

10-16-2015

Optimal Portfolio Strategies for New Product Development

Emre Nadar
Bilkent University

Mustafa Akan
Carnegie Mellon University, akan@cmu.edu

Laurens Debo
University of Chicago

Alan Scheller-Wolf
Carnegie Mellon University, awolf@andrew.cmu.edu

Follow this and additional works at: <http://repository.cmu.edu/tepper>

 Part of the [Business Commons](#)

Optimal Portfolio Strategies for New Product Development

Emre Nadar

Department of Industrial Engineering, Bilkent University, 06800 Ankara, Turkey, emre.nadar@bilkent.edu.tr

Mustafa Akan

Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA 15213, akan@andrew.cmu.edu

Laurens Debo

Tuck School of Business, Dartmouth College, Hanover, NH 03755, Laurens.G.Debo@tuck.dartmouth.edu

Alan Scheller-Wolf

Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA 15213, awolf@andrew.cmu.edu

We study the portfolio selection problem in a new product development setting with many projects in parallel, each lasting several stages, in the face of uncertainty. Each stage of the process performs an experiment on a selected number of projects in the stage, depending on the amount of (scarce) budget allocated to the stage. Projects become differentiated through their experimental results, and all available results for a project determine its *category*.

We model the problem as a Markov decision process. We seek an optimal policy that specifies, for every configuration of projects in categories, which projects to test and/or terminate. For two special cases we characterize the optimal project promotion policy as following a *new* type of strategy, *state-dependent non-congestive promotion* (SDNCP). SDNCP implies that a project with the highest expected reward in any stage is advanced to the next stage if and only if the number of projects in each successor category is below a congestion-dependent threshold. For the general problem, numerical experiments reveal the outstanding performance of SDNCP (optimal in 72 of 77 instances with maximum deviation from optimal of 0.67%), highlighting when and how a *fixed* non-congestive promotion policy, which is easier to implement, may fall short.

Key words: New product development; Markov decision processes; Bayesian updating; optimal control

1. Introduction

Developing new products and swiftly launching them successfully into the market have become critical in today's competitive environment. For this reason firms often pursue many new product development (NPD) projects in parallel (e.g., Ding and Eliashberg 2002, Ulrich and Eppinger 2003, and Girotra et al. 2007). But this can be a very challenging task from the *operations management* perspective: Projects compete for access to scarce resources in general, in the absence of precise knowledge regarding their outcomes (Kavadias and Chao 2008). Only through experimentation can uncertainty be reduced in many settings: A typical NPD process consists of a series of distinct experimental stages allowing information generated in each stage to be incorporated into portfolio decisions in later stages (Cooper 2008 and Artmann 2009). Firms terminate product candidates with little promise before testing them in expensive downstream stages, focusing available resources

on much stronger candidates (Thomke 2008).

Information gathered as a project progresses through experimental stages is typically imperfect, with potentially varying levels of accuracy across stages (e.g., Gino and Pisano 2005). In addition, projects may face uncertainty about not only their outcomes but also their experimentation times (e.g., Loch and Terwiesch 1999, and Sommer et al. 2008). Another significant complication is that each stage may require a unique specialized resource, may have a fixed budget, or may share a fixed budget with other stages (e.g., Blau et al. 2004). Taken together, these elements require concurrent capacitated resource allocation decisions, across stages, under uncertainty.

Improperly managing such constrained resources may lead to significant delays in project completion times, reducing the firm's profitability (Adler et al. 1995). One practical solution to prevent such delays is to limit the number of projects in each stage (Thomke and Reinertsen 2012), but it is not clear how to best implement such a policy or how effective it would be. Part of the problem is that, to our knowledge, the NPD portfolio selection literature has not yet developed a comprehensive modeling framework that explicitly captures all these aspects of the problem. Our paper is the first attempt to fill this gap: By employing the theory of Markov decision processes (MDPs), we study the NPD portfolio selection problem under imperfect information across stages, random experimentation times, and stage-dependent resources.

Dynamic portfolio management has received much attention in the NPD literature; we refer the reader to Kavadias and Loch (2004) and Kavadias and Chao (2008) for comprehensive reviews. Many authors have viewed the project prioritization problem as a multi-armed bandit (MAB) problem in which projects compete for a critical resource that can be utilized by only one project at a time. A project utilizing the resource undergoes Markovian transitions and returns an immediate state- and time-dependent reward. Gittins and Jones (1972) introduced the *Gittins Index*, a number that can be assigned to each project at any time that corresponds to the reward that would make the controller indifferent between working on the project and terminating it in exchange for that reward: It is always optimal to work on the project with the highest Gittins index. See Gittins (1979, 1989), Whittle (1980, 1981, 1988), Ross (1982), Weber (1992), Banks and Sundaram (1994), Bertsimas and Niño-Mora (1996), Kavadias and Loch (2003), and Bertsekas (2007) for further analysis of the MAB problem. In variants of the MAB problem, multiple resources are dynamically allocated to several projects, new projects may arrive, all projects may change state after utilization of a resource, and a common fixed changeover cost and/or delays may be incurred by reallocation

of resources. But, unlike the MAB problem, in our study projects use different (scarce) resources in different stages of the NPD process, leading to concurrent resource allocation decisions across stages.

The project initiation and prioritization problems can also be viewed through the lens of queueing theory: The problem of when to initiate a project by testing a new product idea is similar to the admission control problem in which different customers (or projects) may bring different rewards if accepted. As the number of customers in the system nears the capacity, the controller tends to reject customers with smaller rewards in anticipation of future customers with larger rewards. See Miller (1969), Lippman and Ross (1971), Stidham (1985), Haviv and Puterman (1998), Lewis et al. (1999), and Lewis and Puterman (2000) for analysis of queues with admission control. Similarly, the project prioritization problem can be modeled as a multiclass queue with stochastic completion times for each class of job (or project). The *cμ rule* is optimal for linear delay costs: It is always optimal to give priority to the job with the highest delay cost divided by the expected processing time. For non-linear delay costs, the *generalized cμ rule* is asymptotically optimal in heavy traffic and in due date scheduling. See Harrison (1975), Wein (1992), Ha (1997a), van Mieghem (1995, 2000), and Mandelbaum and Stolyar (2004) for analysis of multiclass queues. Unlike the above papers, our study takes into account the evolution of projects through experimental results in project promotion, allowing projects to be delayed or even terminated based on this evolution.

Another stream of research has studied the dynamic and stochastic knapsack problem: Each project arrives in time as a stochastic process and has a demand for a limited resource. The demands and their rewards are random, and become known upon arrival. If a demand is accepted, the reward is received, otherwise a penalty is paid. The objective is to maximize the expected reward in the finite or infinite horizon. See Papastavrou et al. (1996), and Kleywegt and Papastavrou (1998, 2001). These papers again fail to incorporate intermediate project reviews into decision-making.

Several other authors have incorporated learning into the NPD process, allowing for information updating: Roberts and Weitzman (1981) consider a single project that must pass through a number of experimental stages, each incurring a random cost. The reward is received after all stages are complete. The *state* of the project is described by the number of stages from the end. The reward is a state-dependent random variable, which becomes less variable as the state decreases (and thus more information becomes available). The authors establish the optimal stopping rule when the state variable is continuous, the reward is normal, and the standard deviation of the reward in any

stage is proportional to the expected cost of carrying the project through to completion from that stage. The authors consider neither a multi-project environment under scarce resources nor random experimentation times. Vishwanath (1992) considers a finite number of projects, each yielding an unknown reward at an uncertain time. Each project has its own independent joint probability distribution for the reward and the time to collect the reward (if selected). Several projects may be undertaken in parallel, and the projects may be selected sequentially in any order desired. The *highest expected utility* (assessed for each project in isolation) rule is the best order for parallel project selection to maximize the total expected discounted utility. Although Vishwanath (1992) allows for information acquisition as a selected project proceeds, resource scarcity is ignored.

Finally, Adler et al. (1995) model the NPD process as a queueing network: Resources are “workstations,” each consisting of one or more identical servers in parallel. And projects are “jobs,” each consisting of several activities to be performed by specified workstations in specified orders. The authors develop a simulation model for their queueing network under exponentially distributed activity times. Their simulation experiments, using the data from real projects in a real organization, indicate that restricting the total number of projects under way at any time may greatly reduce project completion times. But this comes at a cost: Reduced throughput. Adler et al. (1995) thus argue that limiting the number of projects in the network would be particularly attractive if projects can be selected according to their probabilities of success. (Projects are *not* dynamically prioritized during execution in their model.) In this paper we develop a mathematical abstraction that rigorously incorporates projects’ probabilities of being in any state (based on the latest information) into project promotion decisions. This enables us to show that a *congestion-dependent* critical level on the number of projects of each type proves more effective than a *fixed* critical level on the total number of projects in the entire system. See also Blau et al. (2000), Blau et al. (2004), Reppenning (2001), Subramanian (2003), and Verma et al. (2011) for simulation-based methods.

Specifically, we model the portfolio selection problem as an infinite-horizon MDP under the total expected discounted reward criterion: An ample source of new product ideas is available. Each project can be initiated by testing a new product idea, and undergoes a different experiment in each stage of the NPD process. Experiments generate signals about the true nature of a project – the quality of these signals can differ across stages. Beliefs about the true nature of a project are updated after each observed signal according to a Bayesian rule. Projects thus become differentiated through their signals; all available signals for a project determine its *category*. Experimentation

times in any stage are independent and exponentially distributed with finite rate, due to constrained resources. Each experiment incurs a fixed cost upon completion; both experimentation rates and costs can also differ across stages. Returns of a project are earned only after the resulting product is launched into the market. But some projects with little promise may not pay off due to high experimentation and/or delay costs; thus projects in any stage may also be terminated at no cost.

The state of the above system consists of the numbers of projects in each category. Given the system state, a control policy specifies whether or not to test a project in each stage, and which project to select if a project is to be tested (i.e., project promotion); and whether or not to terminate a project, and which project to select if a project is to be terminated (i.e., project termination).

We characterize the structure of the optimal stationary *promotion* policy via a *new* type of policy, state-dependent non-congestive promotion (SDNCP) in two special cases of the general problem: (a) when there is a single experimental stage and projects are never terminated (Theorem 1), or (b) when there are multiple mandatory stages that do not provide information (Theorem 2). We also prove that projects are never terminated at optimality in case (b). An SDNCP policy implies that, in each stage, it is optimal to advance a project with the highest expected reward to the next stage if and only if the number of projects in each successor category is less than a congestion-dependent threshold. Thus such a policy is “forward looking,” i.e., taking into account downstream congestion. We show that threshold values weakly decrease as a later stage becomes more congested or as an earlier stage becomes less congested. Thus the SDNCP policy is also “backward looking,” i.e., considering upstream congestion. Importantly, a stage becomes more congested with an increase in the number of projects, but *also* in the expected reward of any project, in the stage.

We adapt the SDNCP policy to our general model, evaluating its use as a *heuristic* project promotion policy. We compare the SDNCP policy to two other heuristics: a fixed non-congestive promotion (FNCP) policy with fixed thresholds across stages, and a continuous promotion (CP) policy. The FNCP policy limits the number of active projects in each stage, which is one of those proposed solutions in Thomke and Reinertsen (2012). The CP policy initiates a project only when another project has been completed or terminated; such a policy is equivalent to the “pull” strategy in Adler et al. (1995).

Because our optimal structural results provide no insight into project *termination* decisions in the general problem, we do *not* impose any policy structure for project termination in our heuristics. Taking the average reward rate as our optimization criterion, we thus formulate linear programs to

find the globally optimal policy and the optimal CP policy, and mixed integer programs to find the optimal SDNCP and FNCP policies. (The optimal CP policy always promotes projects if feasible, which leads to the linear program.) As we impose a deterministic policy structure for project *promotion* in each of our heuristics, the optimal SDNCP, FNCP, and CP policies can potentially yield fewer randomized decisions, in comparison with the globally optimal policy, making their implementation relatively simple.

