that hypothesis. In fact, one player commented in version #221, "I found this version easier to play because I was just focusing on my numbers and not watching what everyone else was doing." This player didn't purchase a policy in version #321, which matches what they said. Fourth, if players buy a policy at the start of the game, their cash balance in Week 1 will be roughly -$14,000. Voluntarily going that far in the red is psychologically painful, and it will be many weeks before their cash balance is positive. Putting it another way, purchasers must delay gratification, which is always hard for humans. Even a player who purchased a policy commented in version #321, "I do think it was easier with being able to see the other players' supply and demand, but that insurance really sets you back at the beginning. The wholesaler in particular in my game had a hard time getting out of the red. I wonder if a weekly pay out for the insurance might be feasible." This comment indicates there is a psychological hurdle to purchase a policy at the beginning. No doubt other factors also played a role for some of the players who did not buy a policy, but still more than half of the players did purchase them. This confirms that many humans would be willing to pay the price required to become consumers of shared information under insured access.

These results validate our claim that the introduction of insured access does encourage information sharing and produces better outcomes for the VO (the supply chain in the experiments) as a whole, even when human decisions are involved.

Table 5.28 summarizes the bonus amounts we granted in this version. There was one

Table 5.28: Bonus Granted in #321 Insurance, Steady, Distributor

Range of Supply Chain Balance (B_{sc})	% Workers	# Workers	Bonus Granted
$B_{sc} \geq \$300,000$	Top 27%	17	$2.00
$\$300,000 > B_{sc} \geq \$235,000$	Top 27 - 60%	21	$1.50

more worker in the range of $\$300,000 > B_{sc} \geq \$235,000$, but that worker didn't submit their assignment and we could not grant them a bonus.

5.3.5 #321 Insurance, Steady, Distributor (Second Trial)

The previous round of #321 was our players' first experience with deciding whether to purchase insurance to gain access to shared information. We wanted to see how their decisions evolved over time, given detailed feedback on their performance on previous versions. To this end, when players logged in, we showed them a bar chart of each distributor's final balance from the previous trial of #321. The bars were in one of two colors, corresponding to whether that particular distributor had bought insurance or not. To make it easy for a player to see how she was ranked against the other players in the previous round of #321, we highlighted her bar. So that players could see how their rankings had evolved across the different versions, we showed them similar bar charts for the previous versions as well. We challenged them to play again and try to move up in the distributor rankings.

As explained in the previous section, for the supply chain final cash balance, the top echelon of players in the first round of #321 was divided fairly evenly between purchasers and non-purchasers. In particular, the top scorer was a non-purchaser, followed by three purchasers. However, the top players with respect to distributor final cash balances were non-purchasers. For round 2, we would expect players to notice at least these top few players' decisions not to purchase, as well as their own approximate ranking in round 1. Some players may have gone a step further and scrolled down to check whether their past rankings tended to be higher with or without insurance.

We allowed all 62 players who played the first trial of version #321 Insurance, Steady, Distributor to accept our HITs for this second trial. In this version, players can see the

Table 5.29: Final Supply Chain Cash Balances in #321 Insurance, Steady, Distributor (Second Trial)

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	70	$97,131.43	$224,900.00	$486,367.53
	Best	53	$120,120.75	$229,300.00	$519,340.81
Excluded	All	69	$147,134.78	$226,600.00	$249,869.46
	Best	52	$186,913.46	$230,200.00	$184,174.13

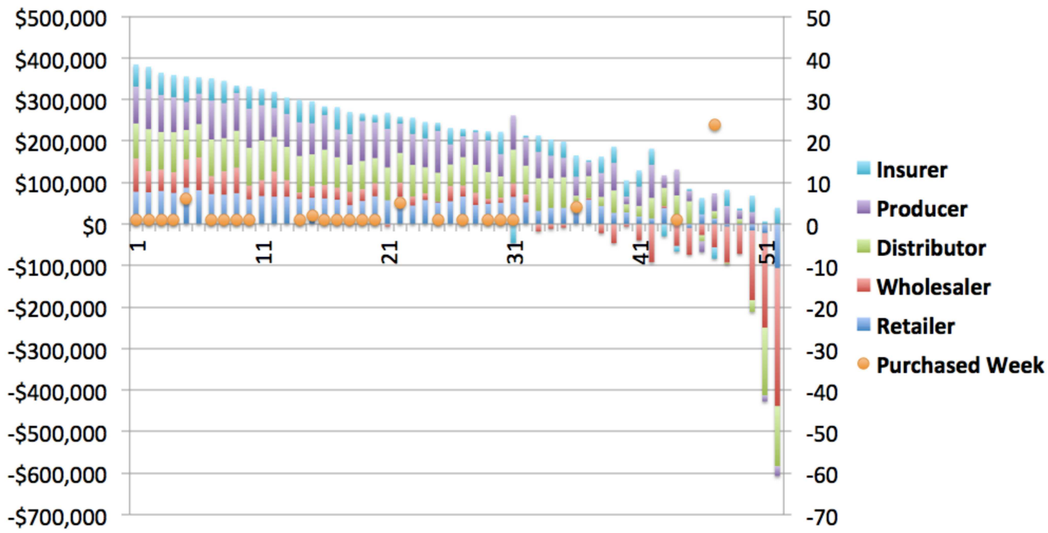
status of the others freely only if they purchase an insurance policy. The end-customer demand pattern is Steady, in which the mean demand remains constant over time. We use the Stock Management Structure (SMS) ordering strategy. We use the same γ and β values as those used in version #321 Insurance, Steady, Distributor, shown in Table 5.20. If they played the same version more than once, only the best score is shown. Scores below $0 are omitted.

Before their game begins, we show players the tutorial text in Appendix C.4, which is exactly the same as the one used in the first trial of version #321. The players' comments after they played the second trial of #321 are shown in Appendix D.5.

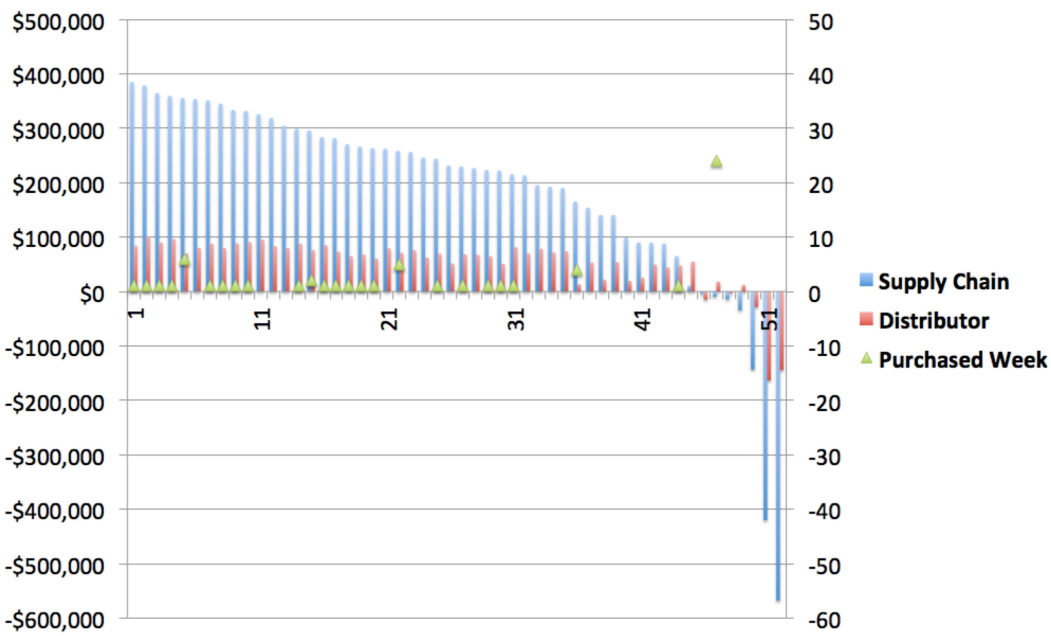
We collected 70 data points for this version. If we retain only the best result for each player, we collected 53 data points. Figure 5.17 shows final cash balances of the four positions and insurer in a stacked column chart and and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. The charts omit an outlier whose final supply chain cash balance was -$3,353,100. For players who played this second trial of version #321 multiple times, the charts show only the best result. Thus, the charts have 52 data points. Figure 5.25 shows the same results, but the data points are sorted by the distributor balance. We focus on the human players' own position in the following discussions.

Table 5.29 summarizes statistical information about the final supply chain cash balances for this version, and Table 5.30 summarizes statistical information about the final distributor cash balances.

We are interested in how many human players purchased policies in which week during

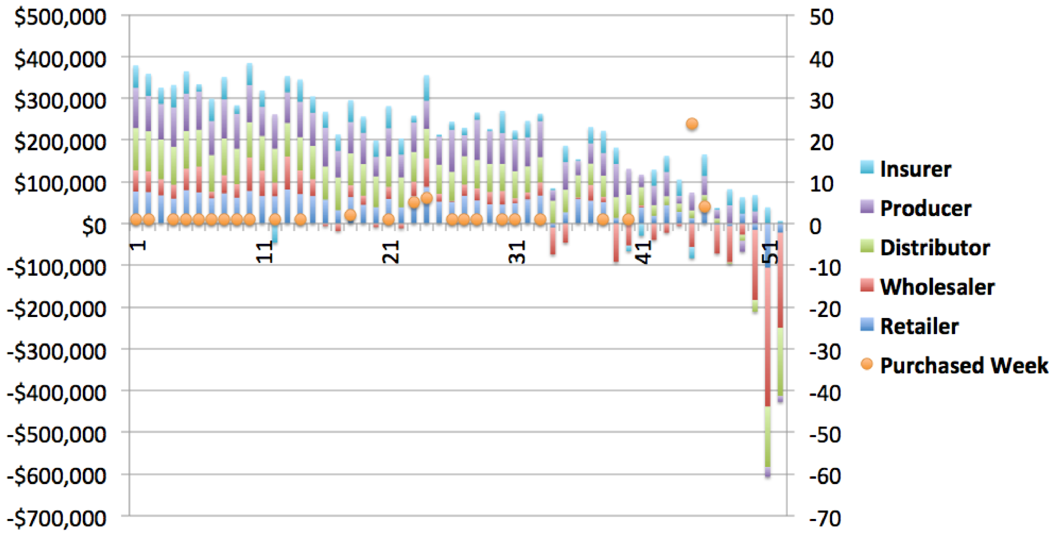


(a) Final Cash Balance of Each Position

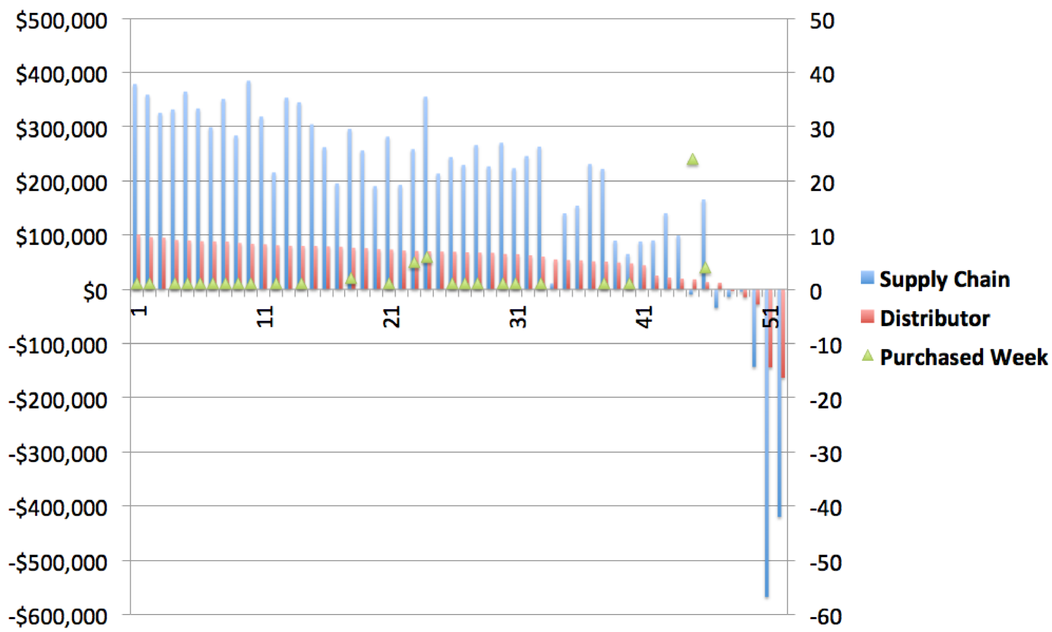


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.24: Final Cash Balances for 52 Players in #321 Insurance, Steady, Distributor (Second Trial) [Sorted by Supply Chain Balance]



(a) Final Cash Balance of Each Position



(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.25: Final Cash Balances for 52 Players in #321 Insurance, Steady, Distributor (Second Trial) [Sorted by Distributor Balance]

Table 5.30: Final Distributor Cash Balances in #321 Insurance, Steady, Distributor (Second Trial)

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	70	$19,484.29	$64,850.00	$174,738.02
	Best	53	$27,909.43	$68,200.00	$187,072.51
Excluded	All	69	$38,014.49	$65,000.00	$81,198.07
	Best	52	$52,659.62	$68,750.00	$50,788.18

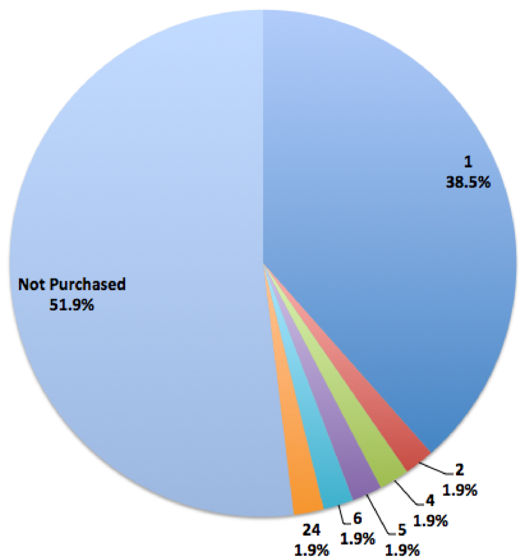


Figure 5.26: Week in Which Players Purchased Policies

games in this second trial of version #321. Figure 5.26 illustrates the percentages of players who purchased policies in each week in the second trial of the game version #321 Insurance, Steady, Distributor. The figure is for the 52 players excluding an outlier whose final supply chain cash balance was -$3,353,100. For those who played this version multiple times, only data for the game they played the best is used. We see that slightly less than half (48.1%) of the players purchased policies this time, while slightly more than half did in the first trial. One possible reason for the slight decline is that players observed that the players with highest distributor balances in the first trial did not purchase policies, and this influenced their decision. However, the best players in the second trial did purchase policies, as shown in Figure 5.25, where nine out of the top ten players are purchasers.

We now examine the differences between the first and second trials of this version by analyzing the outcomes of those who played version #321 in both trials. We divide human players into the following four groups: (1) players who were non-purchasers in both trials, (2) players who purchased policies in both trials, (3) players who didn't purchase polices in the first trial but did purchase in the second, and (4) players who purchased policies in the first trial but didn't in the second. Table 5.31 and Table 5.32 summarize how the human players' decisions on whether to purchase policies or not affected their final distributor balances. In the tables, NP means "Not Purchased" and P means "Purchased".

Table 5.31 shows the number and percentage of each group, the number and percentage of players in the corresponding group whose distributor balances decreased in the second trial, and the number and percentage of players in the corresponding group whose distributor balances increased in the second trial. The majority of players (62.27%) made the same purchasing decisions in the two trials; these are divided almost evenly between non-purchasers and purchasers (30% and 32%, respectively). The remaining 38% of players reversed their decisions in the two trials.

Table 5.32 shows the number and percentage of each group, and the average, median, and standard deviation of the differences of the final distributor balances between the two trials in the corresponding group.

Overall, the players generally tended to do worse in the second trial, although this trend is not statistically significant. One contributing factor is that we ran the second trial a few months after the first trial, and the human players' skill level might have declined through lack of practice. (In subsequent versions played soon afterwards, the scores of the weaker players climbed back up into positive territory.) However, Table 5.31 and Table 5.32 show that one subgroup bucked this trend: players who didn't purchase policies in the first trial but did in the second tended to do better in the second trial, in terms of the percentage of players who did better in the second trial and the median of the difference in the final distributor balance between the two trials. This group went against the model set forth by the best players in the first trial, suggesting that they knew that what worked for the best players in the first trial had not worked so well for them.

Table 5.31: Policy Purchase Decisions and Effects to Final Distributor Cash Balances in #321 Insurance, Steady, Distributor

Decisions Changed?	Decision Details	#	%	# Decreased	% Decreased	# Increased	% Increased
Same in Both Trials	NP → NP	16	30.19%	12	75.00%	4	25.00%
	P → P	17	32.08%	9	52.94%	8	47.06%
	Total	33	62.27%	21	63.64%	12	36.36%
Different	NP → P	8	15.09%	3	37.50%	5	62.50%
	P → NP	12	22.64%	8	66.67%	4	33.33%
	Total	20	37.73%	11	55.00%	9	45.00%
Total		53	100.00%	32	60.38%	21	39.62%

Table 5.32: Policy Purchase Decisions and Differences of Final Distributor Cash Balances in #321 Insurance, Steady, Distributor

Decision	Decisions	#	%	Average	Median	Standard Deviation
Same in Both Trials	NP → NP	16	30.19%	-$27,293.75	-$18,950.00	$52,444.67
	P → P	17	32.08%	$11.76	-$700.00	$23,712.18
	Total	33	62.27%	-$13,227.27	-$3,100.00	$41,981.55
Different	NP → P	8	15.09%	-$3,875.00	$2,000.00	$19,462.62
	P → NP	12	22.64%	-$127,583.33	-$9,400.00	$382,479.89
	Total	20	37.73%	-$78,100.00	-$2,750.00	$297,826.26
Total		53	100.00%	-$37,707.55	-$3,100.00	$185,747.70

Table 5.33: Bonus Granted in #321 Insurance, Steady, Distributor (Second Trial)

Range of Distributor Balance (B_d)	% Workers	# Workers	Bonus Granted
$B_d \geq \$80,000$	Top 25%	13	$2.00
$\$80,000 > B_d \geq \$60,000$	Top 25 - 62%	20	$1.50

Table 5.33 summarizes the bonus amounts we granted in this version. We also granted $1 bonus to workers who were not qualified for the amounts shown in the table but played the first trial of version #321 Insurance, Steady, Distributor, as part of their invitation to play again.