We analytically show that SDNCP outperforms FNCP and CP with respect to objective value (Proposition 1). We then generate 77 instances of the general NPD problem: Remarkably, we find that SDNCP yields the globally optimal reward in 72 of these instances and its maximum distance from optimal reward is only 0.67% in the other 5 instances. We also find that SDNCP performs better than FNCP and CP by up to 7.98% and 64.61% of the globally optimal reward, respectively, on the same test bed. The average distances of FNCP and CP from optimal reward are 1.62% and 6.74%, respectively. Numerical results indicate that SDNCP has the greatest benefit over FNCP and CP (i) when downstream stages are slower than upstream stages, (ii) when project holding and/or experimentation costs are higher, or (iii) when experimental results are less accurate. However, the computation times of FNCP and CP are several orders of magnitude lower than those of SDNCP. Thus the controller might prefer to use FNCP if none of conditions (i)-(iii) holds.

We contribute to the NPD literature in several important ways: First, we develop a novel MDP formulation for the portfolio selection problem: Managing scarce resources effectively under uncertainty is rather difficult in an NPD process. Our MDP model enables a clean analytical formulation to handle such complexity, allowing us to generate structural insights into optimal policies. Second, we show that optimal project promotion decisions are congestion-dependent, and that “congestion” is driven by not only the number of projects but also the breakdown of projects by stage and expected reward. Last, we prove that common heuristics limiting the number of projects in the entire system or in each stage are inferior to our SDNCP policy in the general problem. Numerical experiments reveal when FNCP and CP significantly deviate from the optimal policy, producing high-level guidelines for control policy choices in different environments.

The rest of the paper is organized as follows: Section 2 formulates our general NPD model under the discounted reward criterion. Section 3 establishes the optimal policy structure when there is a single experimental stage and projects are never terminated. Section 4 establishes the optimal policy structure when there are multiple non-experimental stages. Section 5 presents our

heuristic policies for the general model in Section 2. Section 6 presents our numerical results for the heuristics under the average reward criterion. Section 7 offers a summary and concludes. Proofs of all analytical results (except Proposition 1) are contained in an online appendix.

2. Problem Formulation

We consider the problem of project selection and resource allocation in a continuous-time NPD process (e.g., a new drug development process, cf. Figure 1). Each NPD project passes through a finite number of experimental stages (e.g., safety, efficacy, and general tests; see DiMasi et al. 2003 and Girotra et al. 2007) before the resulting product (e.g., “Crestor”) is placed on the market. Define $\mathcal{M} = \{1, 2, \dots, m\}$ as the set of experimental stages, and i as the index for the stage (e.g., $m = 3$ in Figure 1). The true ultimate *nature* of a project falls into one of a number of states (e.g., “success” or “failure”), and initial expectations about the *nature* are the same across all projects. Each experiment generates a piece of new information (e.g., “good” or “bad” signal) about the *nature* of the project; uncertainty pertaining to the ultimate outcome of the project is further resolved in each stage. Define K as the number of possible signals that can be generated in each stage for each project, and k as the index for the signal (e.g., $K = 2$). There exists a one-to-one correspondence between the set of signals in each stage and the set of states for the true *nature*. Both sets consist of integers from 1 to K such that a lower integer indicates a project with higher return (e.g., $k = 1$ means a “good” signal that corresponds to a “success”, and $k = 2$ means a “bad” signal that corresponds to a “failure”).

All available signals for a project determine its *category*; projects become differentiated through their *categories*. Define $\mathcal{N} = \{0, 1, \dots, n\}$ as the set of project categories, and j as the index for the category. Note that K^i is the number of categories in stage i , and $n = K + K^2 + \dots + K^m$ (e.g., $n = 14$ in Figure 1). Different stages of the NPD process utilize different resources (e.g., specialized testing equipment or specialists with unique areas of expertise), and all these resources are limited. Define \mathcal{W}_i as the set of project categories waiting for access to resources of stage i for experimentation (e.g., $\mathcal{W}_1 = \{0\}$, $\mathcal{W}_2 = \{1, 2\}$, and $\mathcal{W}_3 = \{3, 4, 5, 6\}$). Also, define \mathcal{W}_{m+1} as the set of project categories that have completed all stages except the product launch stage $m + 1$ (e.g., $\mathcal{W}_4 = \{7, 8, \dots, 14\}$).

Experimental results *imperfectly* reveal the true nature of a project throughout the NPD process. Define $\Phi^{(i)}$ as an $K \times K$ conditional probability matrix in stage i , with the rows and the columns indicating the states of the true nature and the signals of the test, respectively: $\phi_{k,k'}^{(i)}$ is

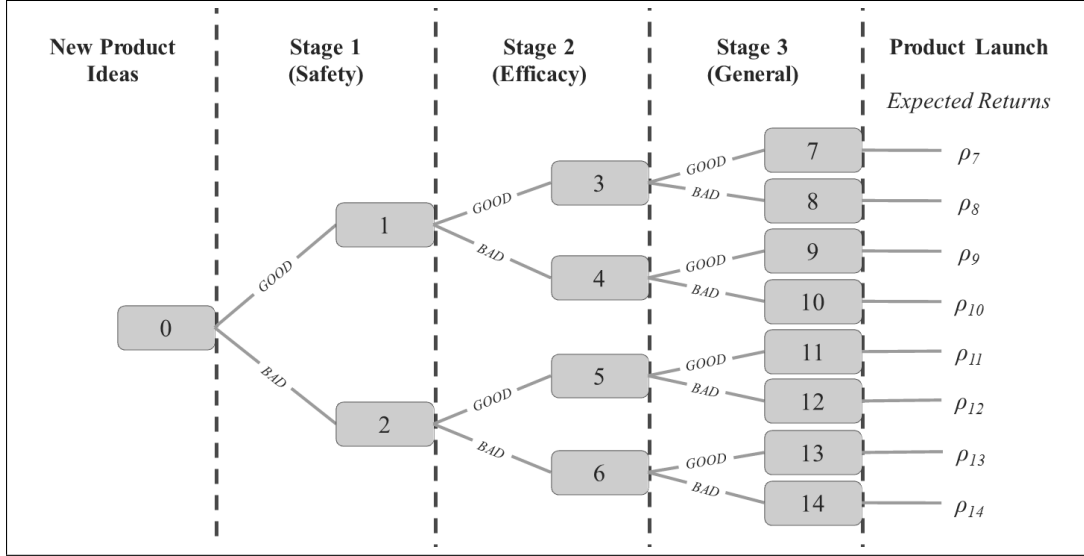


Figure 1 A new drug development process. Each node represents a different project category, and each arc in a given stage represents a different signal.

the probability that the experiment in stage i generates signal k' for projects with true nature k . Note that if $\phi_{k,k}^{(i)} = 1, \forall k$, then the experiment in stage i perfectly reveals the true nature. Thus we assume $\phi_{k,k}^{(i)} < 1, \forall i, k$. Also, note that we may have $\phi_{k,k'}^{(i)} = 0$ for some nature k and some signal k' in some stage i . Thus the total number of signals that can be actually observed can differ across states of the true nature and stages. Beliefs about the true nature of a project undergo Bayesian updating after each experiment. Define $\mathbf{p}_j = (p_{j,1}, \dots, p_{j,K})$ as the probability distribution for the true nature of a project in category $j \in \mathcal{W}_i$. Suppose that a project in category $j \in \mathcal{W}_i$ becomes category $j' \in \mathcal{W}_{i+1}$, returning signal k' in stage i . Then the posterior probability distribution for the true nature of the project $\mathbf{p}_{j'} = (p_{j',1}, \dots, p_{j',K})$ is calculated as

$$p_{j',k} = \frac{p_{j,k} \times \phi_{k,k'}^{(i)}}{f_{j \rightarrow j'}}, \quad \forall k,$$

where $f_{j \rightarrow j'}$ is the probability that a project in category $j \in \mathcal{W}_i$ falls into category $j' \in \mathcal{W}_{i+1}$, returning signal k' in stage i , i.e.,

$$f_{j \rightarrow j'} = \sum_{1 \leq k \leq K} p_{j,k} \times \phi_{k,k'}^{(i)}.$$

The state of the system at time t is the vector $\mathbf{X}(t) = (X_1(t), \dots, X_n(t))$, where $X_j(t)$ is a nonnegative integer denoting the number of projects in category j at time t . Projects held in the NPD process incur a holding cost per unit time which is convex and strictly increasing in the total number of projects. Denote by $h(\mathbf{X}(t)) = h(\sum_j X_j(t))$ the holding cost rate at state $\mathbf{X}(t)$. Also, denote

by r_k the reward that can be obtained from the launch of a new product with ultimate outcome k . The expected reward for a project in category j is calculated by $\rho_j = \mathbf{r} \cdot \mathbf{p}_j$, where $\mathbf{r} = (r_1, \dots, r_K)$ is a K dimensional nonnegative vector whose elements are in descending order. Thus $\rho_j \geq 0, \forall j$. The returns of a project are earned only after the project is complete and the resulting new product is launched.

The scarce resources in any stage can be utilized by at most one project at any time. This assumption is benign when there is *diseconomies* of scope in resource sharing. Both experimentation and product launch times are independent and exponentially distributed. Define μ_i as the experimentation rate in stage i , and μ_{m+1} as the product launch rate. The system incurs an experimentation cost c_i upon completion of an experiment, but there are no costs associated with interrupted experiments. This assumption is not restrictive in the two special cases of the model introduced in Sections 3 and 4, as it is never optimal to interrupt any experiment once it has been initiated in those cases. Projects may also be terminated, only one at a time, at no cost. Termination time for a project is again exponentially distributed with finite mean $1/\lambda$ (which may be arbitrarily small). Although $\rho_j \geq 0, \forall j$, note that it might be desirable to terminate a project due to high experimentation and holding costs, outweighing its expected reward.

Since all inter-event times are exponentially distributed, the system retains no memory, and decision epochs can be restricted to times when the state changes. Using the memoryless property, we can formulate the problem as an MDP and focus on Markovian policies for which actions at each decision epoch depend solely on the current state. A control policy ℓ specifies for each state $\mathbf{x} = (x_1, \dots, x_n)$, the action $\mathbf{u}^\ell(\mathbf{x}) = (y_0, \dots, y_n, z)$, $y_j \in \{0, 1\}, \forall j$, and $z \in \{0, 1, \dots, n\}$, where $y_j = 1$ means advance a project from category j to the next stage, $y_j = 0$ means do not advance a project from category j , $z > 0$ denotes the category from which a project is terminated, and $z = 0$ means do not terminate any project. Denote by $\mathcal{U}(\mathbf{x})$ the set of admissible actions in state \mathbf{x} . The action $\mathbf{u}^\ell(\mathbf{x}) = (y_0, \dots, y_n, z) \in \mathcal{U}(\mathbf{x})$ must satisfy the following conditions:

- A project in any category can be tested or terminated only if at least one project exists in this category, i.e., $y_j = 0$ and $z \neq j$ if $x_j = 0, \forall j > 0$.
- At most one project can be tested at any time in any stage, i.e., $\sum_{j \in \mathcal{W}_i} y_j \leq 1, \forall i$.
- At most one new product can be introduced at any time, i.e., $\sum_{j \in \mathcal{W}_{m+1}} y_j \leq 1$.
- A project cannot be terminated if it is to be tested, i.e., $z \neq j$ if $y_j = 1$ and $x_j = 1, \forall j > 0$.
- A project cannot be tested if it is to be terminated, i.e., $y_j = 0$ if $z = j$ and $x_j = 1, \forall j > 0$.

Let v denote a real-valued function defined on \mathbb{N}_0^n (\mathbb{N}_0 is the set of nonnegative integers and \mathbb{N}_0^n is its n -dimensional cross product). Also define $0 < \alpha < 1$ as the discount parameter. For a given policy $\ell = \tilde{\ell}$ and a starting state $\mathbf{X}(0) = \mathbf{x}$, the expected discounted reward over an infinite planning horizon, $v^{\tilde{\ell}}(\mathbf{x})$, can be written as

$$v^{\tilde{\ell}}(\mathbf{x}) = E \left[\sum_{j \in \mathcal{W}_{m+1}} \int_0^\infty e^{-\alpha t} \rho_j dN_j(t) - \sum_{i \in \mathcal{M}} \int_0^\infty e^{-\alpha t} c_i dN^{(i)}(t) - \int_0^\infty e^{-\alpha t} h(\mathbf{X}(t)) dt \middle| \mathbf{X}(0) = \mathbf{x}, \ell = \tilde{\ell} \right] \quad (1)$$

where $N_j(t)$ is the cumulative number of product launches from projects in category $j \in \mathcal{W}_{m+1}$ at time t and $N^{(i)}(t)$ is the cumulative number of experiments performed in stage i up to time t .