5.3.6 #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

In this version, the computer players always request to see their business partners' status. The difference between this version #421 and version #321 is that human players now have options when computer players ask them to release their status information. If they opt out of the insurance program, they get to keep the cash that their partners would have paid for a policy, but they will not be reimbursed for any (simulated) losses they incur due to sharing. If they opt into the program, then they will be reimbursed for their losses. Either way, the computer players will be allowed to see their partners' status. At the end of the game, the insurer's final balance will be split among the players who opted in, as a form of profit sharing.

In this version, the human players can request to see the status of their partners, if that information will be helpful to them. If they make this request, the computer players require them to purchase an insurance policy, just as in version #321.

We allowed all players who played the first trial of version #321 Insurance, Steady, Distributor to accept our HITs for this version. The end-customer demand pattern is Steady, in which the mean demand remains constant over time. We use the Stock Management Structure (SMS) ordering strategy. We use the same γ and β values as those used in version #321 Insurance, Steady, Distributor, shown in Table 5.20.

Table 5.34: Final Supply Chain Cash Balances in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	58	$204,582.76	$261,150.00	$226,432.92
	Best	49	$195,316.33	$252,800.00	$244,637.05
Excluded	All	57	$231,861.40	$264,600.00	$90,870.89
	Best	48	$227,516.67	$258,700.00	$96,090.67

When human players log in to the system, they first see bar charts showing the distributor final cash balances for all human players and all game versions. Each player's own scores are highlighted for easy reference, and we ask them to see if they can move up in the rankings for this new version. If they played the same version more than once, only the best score is shown. Scores below $0 are omitted.

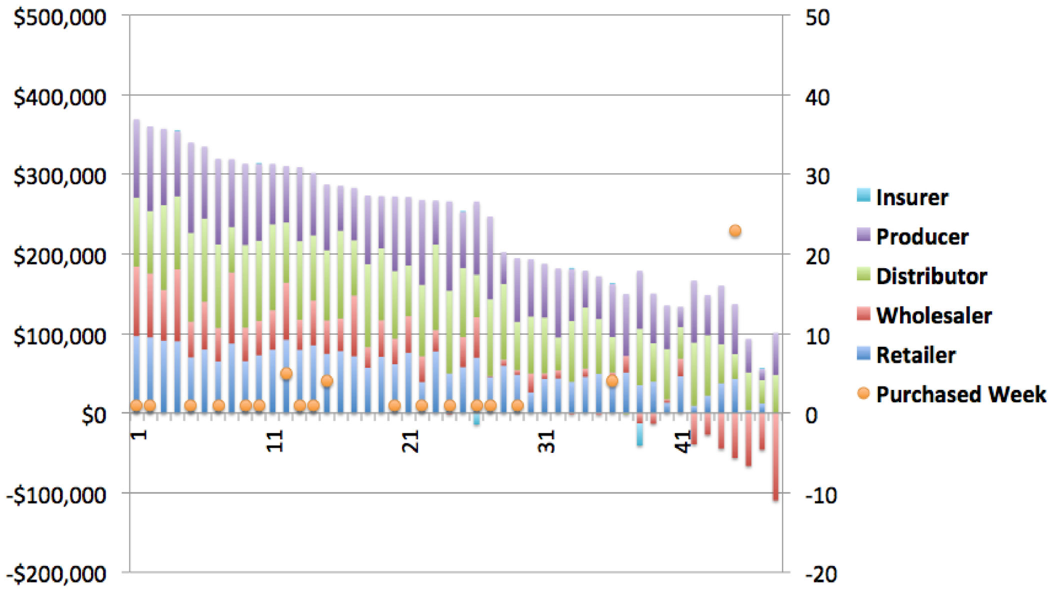
Before the game begins, we show players the tutorial text in Appendix C.5. Players' comments for this version are shown in Appendix D.6.

We collected 58 data points for this version. If we retain only the best result for each player, we collected 49 data points. Figure 5.27 shows final cash balances of the four positions and insurer in a stacked column chart, and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. The charts omit an outlier whose final supply chain cash balance was -$1,350,300. For players who played this version multiple times, the charts show only the best result. Thus, the charts have 48 data points. Figure 5.28 shows the same results, but the data points are sorted by the distributor balance. We focus on the human players' own position in the following discussions.

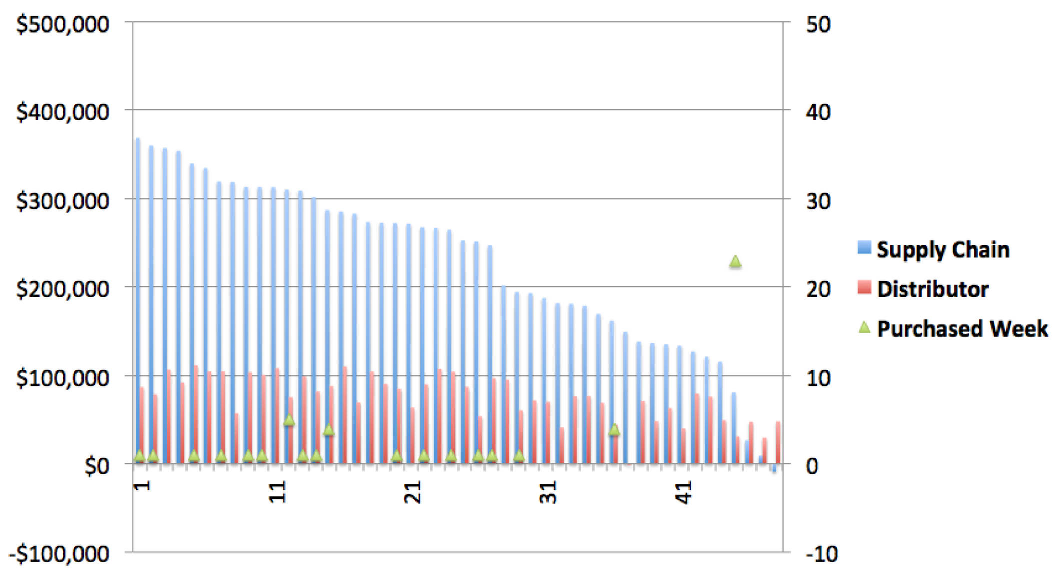
Figure 5.29 also shows the sorted balances, but the data points are colored by the human players' decisions when they are asked to release their status information.

Table 5.34 summarizes statistical information about the final supply chain cash balances for this version, and Table 5.35 summarizes statistical information about the final distributor cash balances.

We now examine the differences between the human players who kept money when they

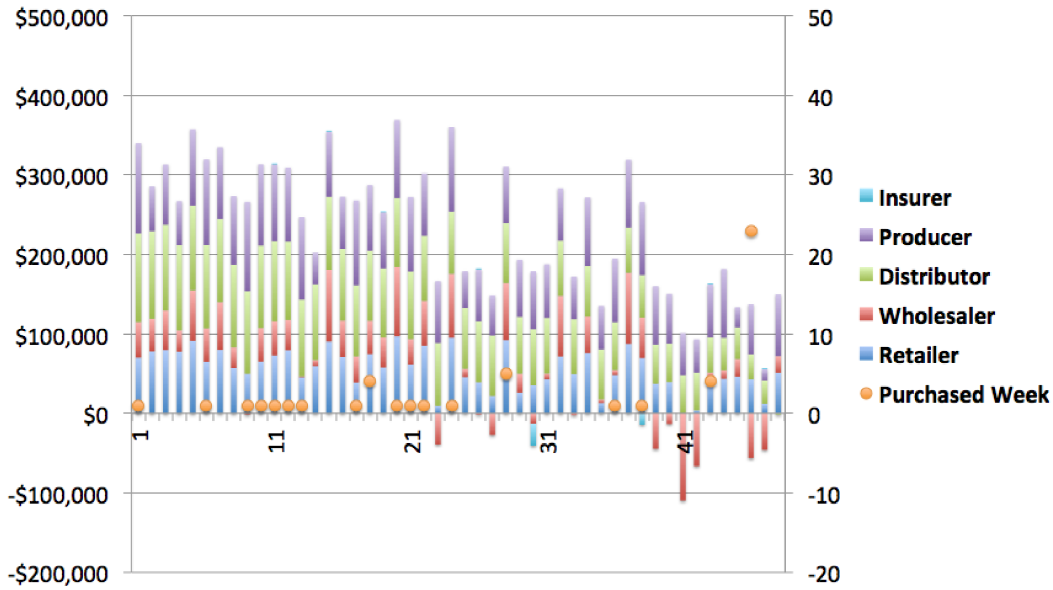


(a) Final Cash Balance of Each Position

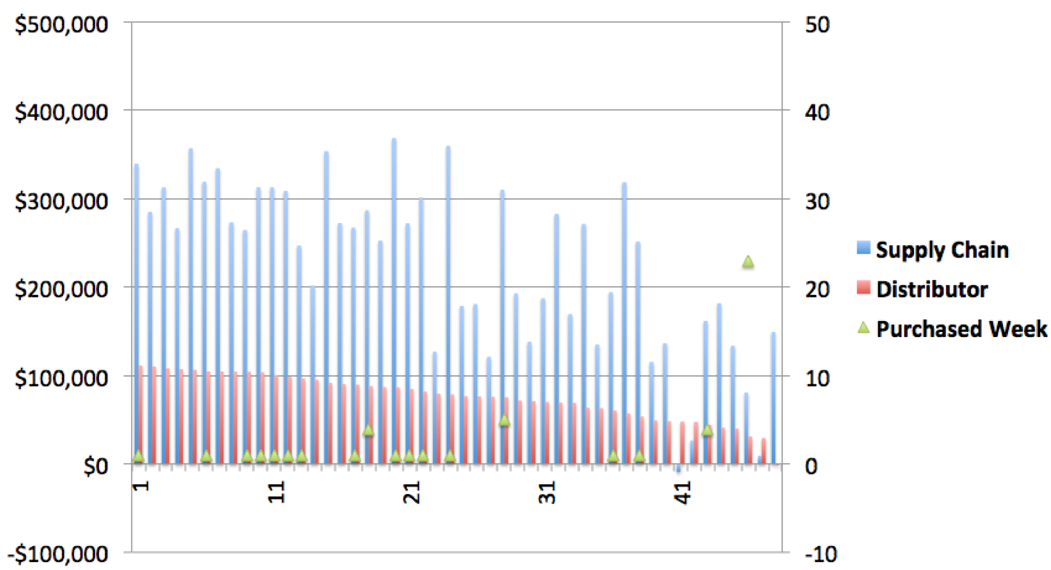


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.27: Final Cash Balances for 48 Players in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor [Sorted by Supply Chain Balance]

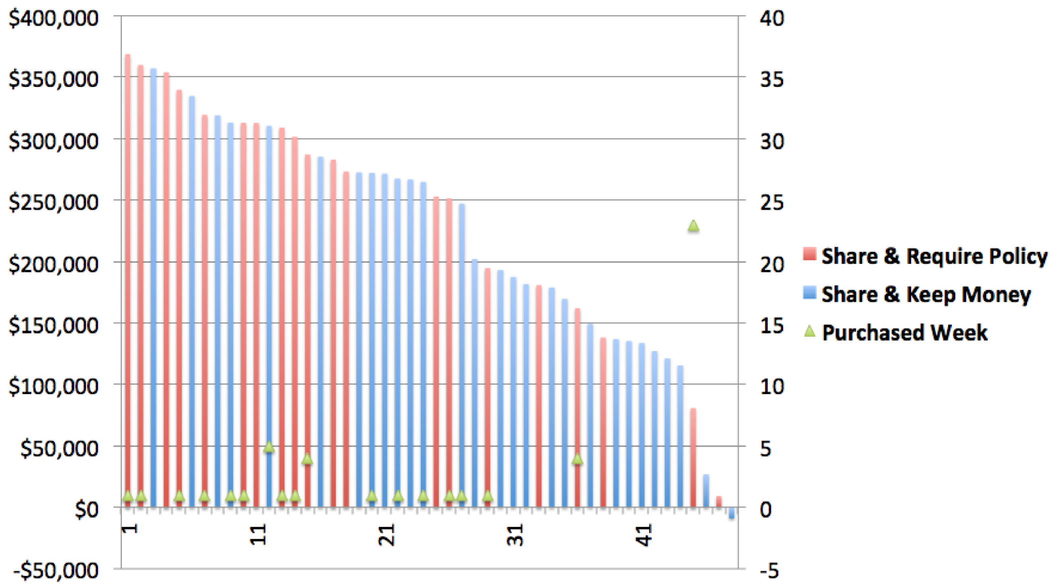


(a) Final Cash Balance of Each Position

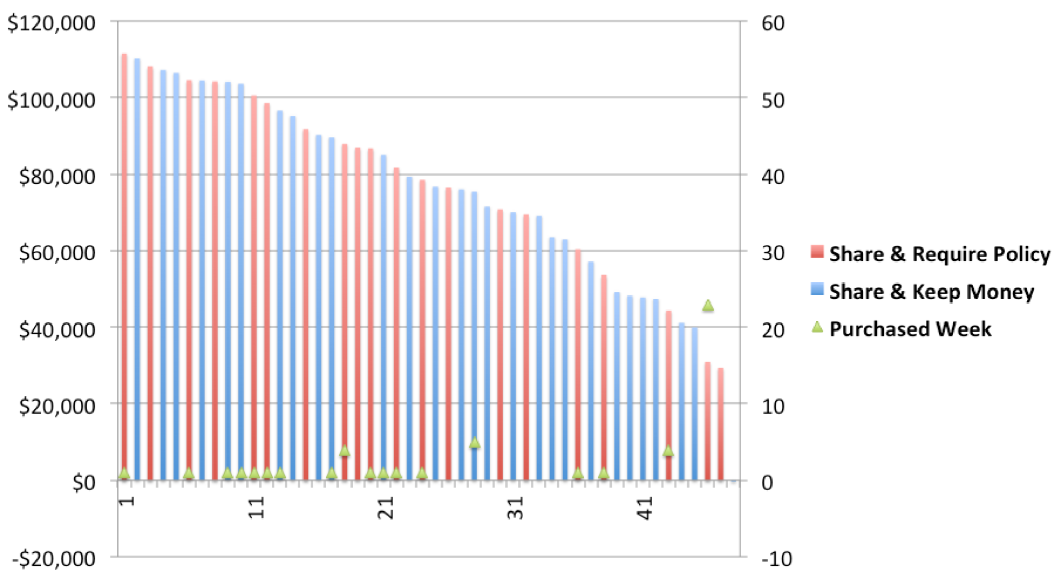


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.28: Final Cash Balances for 48 Players in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor [Sorted by Distributor Balance]



(a) Final Supply Chain Cash Balance



(b) Final Distributor Cash Balance

Figure 5.29: Final Cash Balances for 48 Players in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor (Share & Keep Money vs. Share & Require Policy)

Table 5.35: Final Distributor Cash Balances in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	58	$67,288.76	$81,002.00	$84,019.46
	Best	49	$63,427.18	$76,702.00	$90,972.52
Excluded	All	57	$77,870.42	$81,702.00	$23,979.27
	Best	48	$75,912.46	$77,602.00	$25,521.51

Table 5.36: Final Distributor Cash Balances in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor (Share & Keep Money vs. Share & Require Policy)

Dataset	# Data Points	Average	Median	Standard Deviation
Share & Keep Money	28	$73,837.71	$75,702.00	$26,378.46
Share & Require Policy with Outliers	21	$49,546.48	$81,702.00	$136,268.12
Share & Require Policy without Outliers	20	$78,817.10	$84,202.00	$24,641.46

released their status information and those who made computer players purchase policies. Table 5.36 summarizes statistical information regarding the final distributor cash balances for these two groups. About 57% of the players decided to keep the money themselves, instead of asking computer players to purchase policies when the humans released their status information. This illustrates that humans without specialized financial, actuarial, or organizational expertise are likely to be more interested in increasing their own capital as soon as possible, instead of taking advantage of insured access and its uncertain future profit sharing scheme. Interestingly, players who made computer players purchase policies achieved $8.5K better median final balances than those who kept the money themselves, when the outlier is excluded. However, this difference is not statistically significant at this sample size.

Figure 5.30 shows frequencies of claims the players encountered in the best games of the first trial of version #321 Insurance, Steady, Distributor. Figure 5.30a is for those who chose to keep the money in version #421 and Figure 5.30a is for those who chose to make the computer players purchase policies. We can see that there is a slightly larger percentage of players who didn't encounter any claims in version #321 in the group who chose to keep

Table 5.37: Bonus Granted in #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

Range of Distributor Balance (B_d)	% Workers	# Workers	Bonus Granted
$B_d \geq \$100,000$	Top 22%	11	$2.00
$\$100,000 > B_d \geq \$70,000$	Top 22 - 58%	20	$1.50

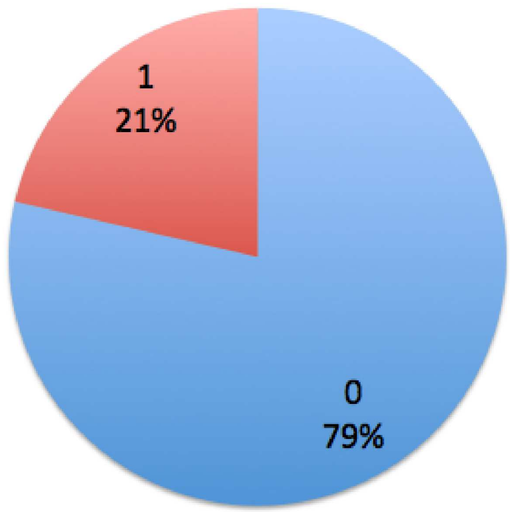
money in version #421. The figure also shows that all players who encountered more than one claim in version #321 decided to make computer players purchase policies in version #421. These observations indicate that people tend to opt for insured access if they have recent prior experience of bad events, to avoid potential future large monetary damages. When people experience unlikely events, it alters their subjective perception of the future probability of similar events, and may trigger their loss aversion instinct.

Table 5.37 summarizes the bonus amounts we granted in this version. We also granted $1 bonus to workers who were not qualified for the amounts shown in the table but played the first trial of version #321 Insurance, Steady, Distributor, to invite them to play the next version.

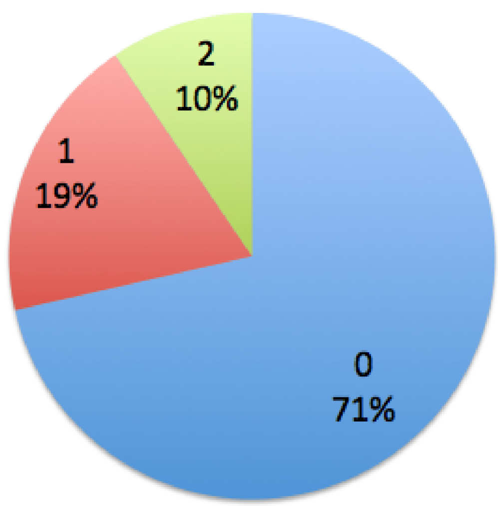
5.3.7 #521 Deny / Share & Require Policy, Steady, Distributor

In this game version #521, we examined the question of how willing players are to share their information with others, given a relatively painless and simple procedure for filing claims and recovering damages attributable to sharing. In this version, the human player has the option to grant (with insured access) or deny the computers players' request to share status information. From information we shared with them in earlier versions, the human players knew that computer players achieve higher final balances when they can see other players' status. Thus an altruistic team player might agree to release information for that reason. On the other hand, with no information release, there is no chance of damages due to sharing.

We allowed all players who played the first trial of version #321 Insurance, Steady, Distributor to accept our HITs for this version. Players can see the status of the others only if they purchase an insurance policy, just as in version #321. The end-customer demand



(a) Share & Keep Money in #421



(b) Share & Require Policy in #421

Figure 5.30: Frequencies of Claims in #321 Insurance, Steady, Distributor

Table 5.38: Final Supply Chain Cash Balances in #521 Deny / Share & Require Policy, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	65	$125,869.23	$250,200.00	$694,997.90
	Best	51	$103,235.29	$267,500.00	$783,790.59
Excluded	All	63	$235,200.00	$250,300.00	$109,013.12
	Best	49	$242,879.59	$276,300.00	$116,195.80

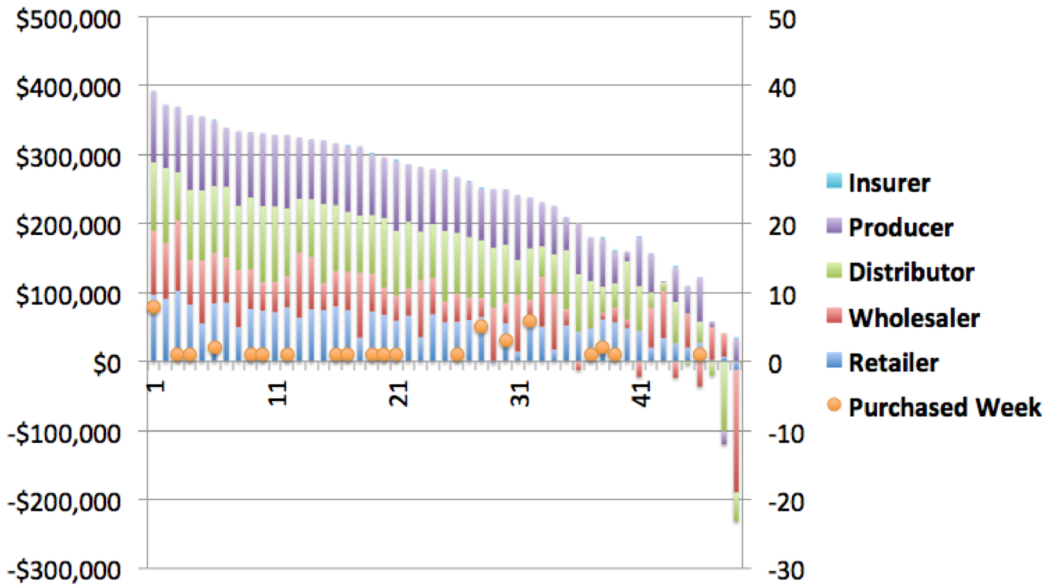
pattern is Steady, in which the mean demand remains constant over time. We use the Stock Management Structure (SMS) ordering strategy. We use the same γ and β values as those used in version #321 Insurance, Steady, Distributor shown in Table 5.20.

When human players log in to the system, they first see bar charts showing the distributor final cash balances for all players and all game versions. The dark bars are used to highlight their own scores. If they played the same version more than once, only the best score is shown. Scores below $0 are omitted. Before the game begins, we show players the tutorial text in Appendix C.6. Players' comments for this version are shown in Appendix D.7.

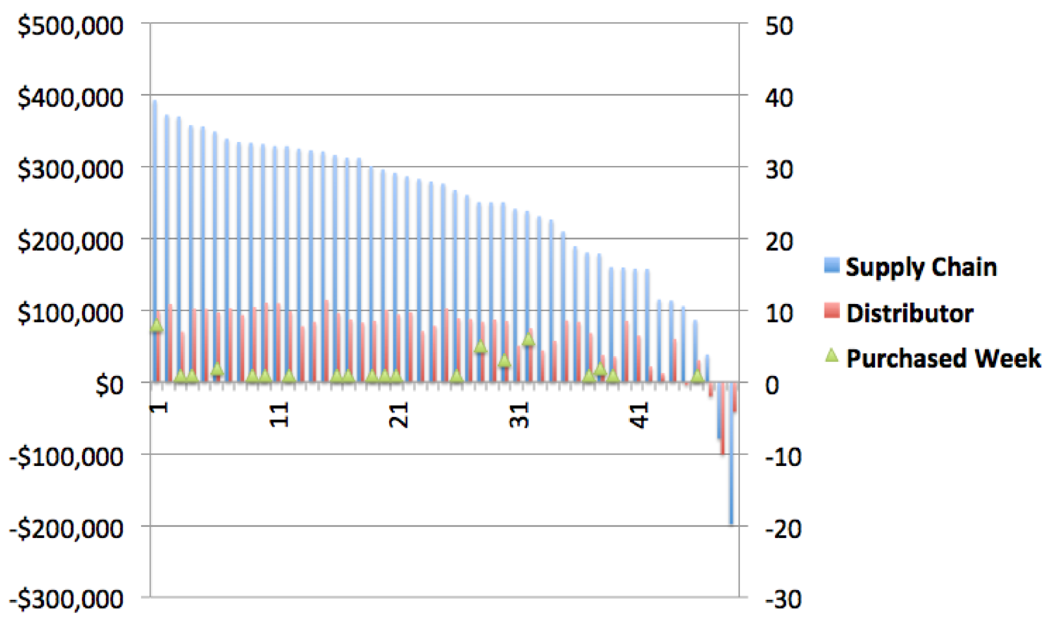
We collected 65 data points for this version. If we retain only the best result for each player, we collected 51 data points. Figure 5.31 shows final cash balances of the four positions and insurer in a stacked column chart, and those of the supply chain and distributor in a clustered column chart. The data points are sorted by the supply chain balance. The charts omit two outliers whose final supply chain cash balances were -$1,629,500 and -$5,006,600. For players who played this version multiple times, the charts show only the best result. Thus, the charts have 49 data points. Figure 5.32 shows the same results, but the data points are sorted by the distributor balance. We focus on the human players' own position in the following discussions.

Figure 5.33 also shows the sorted balances, but the data points are colored by the human players' decisions when they are asked to release their status information.

Table 5.38 summarizes statistical information about the final supply chain cash balances for this version, and Table 5.39 summarizes statistical information about the final distributor cash balances.

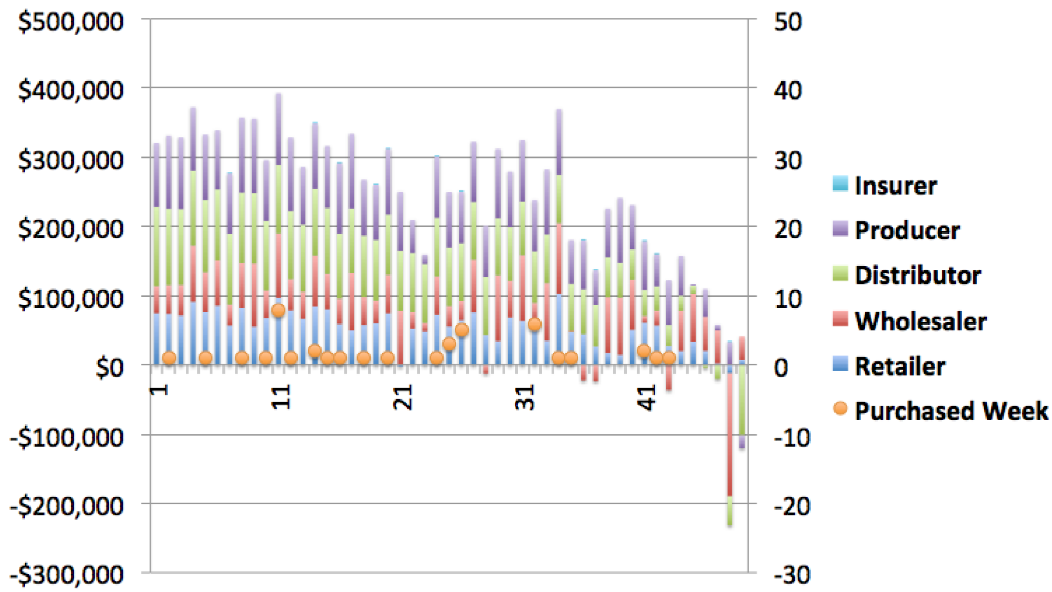


(a) Final Cash Balance of Each Position

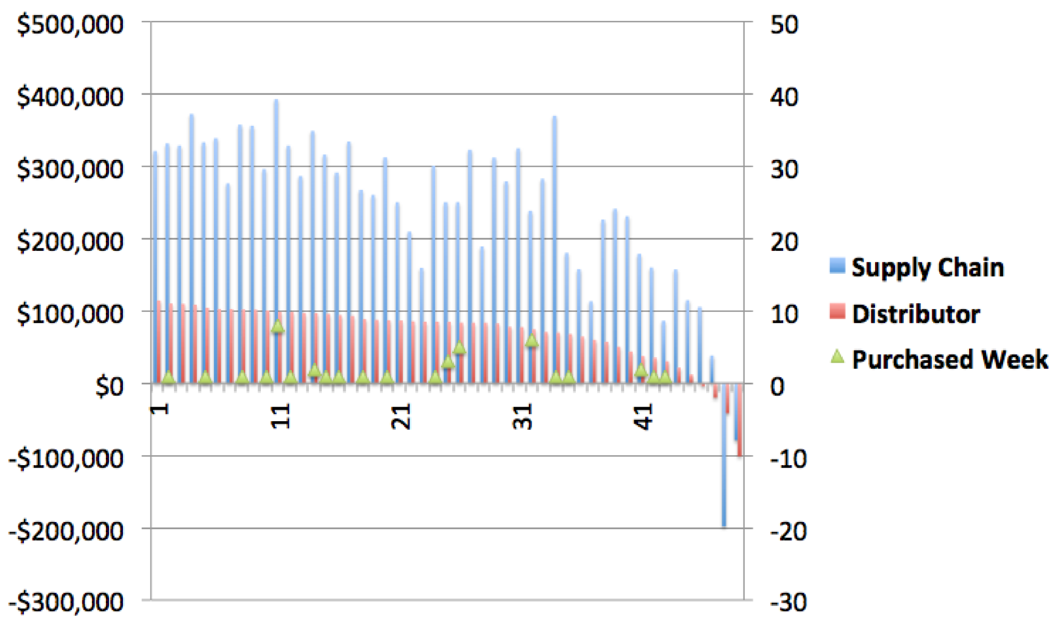


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.31: Final Cash Balances for 49 Players in #521 Deny / Share & Require Policy, Steady, Distributor [Sorted by Supply Chain Balance]

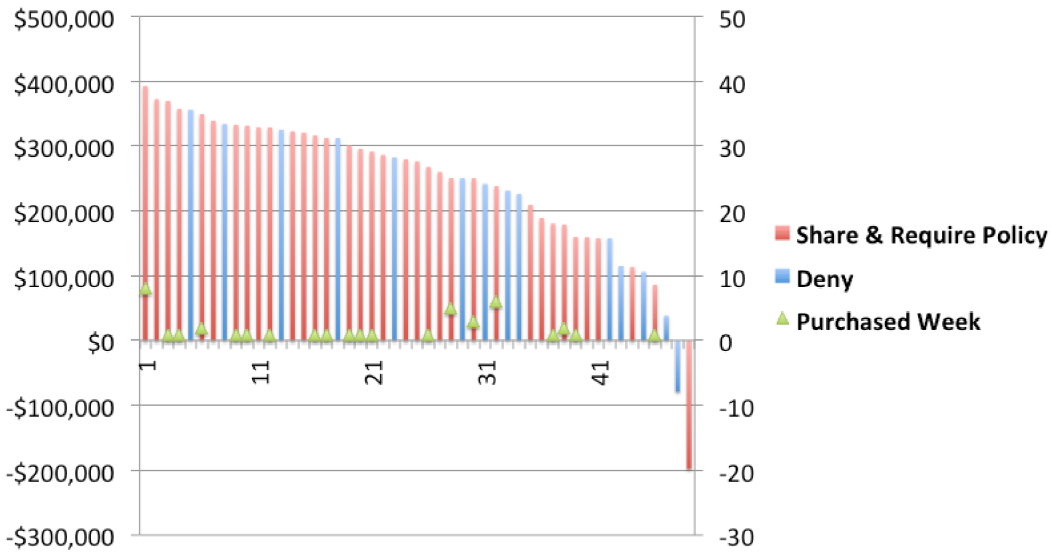


(a) Final Cash Balance of Each Position

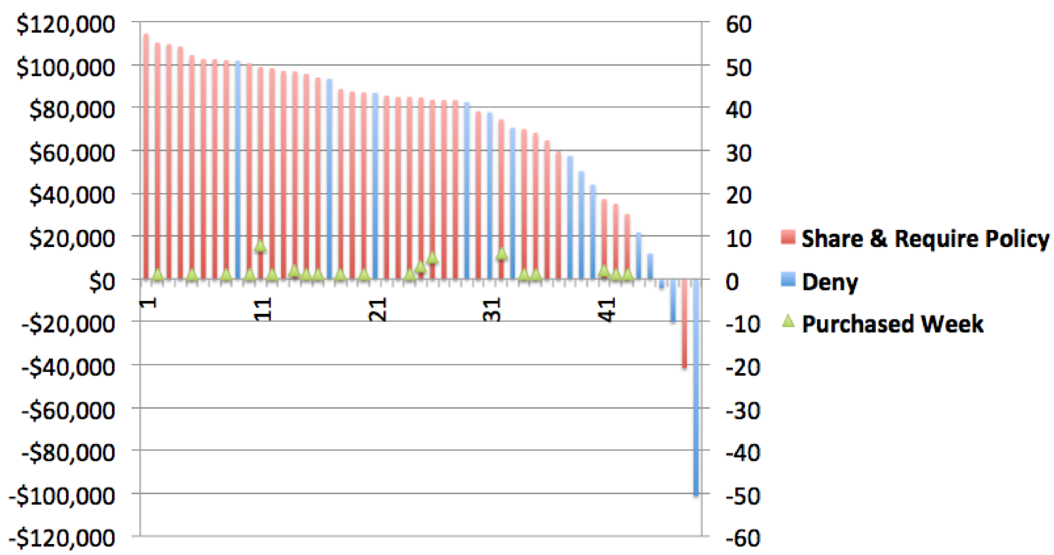


(b) Final Cash Balance for the Human Player and the Entire Supply Chain

Figure 5.32: Final Cash Balances for 49 Players in #521 Deny / Share & Require Policy, Steady, Distributor [Sorted by Distributor Balance]



(a) Final Supply Chain Cash Balance



(b) Final Distributor Cash Balance

Figure 5.33: Final Cash Balances for 49 Players in #521 Deny / Share & Require Policy, Steady, Distributor (Deny vs. Share & Require Policy)

Table 5.39: Final Distributor Cash Balances in #521 Deny / Share & Require Policy, Steady, Distributor

Outliers	Dataset	# Data Points	Average	Median	Standard Deviation
Included	All	65	$1,577.06	$78,102.00	$436,327.48
	Best	51	-$11,837.22	$83,509.00	$492,511.26
Excluded	All	63	$65,737.68	$79,200.00	$41,090.28
	Best	49	$70,107.49	$84,650.00	$42,578.12

Table 5.40: Final Distributor Cash Balances in #521 Deny / Share & Require Policy, Steady, Distributor (Deny vs. Share & Require Policy)

Dataset	# Data Points	Average	Median	Standard Deviation
Deny with Outliers	15	-$186,600.00	$50,200.00	882,447.28
Deny without Outliers	14	$40,828.57	$53,750.00	$55,467.46
Share & Require Policy with Outliers	36	$60,980.61	$86,175.00	$128,437.20
Share & Require Policy without Outliers	35	$81,819.06	$86,948.00	$29,813.39

We now examine the differences between the minority of human players who denied information access requests and those who made computer players purchase policies and released their status information. Table 5.40 summarizes statistical information regarding the final distributor cash balances for these two groups. About 29% of the players decided to deny information access requests instead of asking computer players to purchase policies when they released their status information. This illustrates that most humans are willing to provide information for insured access when it is relatively painless for them, rather than just denying requests to share their information. We can also see that players who made computer players purchase policies achieved much better results than those who denied access requests, when outliers are excluded. Players who made computer players purchase policies made about twice as much as those who denied access requests on average, when outliers are excluded. We also performed a t-test and confirmed that the final distributor balance for the former group is statistically significantly larger than that for the latter at the 1% significance level, when outliers are excluded. The P-value is 0.00929 if we exclude the outliers in both groups.