Define $\nu_{\tilde{\mathbf{x}}|\mathbf{x},\mathbf{u}}$ as the rate at which the system moves from state \mathbf{x} to state $\tilde{\mathbf{x}}$ if action $\mathbf{u} = (y_0, \dots, y_n, z) \in \mathcal{U}(\mathbf{x})$ is chosen in state \mathbf{x} . The time between the transition to state $\tilde{\mathbf{x}}$ and the transition to the next state is exponentially distributed with rate $\nu_{\cdot|\mathbf{x},\mathbf{u}} = \sum_{\tilde{\mathbf{x}}} \nu_{\tilde{\mathbf{x}}|\mathbf{x},\mathbf{u}}$ if action $\mathbf{u} = (y_0, \dots, y_n, z) \in \mathcal{U}(\mathbf{x})$ is chosen in state \mathbf{x} . Define t_k as the time of occurrence of the k th transition. Also let $t_0 = 0$. The state of the system stays constant between transitions, i.e., $\mathbf{X}(t) = \mathbf{X}(t_k) = (X_1(t_k), \dots, X_n(t_k))$ for $t_k \leq t < t_{k+1}$. Following Lippman (1975), we consider a uniformized version of the problem where the rate of transition ν is an upper bound for all states and controls, i.e., $\nu \geq \nu_{\cdot|\mathbf{x},\mathbf{u}}, \forall \mathbf{x}, \mathbf{u}$. Specifically, we will formulate the problem for the choice $\nu = \lambda + \sum_i \mu_i + \mu_{m+1}$. Thus the k th transition time interval $(t_{k+1} - t_k)$ is exponentially distributed with rate $\nu, \forall k$. With the uniform transition rate, we are able to transform the continuous-time control problem into an equivalent discrete-time control problem.

If action $\mathbf{u} = (y_0, \dots, y_n, z) \in \mathcal{U}(\mathbf{x})$ is selected in state \mathbf{x} , the next state is $\tilde{\mathbf{x}}$ with probability $P_{\mathbf{x},\tilde{\mathbf{x}}}(\mathbf{u})$.

Thus:

$$P_{\mathbf{x},\tilde{\mathbf{x}}}(\mathbf{u}) = \begin{cases} \frac{\lambda}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x} - e_j, z = j, \text{ and } j \notin \mathcal{W}_{m+1}, \\ \frac{\lambda + \mu_{m+1} y_j}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x} - e_j, z = j, \text{ and } j \in \mathcal{W}_{m+1}, \\ \frac{\mu_{m+1} y_j}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x} - e_j, z \neq j, \text{ and } j \in \mathcal{W}_{m+1}, \\ \frac{\mu_1 f_{0 \rightarrow j} y_0}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x} + e_j \text{ and } j \in \mathcal{W}_2, \\ \frac{\mu_i f_{j \rightarrow j'} y_j}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x} - e_j + e_{j'}, j \in \mathcal{W}_i, \text{ and } i \notin \{1, m+1\}, \\ \frac{\nu - \lambda \lceil \frac{z}{n} \rceil - \sum_{i=1}^{m+1} \mu_i \sum_{j \in \mathcal{W}_i} y_j}{\nu} & \text{if } \tilde{\mathbf{x}} = \mathbf{x}, \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

where e_j is the j th unit vector of dimension n . In this discrete-time framework, $N_j(t_k)$ is the cumulative number of product launches from projects in category $j \in \mathcal{W}_{m+1}$ at the time of the k th transition, $N^{(i)}(t_k)$ is the cumulative number of experiments performed in stage i at the time of

the k th transition, and $h(\mathbf{X}(t_k))$ is the holding cost rate during the time interval $[t_k, t_{k+1})$. Then, $v^{\tilde{\ell}}(\mathbf{x})$ in (1) can be rewritten as follows.

$$v^{\tilde{\ell}}(\mathbf{x}) = E \left[\sum_{k=1}^{\infty} \left(\frac{\nu}{\alpha + \nu} \right)^k \left(\sum_{j \in \mathcal{W}_{m+1}} \rho_j (N_j(t_k) - N_j(t_{k-1})) - \sum_{i \in \mathcal{M}} c_i (N^{(i)}(t_k) - N^{(i)}(t_{k-1})) \right) - \sum_{k=0}^{\infty} \left(\frac{\nu}{\alpha + \nu} \right)^k \frac{h(\mathbf{X}(t_k))}{\alpha + \nu} \middle| \mathbf{X}(0) = \mathbf{x}, \ell = \tilde{\ell} \right]. \quad (2)$$

Our objective is to identify a policy ℓ^* that maximizes the expected discounted reward. We below formulate the optimality equation that holds for the optimal reward function $v^* = v^{\ell^*}$ (see Section EC.1 of the online appendix for a discussion of the existence of the optimality equation):

$$v^*(\mathbf{x}) = \max_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \left\{ \left(\frac{\nu}{\alpha + \nu} \right) \left(\frac{\mu_{m+1} \sum_{j \in \mathcal{W}_{m+1}} \rho_j y_j}{\nu} - \sum_{i \in \mathcal{M}} \frac{\mu_i c_i \sum_{j \in \mathcal{W}_i} y_j}{\nu} + \sum_{\tilde{\mathbf{x}}} P_{\mathbf{x}, \tilde{\mathbf{x}}}(\mathbf{u}) v^*(\tilde{\mathbf{x}}) \right) - \frac{h(\mathbf{x})}{\alpha + \nu} \right\}. \quad (3)$$

Therefore, our continuous-time control problem is equivalent to a discrete-time control problem with discount factor $\nu/(\alpha + \nu)$ and reward per stage given by

$$\left(\frac{\nu}{\alpha + \nu} \right) \left(\frac{\mu_{m+1} \sum_{j \in \mathcal{W}_{m+1}} \rho_j y_j}{\nu} - \sum_{i \in \mathcal{M}} \frac{\mu_i c_i \sum_{j \in \mathcal{W}_i} y_j}{\nu} \right) - \frac{h(\mathbf{x})}{\alpha + \nu}.$$

As it is always possible to redefine the time scale, without loss of generality we assume $\alpha + \nu = 1$. Letting e_0 denote a zero vector of dimension n , the optimality equation in (3) can be simplified as follows:

$$v^*(\mathbf{x}) = \max_{z \in \{0, \dots, n\}} \left\{ \lambda v^*(\mathbf{x} - e_z) + \sum_{1 \leq i \leq m} \mu_i T_i v^*(\mathbf{x}) + \mu_{m+1} T_{m+1} v^*(\mathbf{x}) \right\} - h(\mathbf{x}) \quad (4)$$

where the operator T_i for project promotion decisions in stage i is defined as

$$T_i v(\mathbf{x}) = \max \left\{ v(\mathbf{x}), \max_{\substack{j \in \mathcal{W}_i \\ \text{s.t. } \mathbf{x} \geq e_j, \text{ and} \\ j \neq z \text{ if } i > 1 \text{ and } x_j = 1}} \sum_{j'} v(\mathbf{x} - e_j + e_{j'}) f_{j \rightarrow j'} - c_i \right\}$$

and the operator T_{m+1} for product launch decisions in stage $m + 1$ is defined as

$$T_{m+1} v(\mathbf{x}) = \max \left\{ v(\mathbf{x}), \max_{\substack{j \in \mathcal{W}_{m+1} \\ \text{s.t. } \mathbf{x} \geq e_j, \text{ and} \\ j \neq z \text{ if } x_j = 1}} v(\mathbf{x} - e_j) + \rho_j \right\}.$$

For a given state \mathbf{x} , the operator T_i , $i \leq m$, specifies whether or not to test a project in stage i , and which project to select if a project is to be tested, and the operator T_{m+1} specifies whether or not to launch a new product, and which project to select if a new product is to be launched.

3. The Case with a Single Experimental Stage

In this section we assume that the NPD process consists of (i) an experimental stage that generates one out of the K signals about the true nature of each project, and (ii) a product launch stage. We also assume that projects are never terminated. Thus:

ASSUMPTION 1. $m = 1$ and $z = 0$.

Assumption 1 implies that the number of project categories equals the number of signals that can be observed, i.e., $n = K$. Let c denote the experimentation cost in stage 1. Under Assumption 1, setting the uniform transition rate equal to $\mu_1 + \mu_2$, the optimality equation in (4) can be written as follows:

$$v^*(\mathbf{x}) = \mu_1 T_1 v^*(\mathbf{x}) + \mu_2 T_2 v^*(\mathbf{x}) - h(\mathbf{x}) \quad (5)$$

where the operators T_1 and T_2 are defined as

$$T_1 v(\mathbf{x}) = \max \left\{ v(\mathbf{x}), \sum_{1 \leq j \leq n} v(\mathbf{x} + e_j) f_{0 \rightarrow j} - c \right\}$$

and

$$T_2 v(\mathbf{x}) = \max \left\{ v(\mathbf{x}), \max_{1 \leq j \leq n \text{ s.t. } \mathbf{x} \geq e_j} v(\mathbf{x} - e_j) + \rho_j \right\}.$$

The operator T_1 specifies when to test a new product idea. The operator T_2 specifies when to launch a new product, and which project to select when a new product is to be launched.

We will establish the optimal project promotion policy through the structural properties of our optimal reward function in (5). We introduce additional auxiliary indices d, l, q , and w for the categories such that $0 < d \leq l \leq q \leq w$ and $\rho_d \geq \rho_l \geq \rho_q \geq \rho_w$. We define \widehat{V} as the set of real-valued functions g on \mathbb{N}_0^n that satisfy the following properties:

(P.1) $g(\mathbf{x} + e_w) \leq g(\mathbf{x} + e_q), \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (q, w)$ s.t. $0 < q < w$,

(P.2) $g(\mathbf{x} + e_w) \leq g(\mathbf{x}) + \rho_w, \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall w > 0$,

(P.3) $g(\mathbf{x} + e_q) + \rho_w \leq g(\mathbf{x} + e_w) + \rho_q, \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (q, w)$ s.t. $0 < q < w$,

(P.4) $g(\mathbf{x} + e_q + e_l) - g(\mathbf{x} + e_q + e_d) \leq g(\mathbf{x} + e_l) - g(\mathbf{x} + e_d), \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (d, l, q)$ s.t. $0 < d \leq l \leq q$,

(P.5) $g(\mathbf{x} + e_q + e_l) - g(\mathbf{x} + e_q + e_d) \leq g(\mathbf{x} + e_w + e_l) - g(\mathbf{x} + e_w + e_d), \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (d, l, q, w)$ s.t.

$$0 < d \leq l \leq q \leq w,$$

(P.6) $\sum_{1 \leq j \leq l} g(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{l < j \leq n} g(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - g(\mathbf{x} + e_q)$
 $\leq \sum_{1 \leq j \leq l} g(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{l < j \leq n} g(\mathbf{x} + e_j) f_{0 \rightarrow j} - g(\mathbf{x}), \forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (l, q)$ s.t. $0 < l \leq q$, and

$$\begin{aligned}
(\mathbf{P.7}) \quad & \sum_{1 \leq j \leq l} g(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{l < j \leq n} g(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - g(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} g(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{l < j \leq n} g(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - g(\mathbf{x} + e_w), \quad \forall \mathbf{x} \text{ and } \forall (l, q, w) \text{ s.t.} \\
& 0 < l \leq q < w.
\end{aligned}$$

The lemma below shows that the optimal reward function in (5) satisfies Properties 1–7:

LEMMA 1. *Under Assumption 1, if $v \in \widehat{V}$, then $Tv \in \widehat{V}$, where $Tv(\mathbf{x}) = \mu_1 T_1 v(\mathbf{x}) + \mu_2 T_2 v(\mathbf{x}) - h(\mathbf{x})$. Furthermore, the optimal reward function v^* is an element of \widehat{V} .*

We now consider the implications of Properties 1–7: Property 1 states that the optimal reward function weakly increases as the number of projects with higher expected reward increases, keeping the total number of projects fixed. Property 2 shows that a new product launch is always beneficial no matter what category is chosen, as long as it is feasible (recall that $\rho_j \geq 0, \forall j$). However, Property 3 says that choosing the category with higher expected reward is more desirable for product launch. Property 4 states that the incentive to replace a project in category l with one having higher expected reward weakly increases as the number of projects with expected rewards lower than ρ_l increases. Property 5 shows that the incentive to replace a project in category l with one having higher expected reward weakly increases when a project with expected reward less than ρ_l is replaced with one having higher expected reward (but lower than ρ_l). Property 6 says that, when $l = 1$, the desirability of testing a new product idea weakly decreases as the number of projects in any category increases. Finally, Property 7 implies that, when $l = 1$, the desirability of testing a new product idea weakly decreases as the number of projects with higher expected reward increases, keeping the total number of projects fixed.