Table 5.41 summarizes the bonus amounts we granted in this version.

Table 5.41: Bonus Granted in #521 Deny / Share & Require Policy, Steady, Distributor

Range of Distributor Balance (B_d)	% Workers	# Workers	Bonus Granted
$B_d \geq \$100,000$	Top 20%	10	$2.00
$\$100,000 > B_d \geq \$80,000$	Top 20 - 57%	19	$1.50

5.4 Summary and Discussion

The crowdsourcing experiments in this chapter examined the questions mentioned in the beginning of this chapter, in the context of the Beer Game. We calculated the value of the information that could be shared in the beer game by running tournaments with four automated players. After observing games played by humans, we also calculated the value of the information to the average human player, which was very close to the value calculated from the tournament with automated players. We set up a version of the Beer Game with insured access, and calculated premiums so that a rational player was likely to want to purchase a policy. We found that about half of the players purchased policies. Some of the non-purchasers used game strategies under which it would not have been helpful to see their partners' status information. Taken all together, this indicates that the majority of people behaved in line with the predictions of the models in Chapter 4, while a substantial minority preferred not to pay the price to access shared information, even though it would have been useful to them.

Would a real-world VO be willing to adopt insured access? Each VO participant has a unique perspective for answering that question. First let us consider the challenges in setting up insured access.

Consumers: The crowdsourcing experiments in this chapter confirm the common-sense notion that sometimes people are willing to pay for useful information. Consumers' objections to insured access are likely to focus on perceived fairness, such as high premium levels and the details of bonus-malus schemes.

Producers: Producers are the least likely to be pleased by any VO information-sharing scheme, but profit-sharing and insurance can lessen the pain, at least in theory. To be adopted successfully from the producers' point of view, an implementation of insured access would require simple administrative procedures, straightforward accountability for and recognition of harm in the rare cases where it does occur, appropriate compensation levels for harm, and attractive profit-sharing. With these conditions in place, three-quarters of all information producers in our experiments agreed to share sensitive information about their business with insured access. Further experiments confirmed that the VO does need to enforce the use of insurance, as in spite of profit-sharing, information producers may tend to prefer to pocket the cash that could otherwise be used to pay for a policy to protect them.

VO member organizations: VO member organizations who will primarily be consumers of shared information may be enthusiastic about insured access, until they realize how much startup capital they will need to contribute to the insurer, while producer-oriented members are more likely to drag their feet. Differences in risk aversion between different members may be a further obstacle, as members must settle on a VO-wide level of risk aversion to be adopted by the insurer, yet more risk-averse members by definition require higher claim payments for a given degree of damage in order to be as satisfied as less risk-averse members. VO members may also be reluctant to share their sensitive historical information about harm with the insurer. VO members will also be concerned with administrative procedures, accountability, and the choice of information to be shared.

An insured access implementation that is working well for a VO will face its next major challenge when inevitably, at some point, a producer experiences significant harm. Will the producer's VO member be satisfied with the claim payment, or want to disband the information-sharing scheme? The biggest challenge will come when a long run of bad luck, or inaccurate actuarial models, push the insurer toward ruin. At that point, the VO members will have to decide whether to abandon the scheme, inject more capital, or modify the underwriting rules until better times come.

Chapter 6

Conclusions

6.1 Summary of Contributions

In this thesis, we proposed approaches to ad-hoc information sharing for virtual organizations that consider benefits while bounding risk.

In the first part of the thesis, we presented a decision framework for a VO to select an optimal portfolio of risky data accesses that will maximize the benefit to the VO subject to a given organizational risk budget. Our framework addresses two core problems that have been under-explored by prior work that targets financial investment portfolios: accommodating the VO's attitude towards risk as quantified by a risk aversion index function, and assessing risk during portfolio construction from more global perspective that considers the sub-additivity and super-additivity of correlated risks that may occur across multiple access requests. We defined risk in two ways: the first is as the variance of the benefit of proposed accesses and the second is as the tail distribution of the net benefit to the VO. We also introduced the notion of risk-invariant VOs, which is used to call a decision maker whose utility function is modeled using an exponential utility function. We showed that when the variance notion of risk is used and the benefits of transactions are not correlated, then risk-neutral and risk-invariant VOs can optimize their portfolios by a hybrid-greedy algorithm Otherwise, the case of accesses with correlated benefits is generally harder to solve than independent accesses. We proposed a simulated annealing solution approach, and analyzed the structure of the problem to devise effective search strategies. Our simulated annealing algorithm runs in two phases. Phase 1 aims to find an initial portfolio that satisfies the problem constraint and that has reasonably good performance. This initial solution is then used to seed the Phase 2 search that progressively improves the quality of solution until we use up a given computational budget.

Simulations of a single timestep illustrate that the simulated annealing algorithm can achieve high competitiveness, with large fraction of non-empty solutions, even in the case of correlated accesses and a low computational budget. Simulations of the continued evolution of the portfolio show that the brute force approach does not always produce the best results, because the portfolio optimization problem considers the temporal evolution of the portfolio as a series of on-line optimization decisions, rather than an off-line retroactive problem. When accesses may have very different probability distributions for their expected benefits, our simulations of the evolution of portfolios illustrate the following:

- The longer that pending requests for access can remain under consideration, the more likely high-cost/high-return accesses can be allowed.
- The VO's capital increases over time.
- The VO's capital increases as the arrival rate increases, when the variance notion of risk is used.

The second part of the thesis presented *insured access*, the first demonstrably sustainable system for encouraging appropriate information sharing in a VO. Before insured sharing starts, VO members agree on the VO's degree α of risk aversion and its maximum tolerable level of risk, i.e., the chance ε that eventually the VO might not be able to compensate an information producer for damages attributable to sharing. The VO sets up an insurer whose actions are governed by α and ε. To obtain access to a piece of information owned by VO member Alice, VO member Bob must purchase a liability policy from the insurer. The insurer will not issue the policy if the VO would be exposed to more than its maximum tolerable aggregate level of risk as a result. Otherwise, the price of the policy is determined by the type of information, the insurer's current capital reserves, Bob's track record, the insurer's bonus-malus scheme, and the insurer's premium pricing principle. If Bob misuses Alice's information and Alice suffers damages as a result, then Alice can submit a claim and be reimbursed for her suffering. In the thesis, we showed the following:

- We showed how to estimate the risk associated with an insured access, i.e., the probability distribution of future damages to the producer. From history data, the insurer can compute how many policies in the class have had total claims of no more than $\$K$, for each value of K. Dividing by the total number of claims produces an approximation to X's distribution. If X belongs to a particular family of distributions, the claims history can be used to estimate the parameters of the distributions, using maximum-likelihood estimation. Extreme value theory provides a basis for statistical modeling of unseen tail events, i.e., damaging experiences that are not yet present in the historical record.

- We showed how reinsurance can cap the risk associated with rare events. To handle the risk of high-damage events it has never observed, an insurer can buy a stop-loss insurance policy from a reinsurer. The stop-loss policy transfers tail risks to the reinsurer and lowers the variance of the insurer's portfolio.

- We provided three schemes to ensure that information producers directly benefit from insured sharing. The first one is to wait until the end of a fiscal period, calculate the level of capital that the insurer must retain for probability of ruin ε, and distribute the excess capital among the producers. The second, more sophisticated methods distribute the insurer's funds in excess of the optimal dividend barrier, which maximizes the total expected present value of the distributions (dividends) before ruin. The third is a fee-for-service model.

- Our simulations of a map-sharing scenario showed that each participating VO member, and the VO as a whole, can expect to benefit from insured access, while the risk of failure of the system is limited by ε. The experiments also showed that consumers who rarely cause claims can benefit from the bonus-malus system.

We used Amazon Mechanical Turk to conduct experiments with humans and insured access. The experiments were based on modified versions of the Beer Game, a well-known supply chain simulation game. Each participant in the game represents one organization in a four-stage supply chain, where the entire chain forms the VO. The goal of the game is to maximize profits for each organization and the entire VO, by ordering just the right amount of beer to meet customer orders. Sharing of information on product orders is known to lead to much more efficient management of the supply chain. We trained the human participants in the Beer Game until they were reasonably proficient, and then let them choose whether they wished to see the order information of the entire VO using insured access.

The experiments showed that slightly more than half the time, the humans chose to use insured access when their VO offers it and they know it is likely to benefit them and the VO. The experiments also showed that players achieve better supply chain and distributor balances on average if they purchase policies and share information. Furthermore, instead of refusing to share their sensitive information, 71% of human players chose to allow other players to see their business status information under the protection of insured access. This illustrates that most humans are willing to provide information for insured access when it is relatively painless for them, rather than just denying requests to share their information. We also saw that human players who made computer players purchase policies made about twice as much money on average as those who denied access requests, when outliers are excluded. These results validate our claim that the introduction of insured access does encourage information sharing and produces better outcomes for the VO as a whole, even when human decisions are involved.

6.2 Comparing and Combining Portfolio Optimization and Insured Access

Portfolio optimization and insured access have different objectives: portfolio optimization seeks to maximize the benefits a VO can expect from information sharing, while insured access protects information producers and encourages them to share. Risk management is a central concern for both approaches, as it should be for every participant in a VO: individual consumers who request access, organizations that belong to the VO, the insurer, and the VO as a whole. Because each participant will have its own interests at heart, each participant will define and manage risk in its own unique way. This thesis assumes that the information-sharing risk models and risk management strategies of different VO participants can be layered to produce better outcomes for the VO. In particular, we reject the notion that risk could be effectively managed as a centralized, VO-level function alone.

For example, human consumers may often rely on instinct in choosing whether to pursue an opportunity that requires shared information. Consumers could instead use quantitative analysis based on decision theory to analyze the potential benefit and harm of a proposed action. They might explicitly consider opportunity costs and, as proposed in Formula 4.4, their degree of risk aversion, current wealth, and how much they would like additional money. In practice, mathematical analysis becomes more likely as one moves from individual human consumers to the higher levels of the VO. In particular, the insurer must rely on actuarial principles and historical data and act accordingly. On the other hand, human consumers may have the best understanding of the purpose for which access is requested, and this information is not always available for decisions made at higher levels of the VO. Thus decision making at each level uses different methods and models.

The diversity of participants' risk models is reflected in the models used by portfolio optimization and insured access. Both seek to bound risk, as measured by a probability distribution over potential outcomes of risky accesses. However, the two approaches include different aspects of risk in their models. The risk model used by insured access in Chapter 4 is for the harm caused to producers as a result of sharing information, while the risk model of portfolio optimization in Chapter 3 is based on the variance or tail distribution of the net benefit to the VO from the consumer's usage of the information. Further, the portfolio optimization risk model considers correlated outcomes, while insured access does not. As one risk model is used to choose between many possible accesses and the other is used to price insurance, these two risk models must necessarily remain distinct. However, several of their features can be altered, generalized, merged, or layered to achieve the VO's goals, as explained in the remainder of this section.

A VO can employ both insured access and portfolio optimization, under the following condition. Since the two approaches use different models for aggregate risk and total acceptable risk, portfolio optimization might recommend a set of accesses that the insurer refuses to underwrite because its chance of ruin will become too high. When this happens, the insurer must be able to tell the portfolio optimizer that its recommendation is unacceptable, so that the portfolio optimizer can recommend the next best option.

Portfolio optimization functions could be decentralized and moved to the VO member level, if benefit models are considered too sensitive to be shared at the VO level. Under this approach, a VO member could review the access requests from consumers in its own organization, and decide which to pass on to the insurer. This decentralization would have significant ramifications. First, the model for net benefits would probably be at the VO member level, rather than a VO-wide perspective. Even if a VO member has a VO-wide model of benefits, it may not be reasonable to expect the VO member to choose the local portfolio that is best for the entire VO, since the VO member is self-interested. In particular, extra measures would be needed to ensure that the risks of harm to producers are adequately covered in the VO member's model for net benefits.

Second, portfolio optimization's cap on risk will be enforced at the VO member level. Since some of this risk will be in the form of harm to producers, extra oversight will be needed to ensure that VO members are adhering to this cap. Further, the VO-level cap on risk will have to be apportioned among the VO members. With local models for benefits, no one will consider the correlation among accesses that originated from different VO members, so the risk assumed by the entire VO might vary in unpredictable ways from the caps enforced by individual VO members. For a particular situation, the VO will need to decide whether the advantages of increased privacy for VO members and potentially simpler risk models outweigh these disadvantages

The insurer's risk modeling and underwriting decisions could be decentralized and moved to the VO member level, if historical damage information and/or models of harm are considered too sensitive to be shared with a VO-level insurer. Adequate capitalization and oversight must still be provided for these decentralized insurers, ideally through a shared VO-level capital reserve fund for insurers and shared underwriting rules.

Insured access's risk model could be broadened to include collateral damage to VO members other than the producer, if the insurer has a perspective sufficiently broad to estimate the chances of collateral damage. For example, in a municipal or emergency response VO, suppose that Bob uses insured access to get a map from Alice that includes the location of a gas pipeline, and the map is eventually used to locate and blow up the pipeline. When subsequent investigation shows that Alice was the source of the map, her reputation will surely suffer, even though Bob's original purpose may have been benign. But other VO members may be hurt as well, either directly or by association. For example, at a minimum, operations will be disrupted for the VO transportation providers whose routes traverse the pipeline.

Consideration of collateral damage could be a slippery slope, as it could be difficult to draw the line where coverage ends. Consumers have their own purposes and goals in using shared information to pursue an opportunity. Each opportunity will have its own risks, including the possibility that the opportunity will turn out badly for the consumer or will cause collateral damage to other VO members. It may be difficult to distinguish between damage caused by the opportunity itself and damage attributable to sharing. As an example analogy, when Alice rents a car from Bob's Autos, she can use the car to pursue many opportunities that would otherwise be unavailable to her, from interviewing for a new job to robbing a bank. The insurance policy for the rental does not need to cover the potential damage caused by Alice shooting a bank guard, even though Alice would not have been able to rob the bank without renting a car. Nor should Bob's Autos expect to share in potential positive outcomes of the purpose for which Alice rented the car, such as the salary from her new job or the cash from the robbery. Thus the risk distributions used in Chapter 4 consider only the harm caused to the producer.

In a VO that supports insured access, **portfolio optimization's model for net benefits could be specialized to exclude harm to producers**, under the following conditions. First, the portfolio optimization group and the insurer must agree on the model for harm. If they do not agree, it is better to retain the two models, as a form of defense in depth. Second, portfolio optimization and the insurer must employ a feedback mechanism for rejected underwriting requests, as the VO must do in any case when the two approaches are layered.

In theory, **portfolio optimization and insured access could integrate their models for feasibility and underwriting decisions**, so that portfolio optimization never recommends an access that the insurer refuses to underwrite. However, the integration would amount to layering the two models, for two reasons. First, the variance of the net benefit of an access and the variance of the harm of an access are distinct considerations, and the same cap cannot sensibly be used for the aggregate of these two measures. Second, in reaching underwriting and feasibility decisions, both approaches consider all accesses that have been requested and whose outcomes are not yet known; but the insurer must also consider the current state of its capital reserves, while the portfolio optimizer does not consider any such value, e.g., the VO's current wealth.

Finally, **insured access could consider correlated harm from accesses**, much as portfolio optimization can choose to consider or ignore correlations.

6.3 Directions for Future Work

Ultimately, the goal of this research is to encourage appropriate information sharing in real organizations. However, there is a significant hurdle in going from models of the behavior of ultra-rational participants to actual human decision makers. The crowdsourcing experiments in Chapter 5 helped us to understand the impact of having humans as consumers, but much more can be done to understand the psychological aspects of human decisions in information-sharing scenarios. The remainder of this section outlines potential additional crowdsourcing experiments with insured access that could be conducted using, for example, a supply chain simulation such as the Beer Game.

The most severe test of insured access is likely to occur after a series of adverse events, such as repeated and significant harm to producers, poor outcomes for consumers, a brush with ruin for the insurer. The VO or some of its members might prefer to abandon insured access at that point. A VO can reach this point even if risks have been properly modeled and managed, because if we repeatedly flip a coin, eventually we will get ten heads in a row. To examine this phenomenon with humans, we can give skilled players the choice of whether the VO will play its next games with or without insured access. Then the humans play one or more games under their preferred paradigm, but with a continual stream of adverse events and poor final outcomes for the VO and its insurer. After this, we ask the humans to choose whether their VO will play the next game with or without insured access, and examine how their preferences have been affected by intervening events. The process can be iterated if desired.

Our current versions of the Beer Game make adverse events almost pain-free for players, by announcing in a popup that harm has occurred, claims have been filed, and the affected players have been reimbursed for a certain amount by the insurer. This circumvents the real-life problems of recognizing that one's organization is suffering in some way, tracing back the cause of the damage, developing a convincing argument that information sharing caused the damage, filing a claim, and perhaps appealing the result. Each of these steps can be tested, as follows.

- The players can be told that orders can get diverted to a competitor because of leaked shared information, and that it is the players' responsibility to detect when this is happening and file a claim. Humans like to attribute causality even to random events, so we anticipate that some players will think that every decline in incoming orders is due to information sharing. Inattentive players are likely to make the opposite mistake, overlooking all but the steepest declines. We can measure these effects on the players' preference to play future games with or without insured access.

- Identification, investigation, filing claims for damages, and filing insurance appeals can be annoying diversions for an organization or player. Time is money, both for organizations and Amazon Mechanical Turk workers, and if the insurance claim process is sufficiently painful, information producers will not want to participate in insured sharing. The game can mimic this annoyance by having players click extra buttons to indicate whether they want to file a claim, why they want to file a claim, ask them whether they want to appeal a proposed settlement, and so forth. The game could also introduce artificial delays via a popup window with words to the effect of "Your claim is important to us. Please hold on while the insurer investigates your claim." We can measure these effects on the players' preference to play future games with or without insured access.

In our experiments with insured access, we told players how much the shared information affected the final cash balance of the VO for computer players. Players can also see their own past games with and without insured access, to estimate how much the information was worth to them based on their own empirical data and to help them in deciding whether to purchase a policy. By varying the price of the policy, we could examine how humans value the shared information, which could be helpful for a VO as it decides which information is worth sharing. Further, if players are accustomed to paying a certain price for insured access policies, then they will be subject to the psychological phenomenon known as *framing*, where people view the usual price as the right price. Even if the damage distribution changes radically, we would expect them to resist price increases and enthusiastically endorse price drops. Framing could lead to irrational consumer decisions if the shared information is no longer as beneficial for consumers as it once was, e.g., if end-customer orders are made public, or if premium prices change significantly due to a bonus-malus scheme.

Appendix A

IRB Approved Documents

A.1 Online Consent Document

For a study of incentives for information sharing

You are invited to participate in a study of the effect of incentives on the sharing of information among organizational partners. This study is conducted by Naoki Tanaka and Marianne Winslett, who are researchers in the Department of Computer Science at the University of Illinois at Urbana-Champaign.

For this study, you will be asked to play versions of an online business game that simulates a four-stage product supply chain: retailer, distributor, wholesaler, and factory. During each game, you will represent one of these four organizations. At each turn, you will be responsible for determining how much product to order from your supplier, based on the stock levels in your warehouse, your customer "/s latest order, and your previous orders.

As your level of skill grows, you may be invited to play different versions of the game. Some versions may allow you to see the orders or warehouse stock levels of other organizations in the supply chain; in other versions, you will only be able to see your own warehouse and order. Depending on the version of the game, your game may last for as little as one turn or as long as a hundred turns. You can play a particular version of the game no more than ten times with a particular set of partners.

At the end of each game, you will receive a small cash payment through the crowdsourcing service. The amount of your payment depends on how well you and the other players in the supply chain fulfill new orders, without keeping a lot of unneeded product in the warehouses. The per-turn rate of pay is set so that a proficient player, i.e., one who has learned to play the game and therefore makes product order decisions reasonably quickly, will make Illinois minimum wage ($8.25/hour) or higher. In addition, players who manage their supply chains particularly well will earn a variable bonus that can in effect double their per-turn rate of pay. The research team calculates bonuses every two weeks at a minimum, and informs the crowdsourcing service of the amount of bonus to be paid to each worker.

Your decision to play the game is completely voluntary, and you can stop playing at any time without penalty. If you do not want to continue with a game session, just close the browser window where you are playing.

Your participation in this research is completely confidential. Information about how you played the game will be reported only under a pseudonym, without any personally identifiable information about you. The results of the study may be reported in technical reports, articles in journals and conference proceedings, a thesis, and seminar and conference presentations.

There are no risks to individuals participating in this study, beyond those that exist in daily life.

For UIUC students or affiliates: your decision to participate, decline, or withdraw from participation will have no effect on your current status or future relations with the University of Illinois.

If you have questions about this project, you may contact Naoki Tanaka at tanaka5@illinois.edu, or Marianne Winslett at winslett@illinois.edu.

If you have any questions about your rights as a participant in this study or any concerns or complaints, please contact the University of Illinois Institutional Review Board at 217-333-2670 (collect calls will be accepted if you identify yourself as a research participant) or via email at irb@illinois.edu.

Please print a copy of this consent form for your records, if you so desire.

I have read and understand the above consent form. I certify that I am at least 18 years old and younger than 65 and, by clicking the Sign Up button, I indicate my willingness to voluntarily take part in the study.

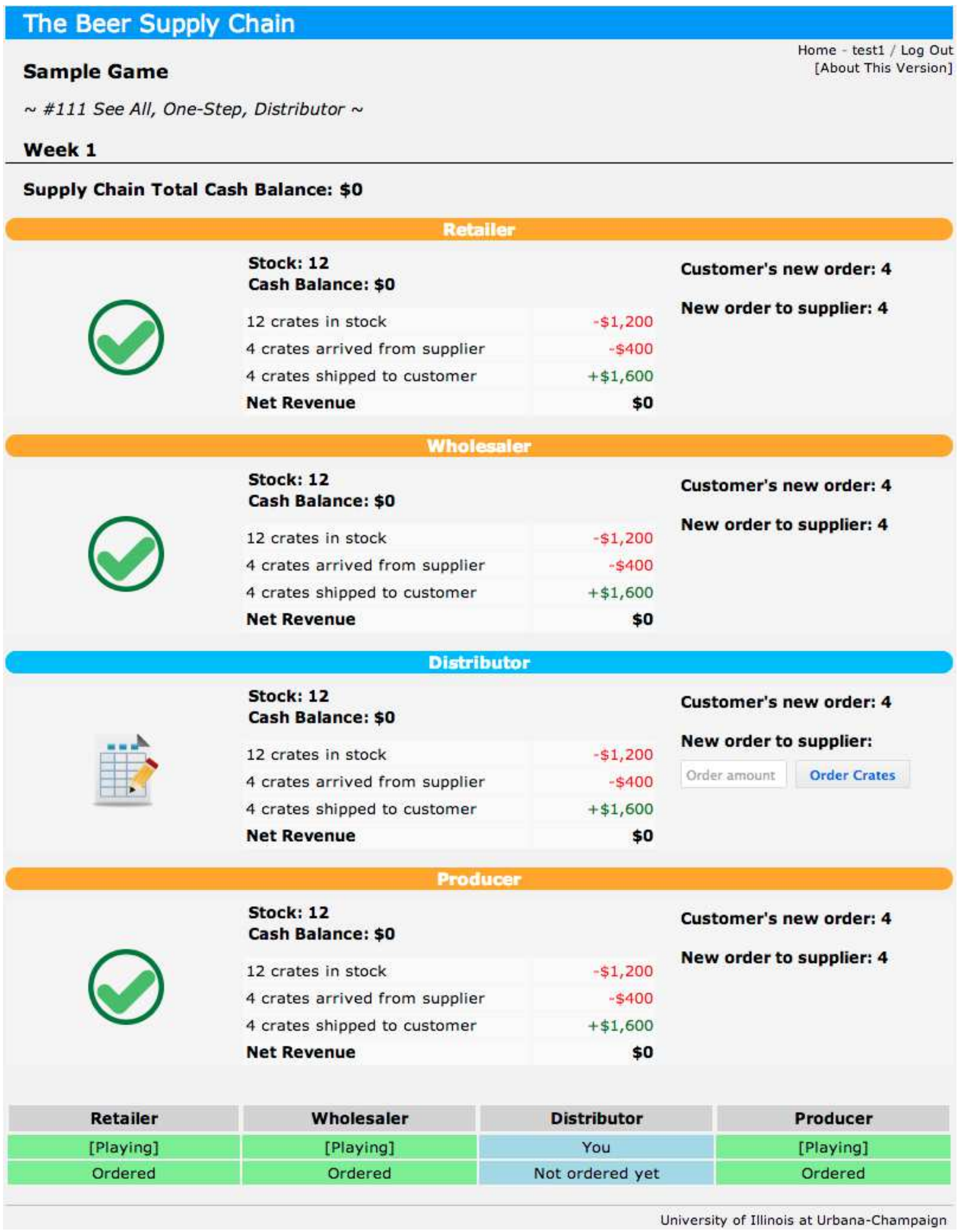
A.2 Recruiting Text

Play an online business simulation game!

In this game, you represent one of four companies in a supply chain: a beer manufacturer, distributor, wholesaler, or retailer. At each turn, you must decide how much beer to order from your supplier, based on how much you already have in your warehouse, how much your customer orders from you, and your previous orders that have not yet arrived. The goal is for everyone to be able to fulfill their customers "/ new orders, without keeping a lot of unwanted beer in the warehouse. It sounds easy, but it isn't!

If you manage your supply chain very well, you will earn a bonus that can double your rate of pay. If you play very well, we will invite you to play other versions of the game.

If you are a beginning player, the game will last for about 50 turns and should take you less than an hour. Your screen will look like this:


The Beer Supply Chain
Home - test1 / Log Out
[About This Version]
Sample Game
~ #111 See All, One-Step, Distributor ~
Week 1
Supply Chain Total Cash Balance: $0
Retailer
Stock: 12
Cash Balance: $0
12 crates in stock -$1,200
4 crates arrived from supplier -$400
4 crates shipped to customer +$1,600
Net Revenue $0
Customer's new order: 4
New order to supplier: 4
Wholesaler
Stock: 12
Cash Balance: $0
12 crates in stock -$1,200
4 crates arrived from supplier -$400
4 crates shipped to customer +$1,600
Net Revenue $0
Customer's new order: 4
New order to supplier: 4
Distributor
Stock: 12
Cash Balance: $0
12 crates in stock -$1,200
4 crates arrived from supplier -$400
4 crates shipped to customer +$1,600
Net Revenue $0
Customer's new order: 4
New order to supplier:
Order amount
Order Crates
Producer
Stock: 12
Cash Balance: $0
12 crates in stock -$1,200
4 crates arrived from supplier -$400
4 crates shipped to customer +$1,600
Net Revenue $0
Customer's new order: 4
New order to supplier: 4
Retailer Wholesaler Distributor Producer
[Playing] [Playing] You [Playing]
Ordered Ordered Not ordered yet Ordered
University of Illinois at Urbana-Champaign

How to play:

1. Accept this HIT.

2. Open a new browser window or tab, and go to https://oak.cs.illinois.edu.

3. When you finish playing, be sure to return to this window and enter your completion code in the box below, so that you "/ll get paid!

Completion code: [text field to input a completion code]

[Submit Button]

Appendix B

Screenshots of Beer Game Web App

Figure B.1 to B.9 are screenshots of our web app. Figure B.1 to B.5 are screenshots for #321 Insurance, Steady, Distributor. Figure B.6 to B.8 are screenshots for #421 Share & Keep Money / Share & Require Policy, Steady, Distributor. Figure B.9 is a screenshot for #521 Deny / Share & Require Policy, Steady, Distributor.

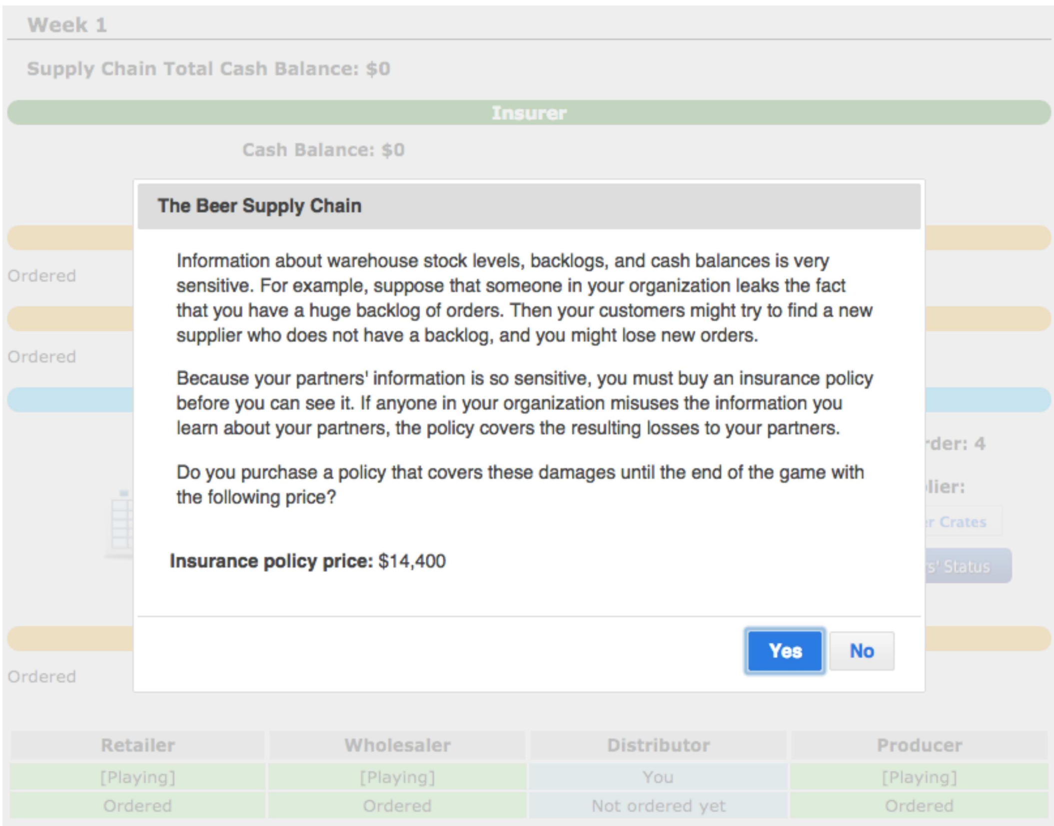


Figure B.1: Screenshot of Beer Game Web App (playing #321 Insurance, Steady, Distributor 1)

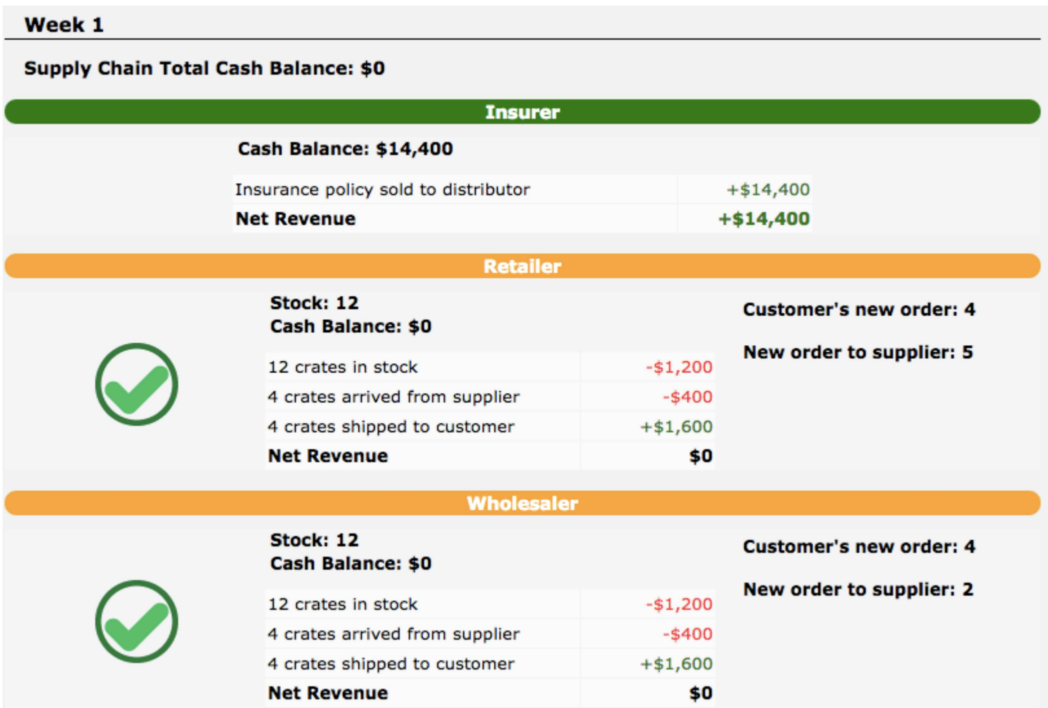


Figure B.2: Screenshot of Beer Game Web App (playing #321 Insurance, Steady, Distributor 2)