The intuition behind Properties 1–3 is straightforward: It is more desirable to have a project with higher expected reward, which will conceivably take priority over those projects with lower expected rewards in resource utilization for product launch. More critically, Properties 4–7 enable us to uncover the role congestion plays in promotion decisions for new product ideas:

- As the number of projects in category l increases, the system becomes more congested from the perspective of a new product idea: If a new product idea is tested in the experimental stage, it might become a project with expected reward less than ρ_l , taking a lower priority than all projects in category l in the queue for the second stage. Such a project waits longer for access to resources of the second stage when there are more projects in category l . Consequently, since any delay in the project completion time is costly, it is less desirable to test a new product idea (Property 6).

- Likewise, when a project is replaced with one having higher expected reward, the system again becomes more congested from the same perspective: A new product idea, once tested, is more likely to see a greater number of high priority projects in the queue for the second stage if a low value project is replaced with a high value project. Thus it is less desirable to test a new product idea (Property 7).
- As the system becomes more congested due to an increase in the number of low value projects, the system anticipates a lower throughput rate in the experimental stage in the future (due to Property 6), and eventually a small number of high value projects. To hedge against future scarcity of high value projects, the controller would prefer to trade a project in category l with one having higher expected reward, and such a trade becomes more desirable as the number of projects with expected rewards *less* than ρ_l increases (Property 4).
- Likewise, as the system becomes more congested due to a rise in the expected reward of a low value project, the controller would prefer to trade a project in category l with one having higher expected reward in anticipation of a small number of high value projects (due to Property 7). Such a trade becomes more desirable when a project with expected reward *less* than ρ_l is replaced with one having higher expected reward (but *less* than ρ_l) (Property 5).

The structural properties of our optimal reward function allow us to establish the optimality of a *new* type of policy: state-dependent noncongestive-promotion (SDNCP) policy.

THEOREM 1. *Under Assumption 1, the optimal stationary project promotion policy is a state-dependent noncongestive-promotion policy with state-dependent promote-up-to levels $S_j^*(\mathbf{x}_{-j})$: It is optimal to test a new product idea if and only if $x_j < S_j^*(\mathbf{x}_{-j})$, $\forall j$, where $\mathbf{x}_{-j} = (x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_n)$ is an $n-1$ dimensional vector of the numbers of projects in categories $k \neq j$. The optimal policy has the following additional properties:*

- i. The optimal promote-up-to level $S_j^*(\mathbf{x}_{-j})$ weakly decreases as the number of projects in category $k \neq j$ increases, $\forall j$.*
- ii. The optimal promote-up-to level $S_j^*(\mathbf{x}_{-j})$ weakly decreases as the expected reward of a project in category $k \neq j$ increases, $\forall j$.*
- iii. It is always optimal to launch a new product if there are projects available for product launch.*
- iv. It is always optimal to choose a project with highest expected reward for product launch.*
- v. It is never optimal to interrupt any experiment.*

Using Property 6, Theorem 1 establishes the optimality of a state-dependent non-congestive promotion policy. Such a policy protects the system against congestion, restricting the number of projects that can be held in each category. Points (i) and (ii) show that the promote-up-to levels weakly decrease as the system becomes more congested with an increase in the number of projects in any category (due to Property 6), or with an increase in the expected reward of any project in the system (due to Property 7). Points (iii) and (iv) state that it is always optimal to launch a new product if it is feasible (due to Property 2), and it is optimal to choose a project with highest expected reward for product launch (due to Property 3). Lastly, point (v) says it is never optimal to interrupt an experiment once it has been initiated.

To our knowledge, we are the first to characterize the optimal project promotion policy for a Markovian NPD process with stage-dependent resources. We also significantly extend the existing NPD literature by showing that optimal promotion decisions depend on *both* the number of projects and the breakdown of projects into categories if an imperfectly informative stage exists. Now suppose that the experimental stage is uninformative, implying that there is only one project category. Then the problem becomes similar to the single-item inventory model introduced by Ha (1997b). A non-congestive promotion policy remains optimal in this special case of our model: it is optimal to promote a new product idea if and only if the number of projects in the system is less than a fixed promote-up-to level. This concurs well with the optimal replenishment policy in Ha (1997b): it is optimal to order an item if and only if the inventory level is less than a fixed base-stock level. But what if there are multiple non-experimental stages? We consider that next.

4. The Case with Multiple Non-Experimental Stages

In this section we assume that the NPD process consists of multiple non-experimental stages that do *not* provide any information about the true nature of any project, but which the project must complete before launch. In the literature there are many examples of non-experimental stages. See, for instance, the queueing network in Adler et al. (1995) and the project scheduling problem in Brucker et al. (1999). We also assume that a project may be terminated even if it is in the process of being tested, and vice versa. (The latter assumption is benign since the optimal policy in this section never terminates a project, cf. Theorem 2.) Thus:

ASSUMPTION 2. $K = 1$, and it is feasible to have $z = j$ and $y_j = 1$ when $x_j = 1$.

Assumption 2 implies that the number of project categories equals the number of stages, i.e., $n = m$. Let ρ denote the expected reward for a new product. Under Assumption 2, the optimality equation in (4) can be written as follows:

$$v^*(\mathbf{x}) = \lambda T_0 v^*(\mathbf{x}) + \sum_{1 \leq i \leq m} \mu_i T_i v^*(\mathbf{x}) + \mu_{m+1} T_{m+1} v^*(\mathbf{x}) - h(\mathbf{x}) \quad (6)$$

where the operator T_0 for project termination decisions is defined as

$$T_0 v(\mathbf{x}) = \max_{\substack{z \in \{0, \dots, m\} \\ \text{s.t. } \mathbf{x} \geq e_z}} v(\mathbf{x} - e_z),$$

the operator T_i for project promotion decisions in stage i is defined as

$$T_i v(\mathbf{x}) = \begin{cases} \max\{v(\mathbf{x}), v(\mathbf{x} - e_{i-1} + e_i) - c_i\} & \text{if } \mathbf{x} \geq e_{i-1}, \\ v(\mathbf{x}) & \text{otherwise,} \end{cases}$$

and the operator T_{m+1} for product launch decisions in stage $m + 1$ is defined as

$$T_{m+1} v(\mathbf{x}) = \begin{cases} \max\{v(\mathbf{x}), v(\mathbf{x} - e_m) + \rho\} & \text{if } \mathbf{x} \geq e_m, \\ v(\mathbf{x}) & \text{otherwise.} \end{cases}$$

We proceed to characterize the structural properties of our optimal reward function in (6). We will use the indices l , q , and w for the stages; the alphabetical ordering $l \rightarrow q \rightarrow w$ corresponds to a progress from earlier to later stages. Let e_{m+1} denote a zero vector of dimension n . We define \tilde{V} as the set of real-valued functions g on \mathbb{N}_0^n that satisfy the following properties:

(P.8) $g(\mathbf{x} + e_q) \leq g(\mathbf{x} + e_w)$, $\forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (q, w)$ s.t. $0 < q < w \leq m$,

(P.9) $g(\mathbf{x} + e_m) \leq g(\mathbf{x}) + \rho$, $\forall \mathbf{x} \in \mathbb{N}_0^n$,

(P.10) $g(\mathbf{x} + e_q + e_{w-1}) - g(\mathbf{x} + e_{q-1} + e_{w-1}) \leq g(\mathbf{x} + e_q + e_w) - g(\mathbf{x} + e_{q-1} + e_w)$, $\forall \mathbf{x} \in \mathbb{N}_0^n$, and $\forall (q, w)$ s.t. $q \neq w$ and $0 < q, w \leq m + 1$,

(P.11) $g(\mathbf{x} + e_l + e_q) - g(\mathbf{x} + e_{l-1} + e_q) \leq g(\mathbf{x} + e_l + e_w) - g(\mathbf{x} + e_{l-1} + e_w)$, $\forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (l, q, w)$ s.t. $0 < l \leq q < w \leq m + 1$,

(P.12) $g(\mathbf{x} + e_w + e_l) - g(\mathbf{x} + e_{w-1} + e_l) \leq g(\mathbf{x} + e_w + e_q) - g(\mathbf{x} + e_{w-1} + e_q)$, $\forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall (l, q, w)$ s.t. $0 \leq l < q \leq w - 1 \leq m$, and

(P.13) $g(\mathbf{x} + 2e_l) - g(\mathbf{x} + e_{l-1} + e_l) \leq g(\mathbf{x} + e_l + e_{l-1}) - g(\mathbf{x} + 2e_{l-1})$, $\forall \mathbf{x} \in \mathbb{N}_0^n$ and $\forall l$ s.t. $1 \leq l \leq m$.

The lemma below shows that the optimal reward function in (6) satisfies Properties 8–13:

LEMMA 2. *Under Assumption 2, if $v \in \tilde{V}$, then $Tv \in \tilde{V}$, where $Tv(\mathbf{x}) = \lambda T_0 v(\mathbf{x}) + \sum_{1 \leq i \leq m} \mu_i T_i v(\mathbf{x}) + \mu_{m+1} T_{m+1} v(\mathbf{x}) - h(\mathbf{x})$. Furthermore, the optimal reward function v^* is an element of \tilde{V} .*

We now consider the implications of Properties 8–13: Property 8 implies that the optimal reward function weakly increases as projects move from one stage to the next stage. Property 9 shows that a new product launch is always beneficial as long as it is feasible (recall that $\rho \geq 0$). Property 10 says that the incentive to promote a project from one stage to the next stage weakly increases if any project in another stage gets promoted. Property 11 states that promoting a project from stage $l - 1$ to stage l is more desirable when a project in stage $q \geq l$ is replaced with another project in stage $w > q$. Furthermore, Property 12 states that promoting a project from stage $w - 1$ to stage w is more desirable when a project in stage $l < q$ is replaced with another project in stage $q \leq w - 1$. Conversely, Property 13 shows that the incentive to promote a project from one stage to the next stage weakly decreases if another project in the same stage gets promoted.

The intuition behind Properties 8 and 9 is as follows: Since delays in project completion times are costly, it is more desirable to have projects that are closer to the product launch stage, as well as to launch a new product immediately if it is feasible. We show in the online appendix that Property 10 implies Properties 11–13, which enable us to uncover the role congestion plays in promotion decisions in each stage:

- From the perspective of a project in stage $l - 1$, a project in stage $q \geq l$ causes more congestion than in stage $w > q$: A project in stage $l - 1$, if promoted along the NPD process, is more likely to catch the project in stage q than in stage w . Hence, if a project in stage w is replaced with a project in stage q , it becomes more likely that projects accumulate in the same stage and create a bottleneck, leading to investments with a lower rate of return for some projects. Thus it becomes less desirable to promote the project in stage $l - 1$ (Property 11).
- Conversely, there is a greater benefit in promoting the project in stage w if a project in stage $l < q$ is replaced with a project in stage $q < w$ (Property 12). The bottleneck of the NPD process is more likely to occur in stage w when projects in stage $l < w$ get closer to stage w . Promoting a project in stage w might help us avoid such an occurrence of the bottleneck.
- Although the incentive to promote a project *increases* as projects in later or earlier stages get promoted (Properties 11 and 12), the incentive to promote a project *decreases* if another project in the same stage gets promoted (Property 13): Successively advancing projects from the same stage may increase the risk of creating a bottleneck in a further stage.