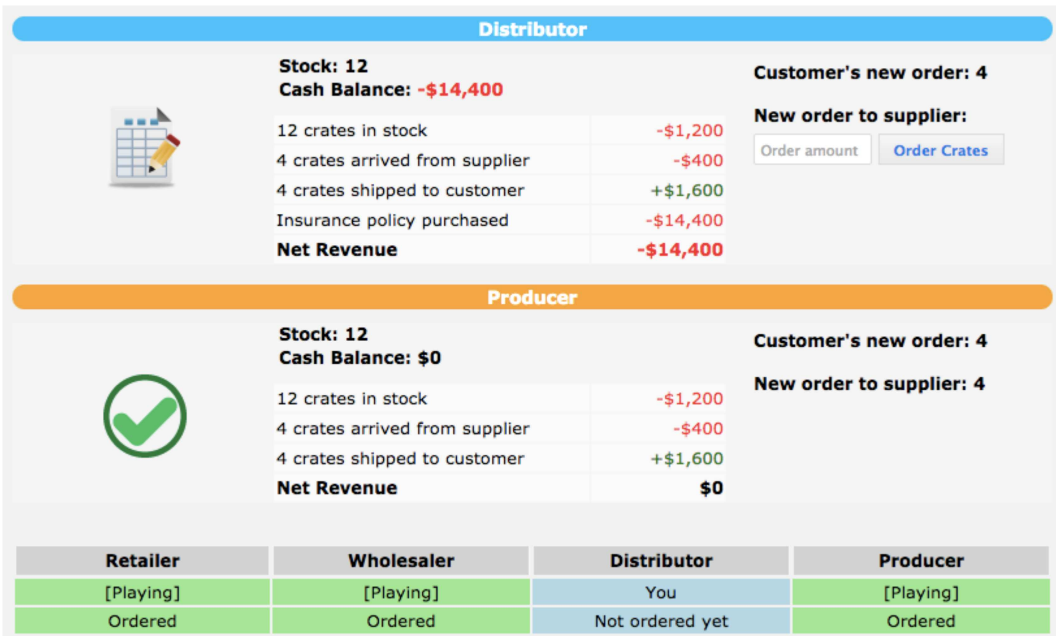


Figure B.3: Screenshot of Beer Game Web App (playing #321 Insurance, Steady, Distributor 3)

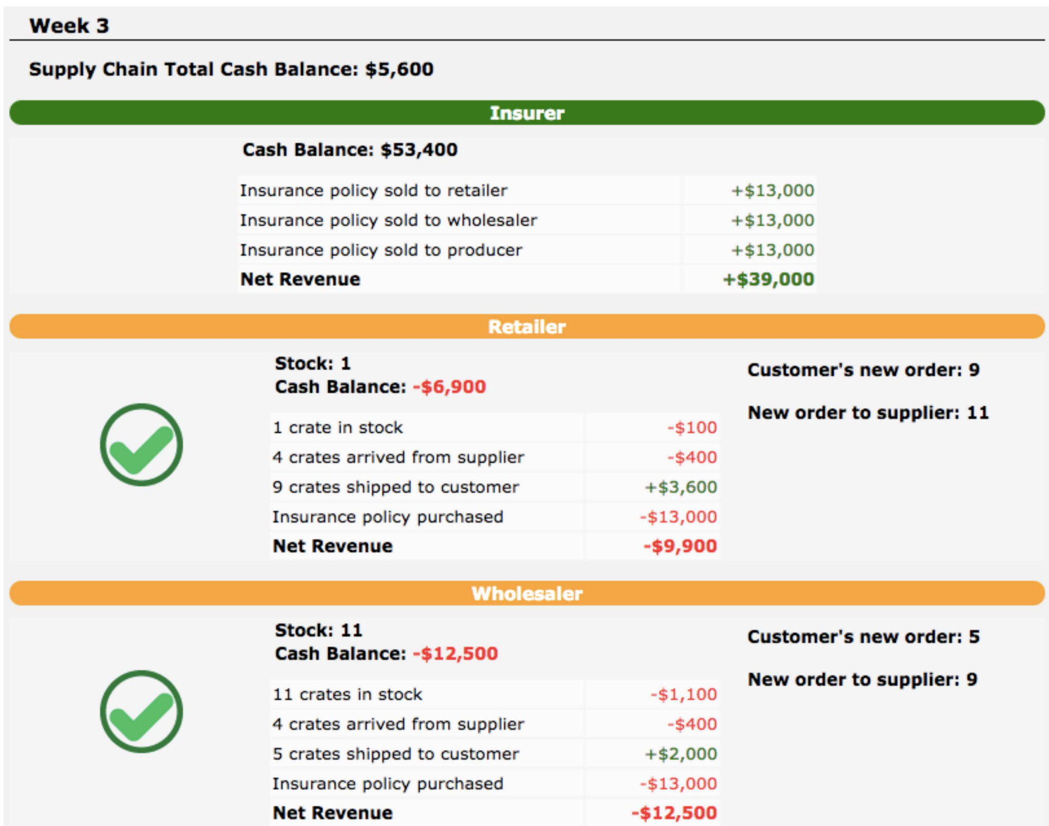


Figure B.4: Screenshot of Beer Game Web App (playing #321 Insurance, Steady, Distributor 4)

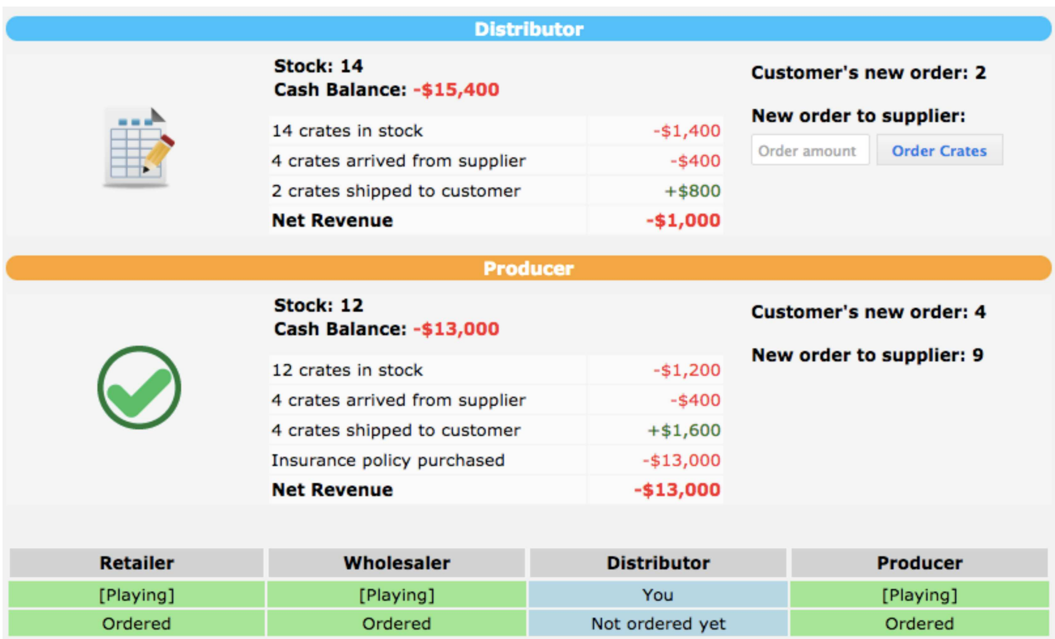


Figure B.5: Screenshot of Beer Game Web App (playing #321 Insurance, Steady, Distributor 5)

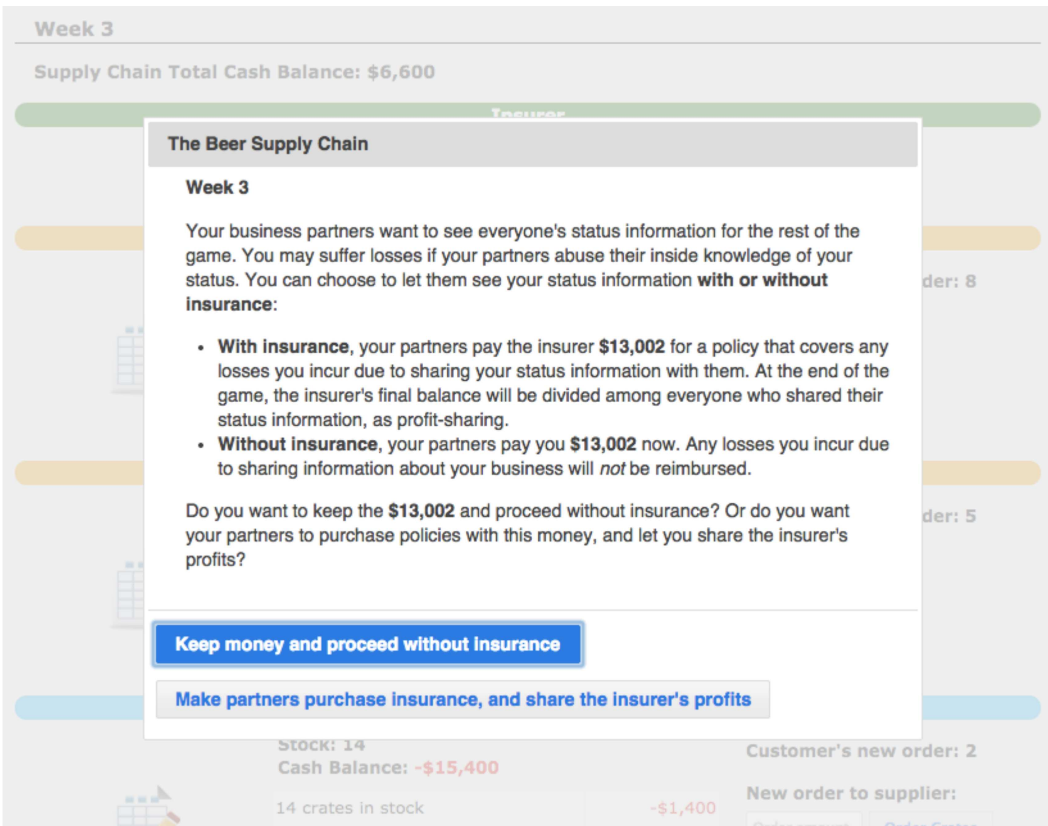


Figure B.6: Screenshot of Beer Game Web App (playing #421 Share & Keep Money / Share & Require Policy, Steady, Distributor 1)

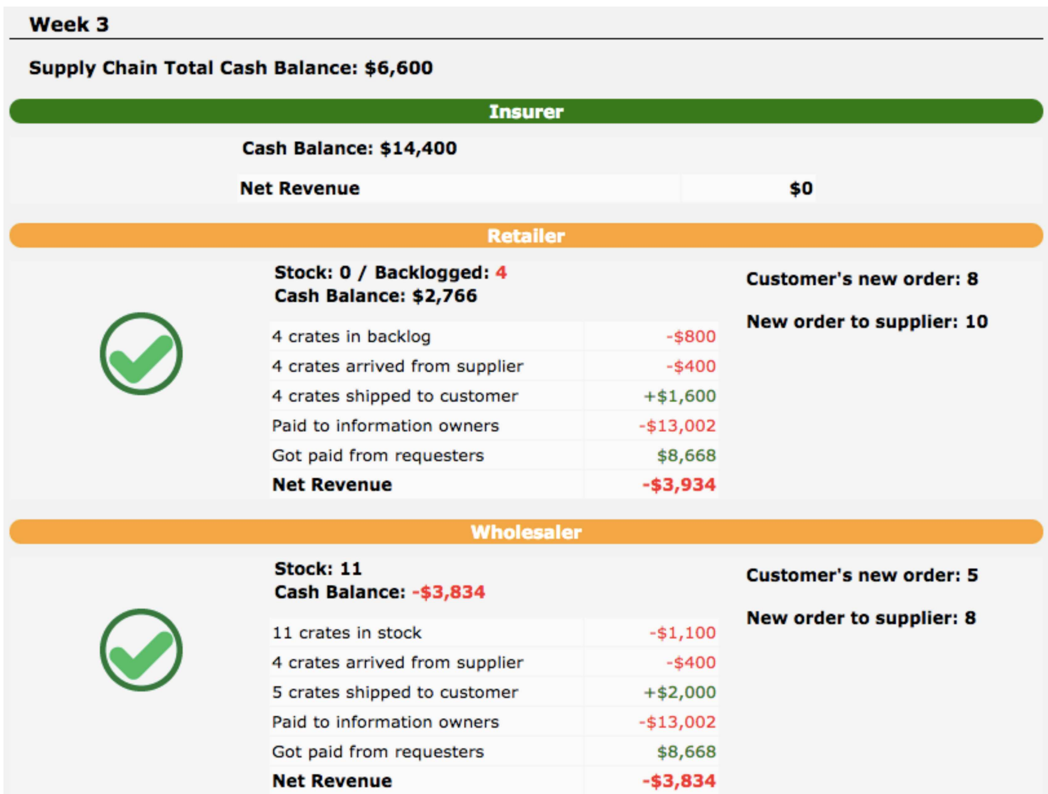


Figure B.7: Screenshot of Beer Game Web App (playing #421 Share & Keep Money / Share & Require Policy, Steady, Distributor 2)

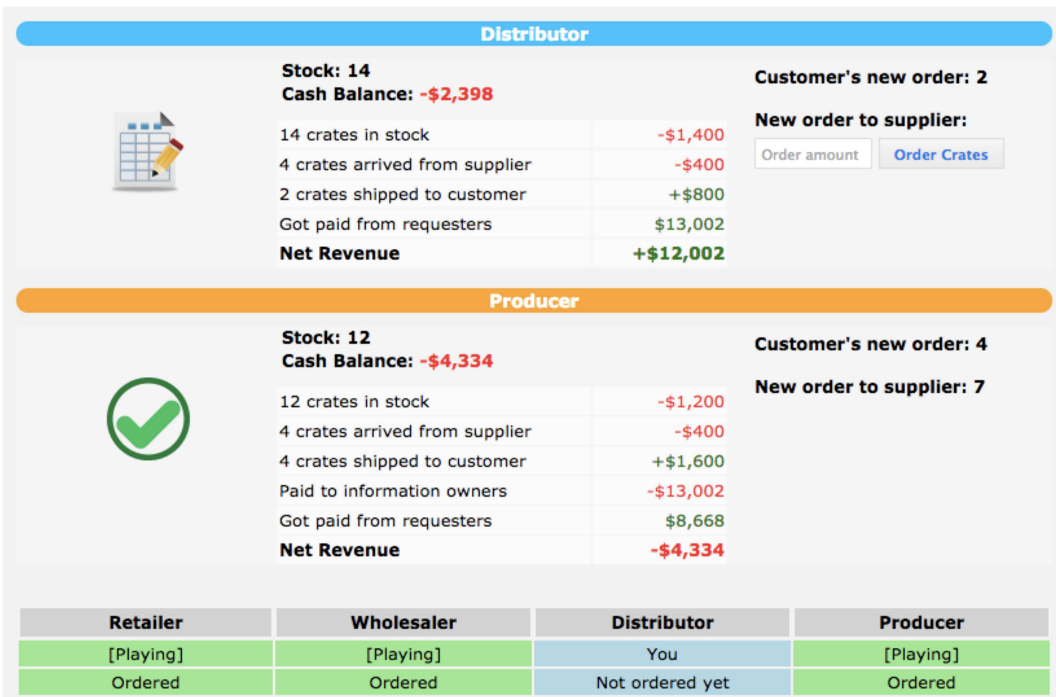


Figure B.8: Screenshot of Beer Game Web App (playing #421 Share & Keep Money / Share & Require Policy, Steady, Distributor 3)

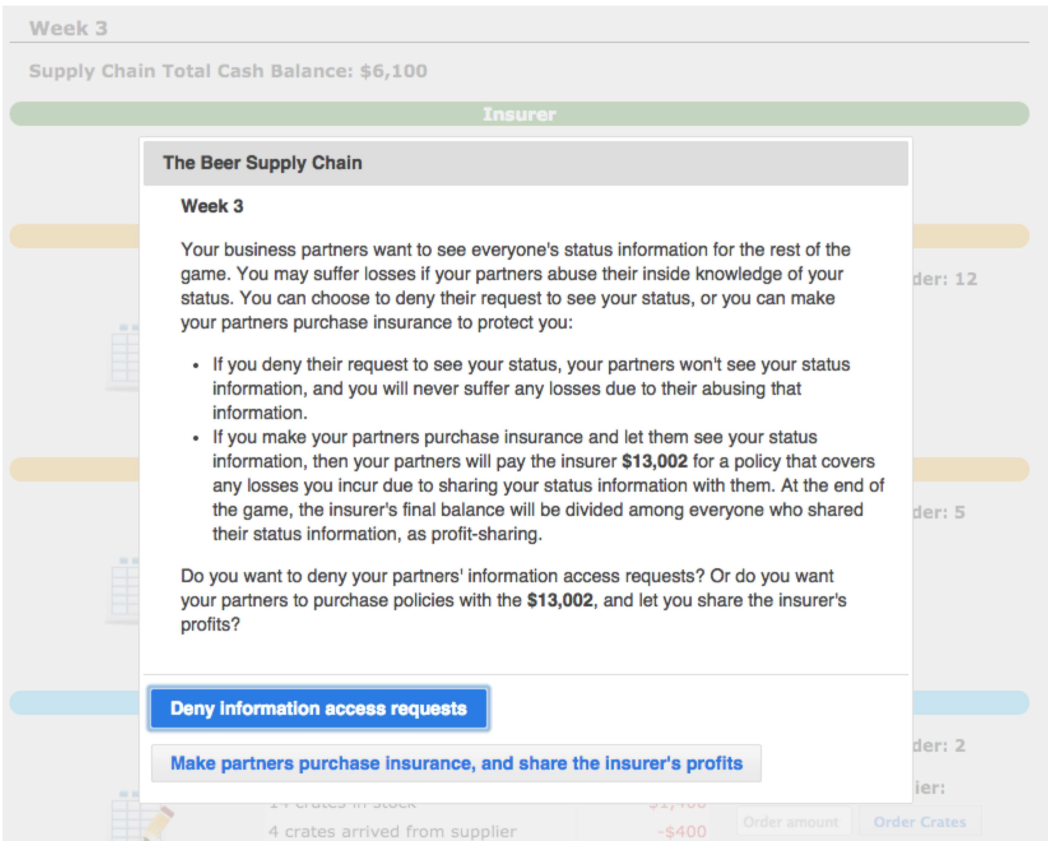


Figure B.9: Screenshot of Beer Game Web App (playing #521 Deny / Share & Require Policy, Steady, Distributor)

Appendix C

Communications with Crowdsourcing Experiment Participants

C.1 #111 See All, One-Step, Distributor

All new players should watch the 4-minute video tutorial below, which explains the basic features of *all* versions of the game.