Using the structural properties of our optimal reward function in (6), we again establish the optimality of an SDNCP policy:

THEOREM 2. *Under Assumption 2, the optimal stationary project promotion policy in each stage i is a state-dependent noncongestive-promotion policy with state-dependent promote-up-to levels $S_i^*(\mathbf{x}_{-i})$: It is optimal to promote a project to stage i if and only if $x_i < S_i^*(\mathbf{x}_{-i})$, where $\mathbf{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m)$ is an $m - 1$ dimensional vector of the numbers of projects in stages $k \neq i$.*

The optimal policy has the following additional properties:

- i. The optimal promote-up-to level $S_i^*(\mathbf{x}_{-i})$ weakly increases as the number of projects in stage $j > i$ decreases.*
- ii. The optimal promote-up-to level $S_i^*(\mathbf{x}_{-i})$ weakly increases as the number of projects in stage $j < i$ increases.*
- iii. The optimal promote-up-to level $S_i^*(\mathbf{x}_{-i})$ weakly increases as projects in stage $j \neq i - 1$ move along the process.*
- iv. It is always optimal to launch a new product if there are projects available for product launch.*
- v. It is never optimal to interrupt any experiment.*
- vi. It is never optimal to terminate any project.*

Using Property 11, Theorem 2 proves the optimality of a state-dependent non-congestive promotion policy: Each stage of the NPD process is protected against congestion through promote-up-to levels. Points (i) and (ii) state that the promote-up-to level in a given stage weakly increases as a later stage becomes less congested (due to Property 11), or as an earlier stage becomes more congested (due to Property 12). Point (iii) shows that the promote-up-to level in stage i weakly increases as projects in stage $j \neq i - 1$ get promoted to the next stage (due to Property 10). Point (iv) shows that it is always optimal to launch a new product if it is feasible (due to Property 9). Point (v) says it is never optimal to interrupt an experiment once it has been initiated. Point (vi) says it is never optimal to terminate any active project.

Theorem 2 significantly broadens current knowledge of NPD processes by revealing the impacts of *both* upstream and downstream projects on optimal promotion decisions when different stages require different resources. A threshold policy similar to the one described in Theorem 2 has been studied in the Markovian inventory literature. See, for instance, Benjaafar et al. (2011) who show the optimality of a state-dependent base-stock policy for an assembly system with multiple stages.

5. Heuristic Policies and Their MIP Formulations

In this section, we take as our optimization criterion the average reward rate over an infinite planning horizon. (Our structural results in Sections 3 and 4 continue to hold in the average reward

case; we provide a proof of this result in the online appendix.) Section 5.1 formulates a linear program to find a global optimal solution to our general problem of Section 2. Sections 5.2 and 5.3 develop two *heuristic* project promotion policies for our general problem, formulating mixed integer programs to find the optimal average reward within each heuristic class. Section 5.4 considers a naïve project promotion policy that always promotes projects if feasible, formulating a linear program to find the optimal average reward in this case. Finally, Section 5.5 ranks our heuristic policies in term of their optimal rewards.

5.1. Global Optimal Solution

Define $\pi_{\mathbf{x},\mathbf{u}}$ as the limiting probability that the system is in state \mathbf{x} and action $\mathbf{u} \in \mathcal{U}(\mathbf{x})$ is chosen. As a computational requirement, we restrict the state space to be finite: Define \bar{x} as an upper bound for the total number of projects in the system, and \mathcal{S} as the set of all states \mathbf{x} obeying $\sum_i x_i \leq \bar{x}$. (The upper bound should be sufficiently high so that the globally optimal reward does not change with a further increase in the upper bound.) Thus, for any state $\mathbf{x} \in \mathcal{S}$ and any action $\mathbf{u} = (y_0, \dots, y_n, z) \in \mathcal{U}(\mathbf{x})$, we must have $y_0 = 0$ if $\sum_i x_i = \bar{x}$. The globally optimal average reward Z^* can be found by solving the following linear program (see Puterman 1994):

$$\begin{aligned}
 (\mathcal{LP}) \quad & \text{maximize} && \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{j \in \mathcal{W}_{m+1}} \mu_{m+1} \rho_j y_j \pi_{\mathbf{x},\mathbf{u}} - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{1 \leq i \leq m} \sum_{j \in \mathcal{W}_i} \mu_i c_i y_j \pi_{\mathbf{x},\mathbf{u}} \\
 & && - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} h(\mathbf{x}) \pi_{\mathbf{x},\mathbf{u}} \\
 & \text{subject to} && \sum_{\mathbf{u} \in \mathcal{U}(\tilde{\mathbf{x}})} \pi_{\tilde{\mathbf{x}},\mathbf{u}} \sum_{\mathbf{x} \in \mathcal{S}} \nu_{\mathbf{x}|\tilde{\mathbf{x}},\mathbf{u}} - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \nu_{\tilde{\mathbf{x}}|\mathbf{x},\mathbf{u}} \pi_{\mathbf{x},\mathbf{u}} = 0, \quad \forall \tilde{\mathbf{x}} \in \mathcal{S}, \tag{C.1}
 \end{aligned}$$

$$\sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \pi_{\mathbf{x},\mathbf{u}} = 1, \tag{C.2}$$

$$\pi_{\mathbf{x},\mathbf{u}} \geq 0, \quad \forall \mathbf{x} \in \mathcal{S}, \quad \forall \mathbf{u} \in \mathcal{U}(\mathbf{x}). \tag{C.3}$$

The first term of the objective function corresponds to the time-average reward for completed projects, the second term corresponds to the time-average experimentation costs, and the last term corresponds to the time-average project holding cost. Constraints (C.1) and (C.2) are the balance equations and normalization constraint that together yield the limiting probability values.

Notice that the above linear program may yield a *randomized* policy as the global optimal solution, i.e., there may exist a state $\mathbf{x} \in \mathcal{S}$ such that $\pi_{\mathbf{x},\mathbf{u}_1} > 0$ and $\pi_{\mathbf{x},\mathbf{u}_2} > 0$, where $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{U}(\mathbf{x})$. We below construct three heuristic policies with *deterministic* policy structure on project *promotion*

decisions, as opposed to the global optimal solution. Because our structural results in Sections 3 and 4 are only restricted to project promotion decisions, we do not impose any structure on project *termination* decisions. Thus our heuristics may still yield a randomized policy (for termination), with potentially many fewer states having randomized decisions.

5.2. State-Dependent Non-Congestive Promotion (SDNCP)

Based on our structural results in Sections 3 and 4, we develop an SDNCP policy for our general problem in Section 2: Define $\mathbf{x}_{-j} = (x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_n)$ as a vector of the numbers of projects in categories $k \neq j$. A project in category $j \in \mathcal{W}_i$ (that is not to be terminated if $j > 0$) is advanced to stage $i < m + 1$ if and only if

- The number of projects in each successor category is less than a state-dependent promote-up-to level, i.e., $x_{j'} < S_{j'}(\mathbf{x}_{-j'})$, $\forall j' \in \mathcal{W}_{i+1}$ s.t. $f_{j \rightarrow j'} > 0$; and
- There is no project with higher expected return in stage $i - 1$ (if $i > 1$).

In addition, a new product is launched from a project in category $j \in \mathcal{W}_{m+1}$ (that is not to be terminated) if and only if there is no project with higher expected return in stage m . We also enforce the following additional properties on the SDNCP policy:

- (a) The promote-up-to level in one category is nondecreasing in the number of projects in any other category in an earlier stage;
- (b) The promote-up-to level in one category is nondecreasing in the expected reward of a project in any other category in an earlier stage;
- (c) The promote-up-to level in one category is nonincreasing in the number of projects in any other category in the same stage or a later stage; and
- (d) The promote-up-to level in one category is nonincreasing in the expected reward of a project in any other category in the same stage or a later stage.

We proceed to the MIP formulation of this heuristic class. First, for $b \in \mathbb{N}_0$, define $\mathbb{S}_j(\mathbf{x}_{-j}, b)$ as the set of state-action pairs (\mathbf{x}, \mathbf{u}) such that the limiting probability that the system is in state \mathbf{x} and action \mathbf{u} is chosen should be zero if the promote-up-to level $S_j(\mathbf{x}_{-j})$ equals b . Also, define $z_b^{S_j(\mathbf{x}_{-j})}$ as a binary variable as follows:

$$z_b^{S_j(\mathbf{x}_{-j})} = \begin{cases} 1 & \text{if } S_j(\mathbf{x}_{-j}) = b, \\ 0 & \text{otherwise.} \end{cases}$$

The optimal solution of the MIP problem should satisfy constraints (C.1)-(C.3) of the LP formulation in Section 5.1. The optimal solution should also select exactly one promote-up-to level in each category, given the numbers of projects in all other categories:

$$\sum_{b \in \mathbb{N}_0} z_b^{S_j(\mathbf{x}_{-j})} = 1, \quad \forall j, \forall \mathbf{x}_{-j}. \quad (\text{C.4})$$

Constraint (C.5) ensures that (a) the promote-up-to level in one category is nondecreasing in the number of projects in any other category in an earlier stage:

$$z_b^{S_j(\mathbf{x}_{-j})} \leq \sum_{b' < b} z_{b'}^{S_j(\mathbf{x}_{-j+e_k})}, \quad \forall j \in \mathcal{W}_i, \forall k \in \mathcal{W}_{i'} \text{ s.t. } 2 \leq i' < i, \forall \mathbf{x}_{-j}, \forall b. \quad (\text{C.5})$$

Constraint (C.6) ensures that (b) the promote-up-to level in one category is nondecreasing in the expected reward of a project in any other category in an earlier stage:

$$z_b^{S_j(\mathbf{x}_{-j})} \leq \sum_{b' < b} z_{b'}^{S_j(\mathbf{x}_{-j-e_k+e_{k'}})}, \quad \forall j \in \mathcal{W}_i, \forall k, k' \in \mathcal{W}_{i'} \text{ s.t. } 2 \leq i' < i \text{ and } \rho_{k'} > \rho_k, \forall \mathbf{x}_{-j}, \forall b. \quad (\text{C.6})$$

Constraint (C.7) ensures that (c) the promote-up-to level in one category is nonincreasing in the number of projects in any other category in the same stage or a later stage:

$$z_b^{S_j(\mathbf{x}_{-j})} \leq \sum_{b' \leq b} z_{b'}^{S_j(\mathbf{x}_{-j+e_k})}, \quad \forall j \in \mathcal{W}_i, \forall k \in \mathcal{W}_{i'} \setminus \{j\} \text{ s.t. } i \leq i', \forall \mathbf{x}_{-j}, \forall b. \quad (\text{C.7})$$

Constraint (C.8) ensures that (d) the promote-up-to level in one category is nonincreasing in the expected reward of a project in any other category in the same stage or a later stage:

$$z_b^{S_j(\mathbf{x}_{-j})} \leq \sum_{b' \leq b} z_{b'}^{S_j(\mathbf{x}_{-j-e_k+e_{k'}})}, \quad \forall j \in \mathcal{W}_i, \forall k, k' \in \mathcal{W}_{i'} \setminus \{j\} \text{ s.t. } i \leq i' \text{ and } \rho_{k'} > \rho_k, \forall \mathbf{x}_{-j}, \forall b. \quad (\text{C.8})$$

Constraint (C.9) links our binary variables to the appropriate limiting probability variables:

$$\sum_{(\mathbf{x}, \mathbf{u}) \in \mathbb{S}_j(\mathbf{x}_{-j}, b)} \pi_{\mathbf{x}, \mathbf{u}} \leq 1 - z_b^{S_j(\mathbf{x}_{-j})}, \quad \forall b, \forall j, \forall \mathbf{x}_{-j}. \quad (\text{C.9})$$

In constraint (C.9), if $z_b^{S_j(\mathbf{x}_{-j})}$ is one, then all limiting probability variables corresponding to the state-action pairs in set $\mathbb{S}_j(\mathbf{x}_{-j}, b)$ are forced to equal zero. Otherwise, this constraint becomes redundant. We next enforce the following constraint that always launches a new product if feasible:

$$\pi_{\mathbf{x}, \mathbf{u}} = 0, \quad \forall (\mathbf{x}, \mathbf{u}) \text{ s.t. } \sum_{j \in \mathcal{W}_{m+1}} y_j = 0 \text{ and either } \sum_{j \in \mathcal{W}_{m+1}} x_j > 1 \text{ or } \left(\sum_{j \in \mathcal{W}_{m+1}} x_j = 1 \text{ and } z \notin \mathcal{W}_{m+1} \right). \quad (\text{C.10})$$

Lastly, the following constraint ensures that, if a project in a given stage is to be promoted, it should be selected from the most valuable category in that stage:

$$\pi_{\mathbf{x}, \mathbf{u}} = 0, \forall (\mathbf{x}, \mathbf{u}) \text{ where } \exists i > 1, \exists j, j' \in \mathcal{W}_i \text{ s.t. } x_{j'} > 0, y_j = 1, \text{ and } \rho_{j'} > \rho_j. \quad (\text{C.11})$$

The optimal average reward of this policy Z_{SDNCP} can be found from the following MIP problem:

$$\begin{aligned} (\text{SDNCP}) \quad & \text{maximize} \quad \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{j \in \mathcal{W}_{m+1}} \mu_{m+1} \rho_j y_j \pi_{\mathbf{x}, \mathbf{u}} - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{1 \leq i \leq m} \sum_{j \in \mathcal{W}_i} \mu_i c_i y_j \pi_{\mathbf{x}, \mathbf{u}} \\ & - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} h(\mathbf{x}) \pi_{\mathbf{x}, \mathbf{u}} \\ & \text{subject to} \quad (\text{C.1}) - (\text{C.11}). \end{aligned}$$

See Bhandari et al. (2008) and Nadar et al. (2015) for similar MIP formulations in different contexts.

5.3. Fixed Non-Congestive Promotion (FNCP)

We next consider a simpler non-congestive promotion policy with fixed promote-up-to levels across stages, to evaluate the importance of the more complex SDNCP policy. Specifically, a project in category $j \in \mathcal{W}_i$ (that is not to be terminated if $j > 0$) is advanced to stage $i < m + 1$ if and only if

- The total number of projects in stage i is less than a fixed promote-up-to level S_i , i.e., $\sum_{j' \in \mathcal{W}_{i+1}} x_{j'} < S_i$; and
- There is no project with higher expected return in stage $i - 1$ (if $i > 1$).

Product launch decisions have the same structure as those in the SDNCP policy: A new product is launched from a project in category $j \in \mathcal{W}_{m+1}$ (that is not to be terminated) if and only if there is no project with higher expected return in stage m . Thus, unlike SDNCP, FNCP takes into account only the total number of projects in stage i in promotion decisions in stage $i - 1$, ignoring the number of projects in any other stage as well as the breakdown of projects into categories throughout the NPD process. Such simplification over SDNCP conceivably alleviates the computational burden, as verified by our numerical experiments in Section 6.

We proceed to the MIP formulation of this heuristic class. Define $\mathbb{S}_i(b)$ as the set of state-action pairs (\mathbf{x}, \mathbf{u}) such that the limiting probability that the system is in state \mathbf{x} and action \mathbf{u} is chosen should be zero if the promote-up-to level S_i equals b . Also, define $z_b^{S_i}$ as a binary variable as follows:

$$z_b^{S_i} = \begin{cases} 1 & \text{if } S_i = b, \\ 0 & \text{otherwise.} \end{cases}$$

The optimal solution of the MIP problem should satisfy constraints (C.1)-(C.3) of the LP formulation in Section 5.1, and constraints (C.10)-(C.11) of the MIP formulation of the SDNCP policy. The optimal solution should also select exactly one promote-up-to level in each stage:

$$\sum_{b \in \mathbb{N}_0} z_b^{S_i} = 1, \quad \forall i. \quad (\text{C.12})$$

Constraint (C.13) links our binary variables to the appropriate limiting probability variables:

$$\sum_{(\mathbf{x}, \mathbf{u}) \in \mathbb{S}_i(b)} \pi_{\mathbf{x}, \mathbf{u}} \leq 1 - z_b^{S_i}, \quad \forall b, \forall i. \quad (\text{C.13})$$

The optimal average reward of this policy Z_{FNCP} can be found from the following MIP problem:

$$\begin{aligned} (\text{FNCP}) \quad & \text{maximize} \quad \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{j \in \mathcal{W}_{m+1}} \mu_{m+1} \rho_j y_j \pi_{\mathbf{x}, \mathbf{u}} - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{1 \leq i \leq m} \sum_{j \in \mathcal{W}_i} \mu_i c_i y_j \pi_{\mathbf{x}, \mathbf{u}} \\ & - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} h(\mathbf{x}) \pi_{\mathbf{x}, \mathbf{u}} \\ & \text{subject to} \quad (\text{C.1}), (\text{C.2}), (\text{C.3}), (\text{C.10}), (\text{C.11}), (\text{C.12}), (\text{C.13}). \end{aligned}$$

5.4. Continuous Promotion (CP)

In this heuristic, which maximizes throughput, a new product idea is always tested as long as the total number of active projects is below \bar{x} , and a project in category $j \in \mathcal{W}_i$ (that is not to be terminated) is advanced to stage $i > 1$ if and only if there is no project with higher expected return in stage $i - 1$. The CP policy should satisfy constraints (C.1)-(C.3) of the LP formulation in Section 5.1, and constraint (C.11) of the MIP formulation of the SDNCP policy. In addition, we enforce the following constraint that always tests a new product idea if feasible:

$$\pi_{\mathbf{x}, \mathbf{u}} = 0, \quad \forall (\mathbf{x}, \mathbf{u}) \text{ s.t. } y_0 = 0 \text{ and } \sum_j x_j < \bar{x}. \quad (\text{C.14})$$

Lastly, the following constraint always promotes a project in each stage and always launches a new product if feasible:

$$\pi_{\mathbf{x}, \mathbf{u}} = 0, \quad \forall (\mathbf{x}, \mathbf{u}) \text{ where } \exists i > 1 \text{ s.t. } \sum_{j \in \mathcal{W}_i} y_j = 0 \text{ and either } \sum_{j \in \mathcal{W}_i} x_j > 1 \text{ or } \left(\sum_{j \in \mathcal{W}_i} x_j = 1 \text{ and } z \notin \mathcal{W}_i \right). \quad (\text{C.15})$$

The optimal average reward of this policy Z_{CP} can be found from the following linear program:

$$\begin{aligned} (\text{CP}) \quad & \text{maximize} \quad \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{j \in \mathcal{W}_{m+1}} \mu_{m+1} \rho_j y_j \pi_{\mathbf{x}, \mathbf{u}} - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} \sum_{1 \leq i \leq m} \sum_{j \in \mathcal{W}_i} \mu_i c_i y_j \pi_{\mathbf{x}, \mathbf{u}} \\ & - \sum_{\mathbf{x} \in \mathcal{S}} \sum_{\mathbf{u} \in \mathcal{U}(\mathbf{x})} h(\mathbf{x}) \pi_{\mathbf{x}, \mathbf{u}} \\ & \text{subject to} \quad (\text{C.1}), (\text{C.2}), (\text{C.3}), (\text{C.11}), (\text{C.14}), (\text{C.15}). \end{aligned}$$

5.5. Analytical Comparison of Heuristic Policies

The proposition below ranks our heuristic policies in terms of their optimal average rewards:

PROPOSITION 1. $Z^* \geq Z_{SDNCP} \geq Z_{FNCP} \geq Z_{CP}$.

Proof of Proposition 1. The first inequality holds since \mathcal{LP} is a relaxation the MIP formulation of SDNCP. The second inequality holds since FNCP is a subclass of SDNCP: SDNCP becomes FNCP if the promote-up-to level $S_j(\mathbf{x}_{-j})$ in each category $j \in \mathcal{W}_{i+1}$ is set equal to $\max\left\{0, S_i - \sum_{k \in \mathcal{W}_{i+1}: k \neq j} x_k\right\}$. Such a promote-up-to level satisfies Properties (a)-(d) in Section 5.2: It is constant across state variables in an earlier or a later stage. It is also constant across state variables in stage i as long as the total number of projects in stage i (in categories other than j) is constant. However, as the number of projects in any other category in stage i increases, the promote-up-to level *either* decreases or remains the same. Finally, the third inequality holds since CP is a subclass of FNCP: FNCP becomes CP if the promote-up-to level S_i in each stage i is a very large number. \square

6. Numerical Experiments

We numerically compare the globally optimal policy to our heuristic policies, investigating how system parameters affect the relative rewards of each policy. We confine our analysis to NPD processes that involve (i) two experimental stages, each generating one out of two signals; and (ii) a product launch stage (i.e., $m = 2$ and $K = 2$). We construct our numerical instances by varying values of $\mu_1, \mu_2, \mu_3, h, c_1, c_2, \phi^{(1)}$, and $\phi^{(2)}$. We assume in all instances that holding cost rates are linear (i.e., $h(\mathbf{x}) = h \sum_j x_j$), initial beliefs for a new product idea are evenly distributed among two possible states of the true nature (i.e., $p_{0,1} = p_{0,2} = 0.5$), and the probability that an experiment reveals the true nature of a project is independent from the true nature in each experimental stage (i.e., $\phi_{1,1}^{(1)} = \phi_{2,2}^{(1)} = \phi^{(1)}$ and $\phi_{1,1}^{(2)} = \phi_{2,2}^{(2)} = \phi^{(2)}$). Lastly, in all instances, we impose $\bar{x} = 5$, $r_1 = 40$, $r_2 = 0$, and $\lambda = 100$.

For each numerical instance we solve the LP and MIP problems in Section 5 to find the maximum average rewards. We compare the heuristic policies in terms of (i) their percentage differences from optimal reward Z^* , calculated as $100 \times \frac{Z^* - Z_H}{Z^*}$ where $H \in \{\text{SDNCP, FNCP, CP}\}$; and (ii) their computation times. We coded the LP and MIP formulations in the Java programming language, incorporating CPLEX 12.5 optimization package, and used a dual processor WinNT server, with Intel Core i7 2.67 GHz processor and 8 GB of RAM. We restricted the computation time of any instance to be no more than one hour.

Table 1 Numerical results for various experimentation and product launch rates.

μ_1	μ_2	μ_3	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
				SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.5	0.5	0.5	2.700	0.000	0.000	1.222	145.04	1.00	0.30
-	-	1.0	3.366	0.000	0.000	0.119	80.70	1.76	0.34
-	-	1.5	3.532	0.000	0.000	0.000	131.28	1.83	0.19
-	1.0	0.5	3.331	0.000	0.000	0.000	96.75	1.54	0.18
-	-	1.0	4.319	0.000	0.000	0.000	148.46	1.63	0.32
-	-	1.5	4.623	0.000	0.000	0.000	142.32	1.82	0.23
-	1.5	0.5	3.509	0.000	0.000	0.000	91.06	1.91	0.16
-	-	1.0	4.635	0.000	0.000	0.000	118.86	1.08	0.20
-	-	1.5	5.003	0.000	0.000	0.000	100.31	1.67	0.20
1.0	0.5	0.5	3.542	0.000	0.508	12.168	139.42	1.60	0.16
-	-	1.0	4.765	0.000	0.021	5.939	35.95	1.39	0.18
-	-	1.5	5.052	0.000	0.000	5.523	156.79	1.27	0.22
-	1.0	0.5	4.608	0.000	2.192	3.516	67.31	1.55	0.25
-	-	1.0	7.224	0.000	0.000	0.000	59.55	1.28	0.33
-	-	1.5	7.991	0.000	0.000	0.000	155.42	1.93	0.19
-	1.5	0.5	4.893	0.000	3.004	3.556	217.39	2.13	0.18
-	-	1.0	7.958	0.000	0.000	0.000	141.91	1.85	0.30
-	-	1.5	8.982	0.000	0.000	0.000	114.09	1.32	0.19
1.5	0.5	0.5	3.832	0.000	2.923	21.529	91.72	1.54	0.26
-	-	1.0	5.311	0.000	0.094	14.385	56.20	1.72	0.24
-	-	1.5	5.619	0.000	0.089	13.970	154.17	1.60	0.22
-	1.0	0.5	5.054	0.000	5.639	11.041	645.43	2.47	0.39
-	-	1.0	8.869	0.000	0.000	0.180	108.65	1.67	0.26
-	-	1.5	10.130	0.000	0.000	0.000	55.67	0.96	0.23
-	1.5	0.5	5.429	0.000	5.102	12.378	146.32	2.25	0.19
-	-	1.0	10.057	0.000	0.000	0.000	72.69	1.81	0.34
-	-	1.5	11.946	0.000	0.000	0.000	535.94	0.97	0.21
Average				0.000	0.725	3.908	148.50	1.61	0.24

$$h = 1, c_1 = c_2 = 4, r_1 = 40, r_2 = 0, \lambda = 100, p_{0,1} = p_{0,2} = 0.5, \phi^{(1)} = \phi^{(2)} = 0.75, \bar{x} = 5.$$

We first vary experimentation and product launch rates, in Table 1. SDNCP yields the globally optimal reward in *each* of the 27 compiled instances. FNCP yields the globally optimal reward in 18 instances and CP yields the globally optimal reward in 14 instances. The average distances of FNCP and CP from optimal reward are 0.73% and 3.91%, respectively.