In this *very* simple version that you are about to play, the demand for beer is completely steady, except for a one-time increase early in the game. You should stabilize your orders for beer as quickly as possible after the demand increases. In this version *only*, your partners are not very smart: they just order the same amount of beer as their customers request from them.

Warning: backlogs are bad, because they mean you don't have enough stock in your warehouse to fulfill your customer's orders, and so you have to pay a lot of penalties and fees. To avoid or get rid of a backlog, order extra beer (though not too much, because you also have to pay for storage). You will have to wait at least four weeks for your extra beer to arrive, so you need to think ahead and order promptly.

If you are one of our top performers on the games you submit for payment, you will qualify to play more complex versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like.

[Video Tutorial [5]]

C.2 #121 See All, Steady, Distributor

In this version, the demand for beer remains roughly steady throughout the year, though it does go up and down a bit every week. Your business partners are very good at figuring out how much to order. Can you play as well as they do?

Warning: backlogs are bad, because they mean you don't have enough stock in your warehouse to fulfill your customer's orders, and so you have to pay a lot of penalties and fees. To avoid or get rid of a backlog, order extra beer (though not too much, because you also have to pay for storage). You will have to wait at least four weeks for your extra beer to arrive, so you need to think ahead and order promptly.

If you are one of our top performers on the games you submit for payment, you will qualify to play more complex versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like.

The 4-minute video tutorial below explains the basic features of *all* versions of the game.

[Video Tutorial [5]]

C.3 #221 See Only Yours, Steady, Distributor

In this version, you cannot see your business partners' stock levels and orders. The demand for beer remains roughly steady throughout the year, though it does go up and down a bit every week. Your business partners cannot see each others' stock levels and orders, either, but they are still very good at figuring out how much to order. Can you play as well as they do?

If you are one of our top performers on the games you submit for payment, you will qualify to play more complex versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like.

The 4-minute video tutorial below explains the basic features of *all* versions of the game.

[Video Tutorial [5]]

C.4 #321 Insurance, Steady, Distributor

Many of you commented that the game got a lot harder when you couldn't see your business partners' status. In the real world, companies don't like to give out detailed information about their orders and inventory, because the information could be misused by their partners. For example, if your company is developing a backlog, your best customer might decide to order from one of your competitors who promises faster delivery. If you have too much stock in your warehouse, a customer might threaten to order from your competitor unless you cut prices. Your competitors might learn about your ordering strategy and copy it. So although everyone would like to see detailed information from their partners, no one likes to give out much information about their own company, even though everyone recognizes that sharing information usually leads to better supply chain management overall.

In this version of the game, the supply chain partners have agreed to share information with each other, in spite of the dangers outlined above. However, information sharing is not free. If you want to look at your partners' information, you have to pay for an insurance policy that covers the potential damage to your partners if your company misuses the shared information. Similarly, if your partners look at your information and abuse that privilege in some way, and you lose business as a result, then you can file a claim with the insurer for reimbursement of the damage to you. In the game version you are about to play, almost all of this happens automatically. You just have to decide whether you want to pay the cost of the insurance policy.

To help you figure out whether you want to pay for the insurance, you may want to know how much the information is worth to your supply chain. Looking at the best games that each of you played, we see that **your supply chains made about $115,000 more on average when you could see all information in the chain**, compared to when you could only see your own company's information. But everyone is different – some people made less money when they could see the entire chain's information!

In this version of the game, the demand for beer is roughly steady throughout the year.

If you are one of our top performers on the games you submit for payment, you will receive a bonus and may qualify to play additional versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like. If you need to, feel free to review the 4-minute video tutorial [5] that explains the basic features of *all* versions of the game.

C.5 #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

In this version of the game, when your partners want to see your status information, you can either (1) have them purchase an insurance policy as usual, or (2) you can pocket the money that they would otherwise have used to purchase a policy to protect you, and you can proceed without insurance. If you do make your partners purchase insurance, then at the end of the game, you'll get a share of the insurer's profits (if any).

Our challenge to you: can you move up in the rankings this time?

In this version of the game, the demand for beer is roughly steady throughout the year. If you are one of our top performers on the games you submit for payment, you will receive a bonus and may qualify to play additional versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like. If you need to, feel free to review the 4-minute video tutorial [5] that explains the basic features of *all* versions of the game.

C.6 #521 Deny / Share & Require Policy, Steady, Distributor

In this version of the game, when your partners want to see your status information, you can either (1) have them purchase an insurance policy as usual, or (2) you can deny their request to see your status information. If you do make your partners purchase insurance, then at the end of the game, you'll get a share of the insurer's profits (if any).

Our challenge to you: can you move up in the rankings this time?

In this version of the game, the demand for beer is roughly steady throughout the year. If you are one of our top performers on the games you submit for payment, you will receive a bonus and may qualify to play additional versions. You can be paid only once for each HIT, but you are welcome to play additional games for fun if you like. If you need to, feel free to review the 4-minute video tutorial [5] that explains the basic features of *all* versions of the game.

Appendix D

Comments from Crowdsourcing Workers

D.1 #111 See All, One-Step, Distributor

- Wow, I did horribly. HORRIBLY.
- Great and fun! Just a bit confusing. I did horrible but I think I would definitely play again!
- Thank you!
- really confusing game :(
- that was alot of fun!
- This task was very very confusing to me as each time I ordered crates I was always in the minus and not make any money. Very confusing study. Sorry just my feedback. Thank you
- I got to say this is the best hit I've done on Mturk by far, I hope my information helps you and I would be interested in helping again anytime! Have a great day.
- ?? don't understand what you did here. The linked page requires a password, and you never gave us a password. That's a stupid error to make, and you wasted my time. Just my two cents, do with that as you wish.
- Thanks.
- this game was very hard

- Was king of boring after a while because its too easy
- Excellent survey, thank you.
- That was fun!!
- Interesting.
- The game froze at about week 40 so I started the game over. The second time the game froze at week 17. I am submitting this because I put a lot of time into the game and didn't do anything wrong to make the game not work for me. I sent an email when the game first froze, asking for guidance but I haven't heard anything yet.
- This was remarkably well presented. The game elements were coordinated effectively to represent the salient points. It's apparent that once supply begins to fluctuate, a "just in time" strategy will entail serious risks which were not present in this tutorial. Thank you for the opportunity to provide data in this study!
- fun, but not realistic since they kept ordering the same numbers repeatedly
- I didn't realize the computer players wouldn't attempt to fix their backlogs, so I might have had a sub-optimal score, I'm sorry.
- I was awful lol
- It was difficult and confusing. I wish there was a chance to start over and play a second time around.
- Games was hard lol so unpredictable
- yeah, that wasnt as accurate as it should have been
- I had to restart the game, I had a problem with it loading the first time.
- That was very fun, although the 'dumb' AI was a little annoying. I realized early on that my backlog escalated up the supply chain, but they wouldn't take more than 8 per turn to clear out theirs. Over the course of the game, that cost the total supply

chain some $100,000. I hope I can play your more advanced game, it seems like it would be a great challenge

- I did terrible. :(
- I really suck at this. I'm so sorry
- I started getting the hang of it in the middle, then is started going all downhill again somewhere in weeks of the 40's. Was fun though.
- interesting
- I would have liked to see more obvious the extra charges for over and under stocking. I think I got behind and could never catch up.
- It did seem a little difficult to follow what was happening; I was doing well then things changed and I was never able to recover. I saw my backlog of inventory, and I saw the orders being placed, but I didn't see the orders being filled/shipped from my inventory. Perhaps I misunderstood something in the video directions, because I remember it being mentioned that the other parts were played by computer, so if there is a problem it is because of the person playing.
- Couldn't quite figure out how to reduce my backlog
- Difficult, need to play a few more times and go over the instructions again to get the hang of it.
- fun! I'd love to do more like that
- I didn't realize there was a lag between the order placed and when it would be received.
- This was fun!
- Fun game, I enjoyed it!
- Took most of the year to realize that "backlog" is not the same as "inventory".
- I wish the directions would have explained better how to get rid of the backlogs.

- Interesting. I kept ordering 0 and kept getting 8 cases. Why??? Thanks.
- None
- Cool game!
- Fun game! I'd like to play again and do better.
- That was awesome!!
- Interesting!
- That was a lot of fun. I'd love to play again in the future.
- After I was playing for a while I understood a little better how to form a strategy.
- this was very fun
- I feel like a moron now.
- so much fun! i loved this game
- Thank you for letting me participate in the task.
- This was really fun
- That was fun
- none
- I wish I could play this some more...I bet I could get better at it.
- Maybe the game wasn't explained very well but I had a lot of trouble. I kept trying to get rid of my backlog by not ordering, but that just made it worse. Then I couldn't get out of the hole I dug for myself!
- You need to pay more for this. Standard pay is .10/min.
- Was pretty interesting once I got the hang of it.

- In the middle I was confusing "stock backlog" with "stock oversupply".... Could never recover from that...
- Very tough but fun to play.
- the game kept freezing up on me several times
- It was actually fun. I think I could do much better if this comes up again though.
- I did just terrible! I didn't understand how I did so poorly. I tried to give them as much as they needed, ended up giving too much, and then gave them none so they could get rid of their back stock - but then the distributor took a hit. :(
- FUN
- WOW!!! That was fun! Unfortunately, I messed up, as I did not get the "hang" of it until further into the study. Don't misunderstand me though, I understood the instructions, but keeping an eye on all the moving parts was very complicated and took some time to understand how to effectively comprehend/interpret the data.
- I need to find a workable strategy for this game! Feel that I don't have a clear idea of how to keep the inventory stocked, but not over stocked.

D.2 #121 See All, Steady, Distributor

- I appreciate it.
- I lost it right at the end. I was thinking 50 weeks and slowed my orders and then got thrown out of my zone and misread the backlogs as in stock.
- so much fun!
- Thank you for letting me take part in the survey. Am not sure why, but i get a feel that I performed poorly in todays task. :(
- I forgot to accept the HIT before playing the same, if you're wondering about the time :)

- Thank you for the helpful. The game was more exciting this time.
- Clicking on the "ok" after the week notification is a bit tedious. Having a week indicator at the top would be nicer.
- I like this game.
- I think this was way more interesting than the last one I did where the orders were a steady 8 for 30 weeks. Love the graphs at the end! Data is beautiful
- the instructions were hard to understand.
- OK, now I really do have a feel for how the simulation works! I felt like I was starting to make better and better decisions as the game went on and I learned how to handle bottlenecks and increases in demand. I'd love to be invited back for future iterations of the simulation.
- That one was harder. Took me a bit to get in the flow. Hope I qualify for more HITs
- I love your game! IT was a blast.
- Fun Study!
- It was more fun this time. I enjoyed it.
- This one was tougher. My strategy didn't work at all.
- Thank you for the increasing the HIT payment
- Very fun game! I am looking forward to more HITS. :)
- This is fun, thanks!
- That four week delay makes it difficult to be as responsive, especially with changing demand. It's an interesting game and definitely more challenging than the last. If I were the retailer, I might think about producing my own beer (of course, there would still be the issue of supplies).

- I enjoy this challenge. I'd love to be able to participate in more.
- Cool game!
- Interesting
- I really enjoy these hits, they have me thinking really hard! Hopefully I did well again this week and I can keep participating. Thank you for the bonuses and qual and opportunity!
- More fun than the first version
- Made $99,900, man I was really feeling that 100k =P
- Most fun I've had on a HIT
- I love this game!
- Did MUCH better this time, any comment on what my scores should be around?
- This is so fun! i hope i qualify for more!
- I am enjoying this game! Unfortunately I royally screwed up. Would love to keep playing it to figure this thing out. The tutorial doesn't do the game justice.
- That was even more fun than the previous round. The fluctuating numbers of the orders coming in kept it pretty realistic and made me really get motivated to do my best.
- This simulation really is great. I like it even more every time I play it. I think I'm going to run it a couple more times just to practice.
- I love this game!
- Thank you for allowing me to provide data. Your time limit amply allowed me to "rest up" before I worked this HIT, which I appreciated. ;)
- i can't believe i am doing so poorly at this

- Thanks for letting me do another one of this. This one was more difficult but still fun
- Love the game, I think I've got a basic strategy down solid now, I could score a bit higher if I tried again, but to really maximize I think it would require so much tracking of each step that it would be like a real job.
- interesting game
- This is actually quite fun. Thank you for the increased pay as well.
- Thank you for all the opportunities so far!
- Lots of fun!

D.3 #221 See Only Yours, Steady, Distributor

- That was terrible x_x I feel like I did really bad this week, which is unfortunate, because I enjoy the hits, but oh well :D Thanks for the ride, it was fun!
- Interesting
- Wow, this has been really interesting and educational!
- I can't believe how much more difficult this version was. I've been trying to figure out a spreadsheet but keeping track of everything coming and going without knowing if my supplier has stock or backlog is tough. Thanks for the challenge.
- I really enjoy playing this game - I like trying to figure out how to make my part of the supply chain work smoothly.
- Man that one was really hard, it was a lot easier when I could see what the others were doing as far as orders. I couldn't catch the fluctuation in this round, I did terrible. Still fun though.
- Ugh. This was awful. I should have done much better.

- Wow, that was insanely hard. Not to mention the fact that the demand ramped up so fast, and having a 4 week break put me into the backlog super quick. I failed to take into account how much going into the backlog hurts as each case there costs double what it costs in the warehouse. Being able to see the trends before they happen (111 and 121) make a world of difference...I was a lot more reactionary in this one than I was in the other two. I find it hard to believe that the 'best' players will be able to make over 200k for their supply chain. In real life I could see how certain times of year could be more trendy for beer drinking and therefore even distributors with no other information from wholesalers and retailers could prepare accordingly, but working purely off what was provided for a customer order was incredibly hard. I planned to eliminate my warehouse stock as fast as possible, but I don't have any control over what I receive in the first four weeks, as my first order isn't processed until week 3 and shipped until week 5. Those first few weeks hit my warehouse pretty hard, and therefore my planned strategy of not ordering a whole lot (0 in fact) for my first two orders supremely backfired and put me in back order mode. Then demand rose as I tried to get my stock back and though I was over-ordering what the customer wanted each week, by the time it was fulfilled the wholesaler was ordering MORE than I had ordered. The there was the great 4 week prohibition where I think the wholesaler had too much in stock so they stopped ordering. *sigh* I guess I'm just trying to explain why I did so poorly and it basically comes down to those first two 0 case orders I made. I was too greedy.
- This version was a lot harder at times. The graph of crates in stock/backlog was very interesting to look at though this time around.
- I was doing pretty good until week 40, and then I just messed up. I will try this round a few times just to teach myself how to keep ahead of the orders I cannot see. Love this game! It is addicting to see how I can best profit.
- this one was way harder then the last two. i tried to get the hang of it near the end but i still did terrible :(. i'd like to try again. thanks and fun game.

- That was much harder! Thank you for the invite.
- I think the difficulty is unrealistic... It would make more sense if the first "see none" level shifted once or twice only with little variations rather than 2+ with multiple variations. thanks though! its still fun!
- Cool!
- I love these HIT's
- It was much harder not seeing the other parts of the supply chain. I could not plan accordingly.
- Love this game!
- Honestly this is one of the funnest hits I have done. I did not let my concentration lapse this time like the last round. It would be fun to be able to do some practice rounds. This is the kind of thing that you just want to have fun with and get good at, almost like a video game. This round was really different and almost easier because I did not have to try to concentrate on the other players. I hope you guys are getting good data because these hits are really fun.
- It seemed completely unpredictable, unlike the previous 2 versions.
- Not seeing what was going on with the other links in the supply chain definitely made it more challenging.
- Wow, this was the most challenging one yet. I actually ended up deep into the negatives as far as profit. I did not expect that at all. I panicked and over ordered and it shows in my stats. But still, as always it's a great game and I would love to give it another try and do better next time.
- Version #221 was really hard to try not to get backlogged since you can't see any of the other orders, but I enjoyed the challenge.
- This is fun, thanks!

- ouch!
- I found this version easier to play because I was just focusing on my numbers and not watching what everyone else was doing. Again, I love this sort of game. I'll gladly play again if you let me!
- Was a lot harder this time. But still fun.
- This one was harder.
- Great game! Harder this time, but still fun!
- This game is pretty fun :) I have played it several times and I really like it!!
- this is very interesting!

D.4 #321 Insurance, Steady, Distributor

- The explanation given for the hidden information and insurer were easy to understand and informative. Well done.
- Thanks again for the invite! I did a little better this time, even without insurance.
- While buying insurance at the beginning was costly, it paid off in the end and I was able to make a substantial profit this round.
- The insurance was interesting, but it lost all of it's money fast. I think not knowing what everyone had would of been better in my game's scenario.
- I totally rolfed it this time. I wanna rematch :-D
- It was fun with new interesting twist to the game.
- It's kind of sad to see that, rather than working together so everyone can profit more, greed and distrust cause businesses to operate less than optimally.
- I don't know why but I'm addicted to this game. I find joy in it.

- Interesting
- I have finally figured this game out! The whole key is to not panic. I am loving this game, and I am always looking through my e-mail to see if you all have brought a new one out. Keep up the good work!
- Compared to the previous round, there were fewer wild fluctuations in order amounts this time, so it was easier to manage the inventory levels, and so I chose not to purchase insurance.
- I'm ashamed of that one x_x I didn't realize insurance would carry with you, I would have gotten it much earlier.
- Thanks for the HITs! These games are a lot of fun.
- I chose not to get insurance initially. Then the company refused to give me a price break on a policy for a term shorter than 52 weeks, so I said Fooey and became self-insured (=played without information, or wrongdoing, or insurance). But I might have played around more if the insurer had offered a 2- or 3- month term policy. ;)
- I like the idea of paying for insurance to see information. It is a good balance to the game.
- I do think it was easier with being able to see the other players' supply and demand, but that insurance really sets you back at the beginning. The wholesaler in particular in my game had a hard time getting out of the red. I wonder if a weekly pay out for the insurance might be feasible.
- this is getting interesting
- Paying the insurance seems worth it to have access to the information. I plan on trying it again without paying for it and see what kind of score I can get. 10k for a year is significant when I only made 52k for the year.
- Insurance is the best 11k I could've spent here. There were enough fluctuations in demand in the chain–not many, but enough–that not having it and not knowing what

was coming would've cost me at least half–if not more–of what we made for the entire supply chain this time.

- Would it be wrong to ask what the bottom line would be to get cutoff from playing more games?
- Interesting twist! Don't think the insurance was worth it though.
- I honestly with that I had more free time to play this game. It's kind of addicting. I liked that the price of insurance changed over time and with each game.
- this game is a lot of fun- I really enjoy it.
- I liked the new twist to the game. It was fun.
- This is so much fun!
- I took a chance and did not take the insurance and I paid for it. I got in a hole and worked my way out of it in the end, but that put me behind. I had a much better second half than first. I should have taken the insurance.
- I love your games, I have a blast every time:)
- Still love this game!! I sometimes think that the computer players make bad decisions when it comes to backlogs and orders.
- Very fun, keep them coming. :)
- Cool!

D.5 #321 Insurance, Steady, Distributor (Second Trial)

- interesting game!
- I accidentally pressed 166 instead of 16 on one of my orders in the first one . I played again, and that's my new completion code.

- Thanks!
- That was fun and interesting. Thanks for inviting me back to play! As a bar manager, my bosses will be happy to know I turned a profit here.
- Thank you for allowing me to provide your data. This is challenging!
- I love this game. I'd love to play other versions that you guy come up with!
- Always enjoyable and loving the challenge
- I love this game, however I did find it challenging to find a balance in this one after a rough start. Wish I could replay!
- I missed your hits! Thank you for bringing them back :D
- Thanks for letting me do your hits again! I always looked forward to your hits in the past!
- Interesting
- I did terrible this round. I kept thinking my backlog was my in stock for several rounds and I confused myself as to why I wasn't shipping. Ugh, Still a fun game tho. Thanks for the chance to play again.
- I'm glad this game came back. It's a great learning experience in regards to how the supply chain works and I enjoy learning about these sorts of things. I feel like I'm getting better and improving at it as well.
- Fun, missed this game!
- Fun game as always!
- I still enjoy this game.
- Cool HIT!
- Fun as usual, I did not tank at the end like I did last time.

- fun game
- I thought the goal before was to maximize the revenue for the whole supply chain, but the charts showed the final cash balance for just the players. Which is the actual goal? Because there are some decisions I might have made differently if I were just trying to maximize my own profits. For instance, I tried REALLY hard to avoid having a backlog, so that any mistakes I made wouldn't also hurt the wholesaler and retailer. I might have bought fewer crates in order to avoid having so much extra stock, if I were just trying to help myself.
- Thanks!
- The insurance payout was a bit of a surprise. Overall, I think being able to know what the others in the supply chain are doing is very helpful, but if the abuse of information hurt me, I might not think it was worth it.
- much easier to estimate overall demand if insurance is purchased.

D.6 #421 Share & Keep Money / Share & Require Policy, Steady, Distributor

- I enjoy working on this study. Thank you for allowing me to provide your data.
- Fun. Thanks!
- I enjoyed the game. Thanks a bunch for inviting me back to play it again!
- Really enjoy these, thanks!
- This is literally my favorite HIT by far. I love that I'm playing a game and using my math and strategy skills and it's fun!
- I did a lot better this round. I will continue to do these as long as you let me.

- I did so horribly I am embarrassed. I was looking at the backlog wrong and confusing it with overstock for a minute, then tried to overcompensate to make it right. And of course once I doubled up on stock the orders dropped to nothing. As always though, it was a lot of fun. I just wish I'd recovered from my mistake with less panic :)
- Great game!
- Love playing this game! Highlight of Turking!
- was fun and challenging again! thanks!
- Ive done this several times now and every time has been a new experience. This time I elected not to see everyone else and just concentrated on mine and I think it was a wise decision. I think I did pretty well this time better than last time for sure.
- always interesting to do these
- Interesting
- challenging with the losses, but a lot of fun! Thank you for this opportunity.
- Cool game!
- There were a few points where I thought it froze, but then it continued after about 30-60s delay. Not sure what was going on there, but figured it was worth mentioning :D
- Tsk Tsk Mr. Wholesaler, trying to leverage that information. Would have been nice to get some money from insurance though.
- It looks like I actually improved slightly. This games just gets more interesting every time :)

D.7 #521 Deny / Share & Require Policy, Steady, Distributor

- Fun game, keep the HITS coming. :) Thank you!
- Fun! Thanks!
- this one was a lot harder... a lot of fluctuations.
- I utterly failed this time because I was reading the orders wrong for the first half of the game. It was rather frustrating mistake on my part, sorry. I love this game and normally do better.
- Thanks for inviting me back to play this game!
- Love this game :D This is so much fun XD thank you!
- Lots of fun! Love this game!
- I understand why someone might deny other chain participants their information, but it just seems like if you have that opportunity, it's just good business to let the others see the information so that everyone is aware of everyone else's supply and demand. Abusing it just seems foolish.
- Thanks for the bonus and invite to the new game! I did a bit better this time.
- ooo. I did horrible this time. I won't try a new strategy. I'll stick to what I was doing before hand.
- darn, didnt do so well.
- Interesting
- Thank you for the opportunity to provide your data. :-)
- Once again I love this game. I dont think I did as well as last time, but I think I did pretty good overall. I had some overages during the first few weeks, but got them under control later.

- Stupid retailer. Tsk Tsk. Also, I meant to order 12 not 21 in week 9, kinda threw my whole game off. Fingers slipped. I think I did alright to pick the pieces back up. I wonder how often this happens in actual business settings...
- this one was pretty hard
- Love this game! (Although I sucked at it this time)
- Cool! Looking forward to more games!

References

[1] `http://www.forbes.com/sites/investopedia/2011/11/08/strange-insurance-policies-from-across-the-globe`.

[2] Amazon Mechanical Turk, https://www.mturk.com/mturk/welcome.

[3] Author's GitHub Repository, https://github.com/naokitnk.

[4] Bcrypt, https://www.npmjs.org/package/bcrypt.

[5] Beer Game Basic Training, https://www.youtube.com/watch?v=ruQC3fQNWT0.

[6] BeerGame by MA-system, http://www.masystem.com/o.o.i.s/1365.

[7] Boost C++ Libraries, http://www.boost.org.

[8] Bullwhips and Beer: Why Supply Chain Management is so Difficult, http://forio.com/blog/bullwhips-and-beer.

[9] Connect Flash, https://www.npmjs.org/package/connect-flash.

[10] Connect, http://www.senchalabs.org/connect/.

[11] Connect Roles, https://github.com/ForbesLindesay/connect-roles.

[12] CrowdFlower by MA-system, http://www.crowdflower.com.

[13] D3.js, http://d3js.org.

[14] ETH Zurich The Beer Distribution Game, http://www.beergame.lim.ethz.ch.

[15] Express, http://expressjs.com.

[16] FORESTIA - A Simulation Game on Sustainable Forest Management, http://www.gameforscience.com/forestia/.

[17] IASimulator, https://github.com/naokitnk/IASimulator.

[18] Jade, http://jade-lang.com.

[19] jQuery, http://jquery.com.

[20] jQuery UI, http://jqueryui.com.

[21] jStat, https://github.com/jstat/jstat.

[22] MA-system BeerGame, http://beergame.pipechain.com.

[23] MongoDB, http://www.mongodb.org.

[24] Mongoose, http://mongoosejs.com.

[25] Nginx, http://nginx.org.

[26] Node.js, http://nodejs.org.

[27] Pasport, http://passportjs.org.

[28] Passport-local, https://github.com/jaredhanson/passport-local.

[29] Reference for Business: Encyclopedia of Business, 2nd ed.: Virtual Organizations, http://www.referenceforbusiness.com/management/Tr-Z/Virtual-Organizations.html.

[30] Run The Models Beer Distribution Game Simulation Model, http://www.runthemodel.com/models2/507/.

[31] Scalable Vector Graphics, http://www.w3.org/Graphics/SVG/.

[32] Simunomics - Massive Multiplayer Business Simulation Game, http://www.simunomics.com/Login.php.

[33] Socket.IO, http://socket.io.

[34] Virtonomics, http://virtonomics.com.

[35] Anonymous. Horizontal Integration: Broader Access Models for Realizing Information Dominance. Technical Report JSR-04-132, MITRE Corporation JASON Program Office, December 2004.

[36] P. P. Boyle and J. Mao. An Exact Solution for the Optimal Stop Loss Limit. *The Journal of Risk and Insurance*, 50(4):719–726, Dec. 1983.

[37] L. L. Cam. Maximum Likelihood: An Introduction. *International Statistical Review / Revue Internationale de Statistique*, 58(2):153–171, Aug. 1990.

[38] P.-C. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger. Fuzzy MLS: An Experiment on Quantified Risk-Adaptive Access Control. In *IEEE Symposium on Security and Privacy*, 2007.

[39] P.-C. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger. Fuzzy Multi-Level Security: An Experiment on Quantified Risk-Adaptive Access Control. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 222–230, May 2007.

[40] Y. Crama and M. Schyns. Simulated annealing for complex portfolio selection problems. *European Journal of Operational Research*, 150(3):546–571, 2003.

[41] R. Croson and K. Donohue. Behavioral Causes of the Bullwhip Effect and the Observed Value of Inventory Information. *Management Science*, 11(3):323–336, Mar. 2006.

[42] R. de Souza. Supply chain dynamics and optimization. *Integrated manufacturing systems*, 11(5):348–364, 2000.

[43] N. Dimmock, A. Belokosztolszki, D. Eyers, J. Bacon, and K. Moody. Using trust and risk in role-based access control policies. In *Proceedings of the Symposium on Access Control Models and Technologies*, pages 156–162, 2004.

[44] N. Dimmock, A. Belokosztolszki, D. Eyers, J. Bacon, and K. Moody. Using trust and risk in role-based access control policies. In *Proceedings of the ACM Symposium on Access Control Models and Technologies*, 2004.

[45] I. Dinov. Expectation maximization and mixture modeling tutorial. *Statistics Online Computational Resource*, 2008.

[46] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, pages 39–43, Oct. 1995.

[47] D. Ferraiolo and R. Kuhn. Role-based access controls. In *NIST-NCSC National Computer Security Conference*, pages 554–563, October 1992.

[48] H. U. Gerber, E. S. W. Shiu, and N. Smith. Methods for estimating the optimal dividend barrier and the probability of ruin. *Insurance: Mathematics and Economics*, 42(1):243–254, Feb 2008.

[49] A. Ghosh and A. Roth. Selling Privacy at Auction. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, pages 199–208, 2011.

[50] M. Gilli and E. Këllezi. An Application of Extreme Value Theory for Measuring Financial Risk. *Computational Economics*, 27(2-3):207–228, May 2006.

[51] C. Haehling von Lanzenauer and K. Pilz-Glombik. Coordinating Supply Chain Decisions: An Optimization Model. *OR Spectrum*, 24(1):59–78, Feb. 2002.

[52] B. Hajek. Cooling schedules for optimal annealing. *Mathematics of Operations Research*, 13(2):311–329, May 1988.

[53] A. C. Harvey and C. R. McKenzie. Algorithm AS182. An algorithm for finite sample prediction from ARIMA processes. *Applied Statistics*, 31:180–187, 1982.

[54] K. Hoo. How much is enough? A risk-management approach to computer security. Working paper, Center for International Security and Cooperation, June 2000.

[55] G. Q. Huang, J. S. K. Lau, and K. L. Mak. The impacts of sharing production information on supply chain dynamics: a review of the literature. *International Journal of Production Research*, 41(7):1483–1517, May 2003.

[56] IBM. ILOG CPLEX 12.2. `http://www-01.ibm.com/software/integration/optimization/cplex-optimizer`.

[57] R. Kaas, M. Goovaerts, J. Dhaene, and M. Denuit. *Modern Actuarial Risk Theory Using R.* Springer, second edition, 2008.

[58] D. Kahneman. *Thinking, Fast and Slow.* Farrar, Straus and Giroux, first edition, 2011.

[59] D. Kahneman and A. Tversky. Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, 47(2):263–291, Mar. 1979.

[60] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4, Nov. 1995.

[61] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science, New Series*, 220:671–680, 1983.

[62] G. Knolmayer, R. Schmidt, and S. Rihs. *Teaching supply chain dynamics beyond the beer game*. Institut f{ü}r Wirtschaftsinformatik der Universit{ä}t Bern, 2007.

[63] G. Lebanon, M. Scannapieco, M. Fouad, and E. Bertino. Beyond k-Anonymity: A Decision Theoretic Framework for Assessing Privacy Risk. In *Privacy in Statistical Databases*, volume 4302 of *Lecture Notes in Computer Science*, pages 217–232. Springer Berlin / Heidelberg, 2006.

[64] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. M. VanBriesen, and N. S. Glance. Cost-effective outbreak detection in networks. In *KDD*, pages 420–429, 2007.

[65] H. Liu, E. Howley, and J. Duggan. Optimisation of the Beer Distribution Game with complex customer demand patterns. In *Evolutionary Computation, 2009. CEC '09. IEEE Congress on*, pages 2638–2645, May 2009.

[66] H. M. Markowitz. Portfolio Selection. *Journal of Finance*, 7:77–91, 1952.

[67] H. M. Markowitz. *Portfolio Selection: Efficient Diversification of Investments*. John Wiley & Sons, 1959.

[68] G. B. Mathews. On the partition of numbers. *Proceedings of the London Mathematical Society*, 28:486–490, 1897.

[69] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs (3rd Ed.)*. Springer-Verlag, London, UK, UK, 1996.

[70] R. O. Michaud. The Markowitz optimization enigma: Is 'optimized' optimal? *Financial Analysts Journal*, January-February 1989.

[71] I. Molloy, P.-C. Cheng, and P. Rohatgi. Trading in risk: using markets to improve access control. In *Proceedings of the New Security Paradigms Workshop*, 2009.

[72] E. Mosekilde and E. R. Larsen. Deterministic chaos in the beer production-distribution model. *System Dynamics Review*, 4(1-2):131–147, 1988.

[73] N. Nissanke. Risk based security analysis of permissions in RBAC. In *Proceedings of the 2nd International Workshop on Security in Information Systems*, pages 332–341, 2004.

[74] A. F. Perold. Large-scale portfolio optimization. *Management Sci.*, 30:1143–1160, 1984.

[75] J. W. Pratt. Risk Aversion in the Small and in the Large. *Econometrica*, 32:122–136, 1964.

[76] C. Riederer, V. Erramilli, and A. Chaintreau. For Sale : Your Data By : You. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, pages 1–6, 2011.

[77] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-Based Access Control Models. *IEEE Computer*, 29(2):38–47, 1996.

[78] M. Sasena, P. Papalambros, and P. Goovaerts. Global optimization of problems with disconnected feasible regions via surrogate modeling. In *AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002.

[79] M. Solomon and M. Chapple. *Information Security Illuminated*. Jones, and Bartlett Publishers, 2005.

[80] J. D. Sterman. Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment. *Management Science*, 35(3):321–339, Mar. 1989.

[81] F. Strozzi, J. Bosch, and J. M. Zaldívar. Beer game order policy optimization under changing customer demand. *Decision Support Systems*, 42(4):2153–2163, 2007.

[82] N. Tanaka, M. Winslett, A. J. Lee, D. K. Y. Yau, and F. Bao. Insured Access: An Approach to Ad-hoc Information Sharing for Virtual Organizations. In *Proceedings of the third ACM Conference on Data and Application Security and Privacy*, pages 301–308, Feb. 2013.

[83] J. S. Thomsen. Hyperchaotic phenomena in dynamic decision making. *Systems analysis modelling simulation*, 9(2):137–156, 1992.

[84] E. Tong, C. Mues, and L. Thomas. A zero-adjusted gamma model for estimating loss given default on residential mortgage loans. In *Proceedings of the Credit Scoring and Credit Control XII Conference*, pages 24–26, Aug. 2011.

[85] B. Travica. The design of the virtual organization: A research model. In *Proceedings of the Americas Conference on Information Systems. August*, pages 15–17, 1997.

[86] A. Tsanakas and E. Desli. Risk measures and theories of choice. *British Actuarial Journal*, 9(4):959–991, 2003.

[87] A. Tsanakas and E. Desli. Measurement and Pricing of Risk in Insurance Markets. *Risk Analysis*, 25(6):1653–1668, 2005.

[88] L. Yan and T. G. Woo. Information sharing in a supply chain with dynamic consumer demand pattern. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 10 pp.–, Jan. 2004.

[89] L. Zhang, A. Brodsky, and S. Jajodia. Toward information sharing: benefit and risk access control (BARAC). In *Proceedings of the 7th IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 45–53, 2006.