For FNCP and CP, the largest optimality gaps occur when the experimentation rate in stage 1 is 1.5 and the product launch rate is 0.5: If the NPD process slows down in further stages, it is more crucial to protect the system against congestion in a sophisticated manner, which can be achieved by SDNCP but not the other heuristics. Thus the optimality gaps are higher under (monotonically) decreasing rates of experimentation and product launch.

Conversely, if the NPD process speeds up in further stages it is more desirable to aggressively promote projects. The optimality gaps are therefore lower for CP under (monotonically) increasing rates of experimentation and product launch. But it is important to note that CP yields the globally optimal reward even when (i) $\mu_1 = 0.5$, $\mu_2 = 1.5$, and $\mu_3 = 0.5$, and (ii) $\mu_1 = 1.5$, $\mu_2 = 1.5$, and $\mu_3 = 1$. In case (i), new product ideas should always be tested in the initial stage as the later stages

Table 2 Numerical results for various holding and experimentation costs.

h	c_1	c_2	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
				SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.5	2	2	8.514	0.000	0.000	0.000	43.65	1.90	0.29
-	-	4	7.500	0.000	0.000	0.000	39.39	1.46	0.21
-	-	6	6.647	0.000	0.000	0.000	46.86	1.70	0.19
-	4	2	6.823	0.000	0.015	0.015	154.52	2.20	0.32
-	-	4	5.904	0.000	0.017	0.017	105.21	1.00	0.21
-	-	6	5.044	0.000	0.000	0.020	37.02	1.32	0.16
-	6	2	5.253	0.000	0.038	0.076	91.78	1.72	0.37
-	-	4	4.312	0.000	0.023	0.093	100.60	1.79	0.30
-	-	6	3.451	0.000	0.145	0.348	75.09	1.40	0.25
1.0	2	2	7.150	0.000	0.028	0.098	37.41	1.09	0.23
-	-	4	6.190	0.000	0.048	0.307	112.78	1.00	0.23
-	-	6	5.321	0.000	0.958	1.485	155.45	1.43	0.14
-	4	2	5.437	0.000	1.104	1.784	126.61	1.68	0.22
-	-	4	4.608	0.000	2.192	3.516	67.14	1.54	0.22
-	-	6	3.826	0.000	3.189	5.541	260.11	1.72	0.13
-	6	2	3.991	0.000	4.260	6.941	106.57	1.22	0.25
-	-	4	3.185	0.000	6.719	10.706	105.32	1.69	0.23
-	-	6	2.443	0.000	5.649	17.601	95.16	1.65	0.15
1.5	2	2	5.979	0.000	0.602	1.823	44.56	1.40	0.15
-	-	4	5.087	0.000	0.865	3.578	94.46	1.57	0.12
-	-	6	4.348	0.000	3.335	7.912	596.93	1.36	0.12
-	4	2	4.420	0.000	3.529	8.394	223.29	1.99	0.16
-	-	4	3.666	0.000	5.674	15.494	52.71	1.58	0.13
-	-	6	2.995	0.000	5.042	25.843	236.45	2.18	0.11
-	6	2	3.087	0.000	7.872	26.336	192.59	2.07	0.18
-	-	4	2.363	0.000	4.486	39.738	467.21	2.27	0.14
-	-	6	1.769	0.000	5.596	64.613	332.18	1.80	0.12
Average				0.000	2.274	8.973	148.19	1.62	0.20

$$r_1 = 40, r_2 = 0, \lambda = 100, \mu_1 = \mu_2 = 1, \mu_3 = p_{0,1} = p_{0,2} = 0.5, \phi^{(1)} = \phi^{(2)} = 0.75, \bar{x} = 5.$$

are no slower, and upon completion of the initial stage, projects should always be tested in the second stage as its throughput rate, which is constrained by the initial stage, is no greater than the product launch rate. In case (ii), since projects with very low expected returns are likely to be terminated prior to the product launch stage, the effective throughput rates in stages 1 and 2 are sufficiently small so that projects should always be tested in stages 1 and 2 in order to keep up with the speed of the product launch stage.

We next vary holding and experimentation costs, in Table 2. Again, SDNCP yields the globally optimal reward in *each* of the 27 compiled instances. FNCP yields the globally optimal reward in 4 instances and CP yields the globally optimal reward in 3 instances. The average distances of FNCP and CP from optimal reward are 2.27% and 8.97%, respectively.

Both FNCP and CP tend to perform worse as holding and/or experimentation costs increase: It is almost always beneficial to terminate projects with little promise under high holding and/or experimentation costs. But positive optimal rewards prove that it is also beneficial to launch new

Table 3 Numerical results for various levels of accuracy in experimental stages.

$\phi^{(1)}$	$\phi^{(2)}$	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
			SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.55	0.55	2.818	0.000	4.400	44.109	1454.90	1.03	0.22
-	0.65	2.818	0.000	4.400	40.277	237.30	1.16	0.13
-	0.75	3.053	0.033	2.588	23.125	503.42	1.38	0.19
-	0.85	3.758	0.160	1.384	9.872	185.80	2.14	0.20
-	0.95	4.776	0.670	0.775	4.271	455.60	1.65	0.22
0.65	0.55	3.295	0.000	7.284	11.897	227.07	1.35	0.15
-	0.65	3.335	0.000	7.976	12.204	200.67	2.17	0.20
-	0.75	3.423	0.000	6.924	11.861	242.89	1.10	0.24
-	0.85	3.930	0.178	2.926	4.173	174.00	1.51	0.27
-	0.95	4.885	0.143	0.655	1.269	171.46	1.76	0.24
0.75	0.55	4.594	0.000	2.133	3.418	179.51	1.15	0.12
-	0.65	4.597	0.000	2.132	3.437	208.98	1.14	0.14
-	0.75	4.608	0.000	2.192	3.516	67.19	1.50	0.24
-	0.85	4.831	0.000	0.269	0.766	188.45	1.78	0.16
-	0.95	5.552	0.000	0.018	0.108	69.94	1.63	0.20
0.85	0.55	6.078	0.000	0.559	0.888	261.50	1.11	0.15
-	0.65	6.078	0.000	0.559	0.888	126.54	1.48	0.12
-	0.75	6.078	0.000	0.559	0.888	50.80	1.28	0.15
-	0.85	6.136	0.000	0.098	0.359	30.79	1.30	0.16
-	0.95	6.574	0.000	0.000	0.046	68.34	1.90	0.15
0.95	0.55	7.640	0.000	0.039	0.118	40.19	1.48	0.16
-	0.65	7.640	0.000	0.039	0.118	40.42	0.91	0.13
-	0.75	7.640	0.000	0.039	0.118	89.70	1.50	0.17
-	0.85	7.640	0.000	0.039	0.118	60.16	1.04	0.13
-	0.95	7.739	0.000	0.013	0.065	24.65	1.66	0.16
Average			0.047	1.920	7.116	214.41	1.44	0.18

$$h = 1, c_1 = c_2 = 4, r_1 = 40, r_2 = 0, \lambda = 100, \mu_1 = \mu_2 = 1, \mu_3 = p_{0,1} = p_{0,2} = 0.5, \bar{x} = 5.$$

products from projects with sufficiently high expected returns. Those categories from which many projects are to be terminated have little impact on congestion whereas the others contribute significantly to congestion. The system is thus better protected against congestion by taking into consideration the breakdown of projects into *categories*. Also, under high holding and/or experimentation costs, notice that the benefit of a new product launch relies heavily on cost savings from significantly reduced time-to-market (or congestion), and thus protecting the system against congestion is crucial. As a result, SDNCP performs substantially better than both FNCP and CP when holding and/or experimentation costs are high.

We also observe that the performances of FNCP and CP tend to deteriorate more rapidly with an increment in experimentation cost of stage 1 than in experimentation cost of stage 2: If the experimentation cost is higher in stage 1 an investment on a new product idea has a much lower rate of return than on a project that shows promise upon completion of stage 1. Thus further caution must be taken when promoting new product ideas, by taking into account the projects not only in stage 1 but also in stage 2. Since this is not possible with FNCP and CP, their optimality

gaps are larger in this case.

Last, we vary values of $\phi^{(1)}$ and $\phi^{(2)}$, in Table 3. SDNCP yields the globally optimal reward in the 20 out of 25 compiled instances. (We examined the globally optimal policy in the instance with the largest optimality gap for SDNCP, i.e., when $\phi^{(1)} = 0.55$ and $\phi^{(2)} = 0.95$: Properties (c) and (d) of SDNCP are violated by optimal actions in several states.) FNCP yields the globally optimal reward in only one instance. The average distances of SDNCP, FNCP, and CP from optimal reward are 0.05%, 1.92%, and 7.12%, respectively.

For FNCP and CP, the largest optimality gaps occur when experimental results are *less* accurate, i.e., when $\phi^{(1)} \leq 0.65$ and $\phi^{(2)} \leq 0.75$: The NPD system has two important goals: (i) choosing the *right* projects for promotion and termination; and (ii) reducing time-to-market (or congestion). If experimental results are *more* accurate, the cost savings from (i) outweigh those from (ii). Otherwise, the performance of the system is mainly driven by the cost savings from (ii), and congestion should be reduced in a more sophisticated manner, leading to larger optimality gaps for FNCP and CP. Also note that CP performs significantly worse (by more than 25%) when $\phi^{(1)} = 0.55$ and $\phi^{(2)} = 0.65$ than when $\phi^{(1)} = 0.65$ and $\phi^{(2)} = 0.55$: Since projects with lower expected rewards in stage 1 are less likely to be terminated when $\phi^{(1)} = 0.55$, the incentive to test a new product idea is lower, and a much larger optimality gap results.

Our overall conclusion is that SDNCP substantially outperforms FNCP and CP (i) when the NPD process slows down in downstream stages, (ii) when holding and/or experimentation costs are higher, or (iii) when experiments are less reliable. In addition, CP performs significantly worse if initial testing is of little accuracy, but very expensive.

We list computation times for the heuristics in the last three columns of Tables 1–3: Computation times of CP are significantly shorter than those of SDNCP and FNCP by up to three orders and one order or magnitude, respectively. (Computation times for the global optimal solution are only slightly greater than those of CP.)

We also considered instances with larger values of m , K , and \bar{x} to draw more general conclusions about our heuristics: We could *not* solve instances for SDNCP and FNCP when $m = 3$ or $K = 3$ since the MIP solver runs out of memory. However we could solve instances for SDNCP and FNCP when $\bar{x} = 6$. Because the SDNCP computation times exceed one hour in many instances, and the basic insights about FNCP and CP when $\bar{x} = 5$ continue to hold, we relegated our numerical results when $\bar{x} = 6$ to the online appendix.

7. Concluding Remarks

We have studied the NPD portfolio selection problem under Markovian assumptions. We show the optimality of SDNCP (a) when there is a single experimental stage and projects are not terminated, or (b) when there are multiple non-experimental stages. These findings are the first to reveal the impact of congestion on optimal project promotion decisions under imperfect information and scarce resources across stages. We also prove that SDNCP outperforms both FNCP and CP, which are simpler versions of SDNCP whose thresholds are constant across states, with respect to objective value in the general problem. In addition, SDNCP yields the globally optimal reward in the vast majority of the instances in Section 6, and its average distance from the optimal reward is virtually zero. However, the average distances of FNCP and CP are 1.62% and 6.74%, respectively. The strong numerical performance of SDNCP demonstrates that project promotion decisions should be based on a broader monitoring of projects across all categories. Our numerical results also indicate that the easy-to-implement FNCP policy may be a very good choice (i) when upstream stages are slower than downstream stages, (ii) when projects have higher expected margins, or (iii) when experiments are more reliable.

An important avenue for future research is to explore the optimal policy in the general problem. Further optimality results may require alternate metrics for congestion, extending our structural results: Section 3 revealed that a stage becomes more congested with an increase in the expected reward of any project. Section 4 revealed that it becomes more desirable to promote a project in a particular stage when projects in earlier or later stages get promoted. Therefore one might intuitively expect the incentive to promote projects to be more subtle (i) if a project in any later stage is promoted to the next stage, returning a signal that increases its expected reward above a certain threshold (thus both reducing *and* increasing congestion), or (ii) if a project in any earlier stage is promoted to the next stage, returning a signal that decreases its expected reward below a certain threshold (again both increasing *and* reducing congestion).

Future research could also extend our model to phase-type or even general experimentation times. (For example, it appears straightforward to prove that SDNCP remains optimal under Erlang experimentation times when there are multiple non-experimental stages.) Another direction is to study the portfolio selection problem when different stages share a single resource. The problem could then be viewed as an MAB problem, with the states of the projects described by the categories, and thus an *index* rule may be optimal. Also, extending our Bayesian framework to

different conjugate priors is an interesting problem to pursue. Lastly, it would be more realistic to use experimental results for one project to update beliefs about other (related) projects.

References

- Adler, P. S., A. Mandelbaum, V. Nguyen, E. Schwerer. 1995. From project to process management: An empirically-based framework for analyzing product development time. *Management Science* **41** 458–484.
- Artmann, C. 2009. The value of information updating in new product development. *Lecture Notes in Economics and Mathematical Systems, Vol. 620*. Springer Berlin Heidelberg.
- Banks, J. S., R. K. Sundaram. 1994. Switching costs and the Gittins index. *Econometrica* **62** 687–694.
- Benjaafar, S., M. ElHafsi, C. Y. Lee, W. Zhou. 2011. Optimal control of an assembly system with multiple stages and multiple demand classes. *Operations Research* **59** 522–529.
- Bertsekas, D. P. 2007. *Dynamic Programming and Optimal Control, Vol. 2*. Athena Scientific, Nashua, NH.
- Bertsimas, D., J. Niño-Mora. 1996. Conservation laws, extended polymatroids and multi-armed bandit problems: A polyhedral approach to indexable systems. *Mathematics of Operations Research* **21** 257–306.
- Bhandari, A., A. Scheller-Wolf, M. Harchol-Balter. 2008. An exact and efficient algorithm for the constrained dynamic operator staffing problem for call centers. *Management Science* **54** 339–353.
- Blau, G. E., B. Mehta, S. Bose, J. F. Pekny, G. Sinclair, K. Kuenker, P. Bunch. 2000. Risk management in the development of new products in highly regulated industries. *Computers and Chemical Engineering* **24** 659–664.
- Blau, G. E., J. F. Pekny, V. A. Varma, P. R. Bunch. 2004. Managing a portfolio of interdependent new product candidates in the pharmaceutical industry. *Journal of Product Innovation Management* **21** 227–245.
- Brucker, P., A. Drexler, R. Möhring, K. Neumann, E. Pesch. 1999. Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research* **112** 3–41.
- Cooper, R. G. 2008. Perspective: The stage-gate idea-to-launch process – update, what’s new, and NexGen systems. *Journal of Product Innovation Management* **25** 213–232.
- DiMasi, J. A., R. W. Hansen, H. G. Grabowski. 2003. The price of innovation: New estimates of drug development costs. *Journal of Health Economics* **22** 151–185.
- Ding, M., J. Eliashberg. 2002. Structuring the new product development pipeline. *Management Science* **48** 343–363.
- Gino, F., and G. Pisano. 2005. Holding or folding? R&D portfolio strategy under different information regimes. Harvard Business School Working Paper.

- Girotra, K., C. Terwiesch, K. T. Ulrich. 2007. Valuing R&D projects in a portfolio: Evidence from the pharmaceutical industry. *Management Science* **53** 1452–1466.
- Gittins, J. C., D. M. Jones. 1972. A dynamic allocation index for the sequential design of experiments. *Progress in Statistics: European Meeting of Statisticians*.
- Gittins, J. C. 1979. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society. Series B (Methodological)* **41** 148–177.
- Gittins, J. C. 1989. *Multi-armed Bandit Allocation Indices*. John Wiley, New York, NY.
- Ha, A. 1997a. Optimal dynamic scheduling policy for a make-to-stock production system. *Operations Research* **45** 42–53.
- Ha, A. 1997b. Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Management Science* **43** 1093–1103.
- Harrison, J. M. 1975. Dynamic scheduling of a multiclass queue: Discount optimality. *Operations Research* **23** 270–282.
- Haviv, M., M. L. Puterman. 1998. Bias optimality in controlled queueing systems. *Journal of Applied Probability* **35** 136–150.
- Kavadias, S., C. H. Loch. 2003. Optimal project sequencing with recourse at a scarce resource. *Production and Operations Management* **12** 433–444.
- Kavadias, S., C. H. Loch. 2004. *Project Selection Under Uncertainty: Dynamically Allocating Resources to Maximize Value*. Kluwer Academic Publishers, Boston, MA.
- Kavadias, S., R. O. Chao. 2008. Resource allocation and new product development portfolio management. C. H. Loch, S. Kavadias, eds. *Handbook of New Product Development Management*. Elsevier, Oxford, UK.
- Kleywegt, A. J., J. D. Papastavrou. 1998. The dynamic and stochastic knapsack problem. *Operations Research* **46** 17–35.
- Kleywegt, A. J., J. D. Papastavrou. 2001. The dynamic and stochastic knapsack problem with random sized items. *Operations Research* **49** 26–41.
- Lewis, M. E., H. Ayhan, R. D. Foley. 1999. Bias optimality in a queue with admission control. *Probability in the Engineering and Informational Sciences* **13** 309–327.
- Lewis, M. E., M. L. Puterman. 2000. A note on bias optimality in controlled queueing systems. *Journal of Applied Probability* **37** 300–305.
- Lippman, S. A., S. M. Ross. 1971. The streetwalker's dilemma: A job shop model. *SIAM Journal on Applied Mathematics* **20** 336–342.

- Lippman, S. A. 1975. Applying a new device in the optimization of exponential queueing systems. *Operations Research* **23** 687–710.
- Loch, C. H., C. Terwiesch. 1999. Accelerating the process of engineering change orders: Capacity and congestion effects. *Journal of Product Innovation Management* **16** 145–159.
- Mandelbaum, A., A. L. Stolyar. 2004. Scheduling flexible servers with convex delay costs: Heavy-traffic optimality of the generalized $c\mu$ -rule. *Operations Research* **52** 836–855.
- Miller, B. L. 1969. A queueing reward system with several customer classes. *Management Science* **16** 234–245.
- Nadar, E., M. Akan, A. Scheller-Wolf. 2015. Experimental results indicating lattice-dependent policies may be optimal for general assemble-to-order systems. *Production and Operations Management* Forthcoming.
- Papastavrou, J. D., S. Rajagopalan, A. J. Kleywegt. 1996. The dynamic and stochastic knapsack problem with deadlines. *Management Science* **42** 1706–1718.
- Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons, New York, NY.
- Repenning, N. P. 2001. Understanding fire fighting in new product development. *Journal of Product Innovation Management* **18** 285–300.
- Roberts, K., M. L. Weitzman. 1981. Funding criteria for research, development, and exploration projects. *Econometrica* **49** 1261–1288.
- Ross, S. M. 1982. *Introduction to Stochastic Dynamic Programming*. John Wiley, New York, NY.
- Sommer, S. C., C. H. Loch, M. T. Pich. 2008. Project risk management in new product development. C. H. Loch, S. Kavadias, eds. *Handbook of New Product Development Management*. Elsevier, Oxford, UK.
- Stidham, S. 1985. Optimal control of admission to a queueing system. *IEEE Transactions on Automatic Control* **30** 705–713.
- Subramanian, D., J. F. Pekny, G. V. Reklaitis, G. E. Blau. 2003. Simulationoptimization framework for stochastic optimization of R&D pipeline management. *AIChE Journal* **49** 96–112.
- Thomke, S. 2008. Learning by experimentation: Prototyping and testing. C. H. Loch, S. Kavadias, eds. *Handbook of New Product Development Management*. Elsevier, Oxford, UK.
- Thomke, S., D. Reinertsen. 2012. Six myths of product development. *Harvard Business Review* **90** 84–94.
- Ulrich, K., S. D. Eppinger. 2003. *Product Design and Development*. Irwin-McGraw Hill, New York, NY.
- van Mieghem, J. A. 1995. Dynamic scheduling with convex delay costs: The generalized $c\mu$ rule. *The Annals of Applied Probability* **5** 809–833.
- van Mieghem, J. A. 2000. Price and service discrimination in queueing systems: Incentive compatibility of $Gc\mu$ scheduling. *Management Science* **46** 1249–1267.

- Verma, D., A. Mishra, K. S. Sinha. 2011. The development and application of a process model for R&D project management in a high tech firm: A field study. *Journal of Operations Management* **29** 462–476.
- Vishwanath, T. 1992. Optimal orderings for parallel project selection. *International Economic Review* **33** 79–89.
- Weber, R. 1992. On the Gittins index for multiarmed bandits. *The Journal of Applied Probability* **2** 1024–1033.
- Wein, L. M. 1992. Dynamic scheduling of a multiclass make-to-stock queue. *Operations Research* **40** 724–735.
- Whittle, P. 1980. Multi-armed bandits and the Gittins index. *Journal of the Royal Statistical Society. Series B (Methodological)* **42** 143–149.
- Whittle, P. 1981. Arm-acquiring bandits. *The Annals of Probability* **9** 284–292.
- Whittle, P. 1988. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability* **25** 287–298.

Online Technical Appendix

EC.1. Existence of the Optimality Equation

Suppose that Assumptions 6.10.1 and 6.10.2 in Puterman (1994) hold. Then, by Theorem 6.10.4 in Puterman (1994), we can establish the existence of a solution to the optimality equation and the convergence of value iteration. We below show that Assumptions 6.10.1 and 6.10.2 hold for our NPD model in Section 2. Recall that the holding cost per unit time ($h(\mathbf{x})$) is convex and strictly increasing in the total number of projects. Let $\bar{\rho}$ denote the maximum expected reward that can be obtained from a product launch, and $r(\mathbf{x}, \mathbf{u})$ denote the reward per stage when the system is in state \mathbf{x} and action \mathbf{u} is chosen.

(1) We can easily verify that Assumption 6.10.1 holds:

$$\sup_{\mathbf{u} \in \mathcal{U}} |r(\mathbf{x}, \mathbf{u})| \leq \bar{\rho} + \sum_i c_i + h(\mathbf{x}) = \bar{\rho} + \sum_i c_i + h\left(\sum_i x_i\right) = w(\mathbf{x}).$$

(2) Next we want to prove that Assumption 6.10.2 holds. Suppose that a new project is launched in state \mathbf{x} at the beginning of period k . Thus the discounted holding cost in period k increases by $h(\mathbf{x} + e_1) - h(\mathbf{x})$ (recall that without loss of generality we assume $\alpha + \nu = 1$). The return from this project can be earned no earlier than the beginning of period $k + 1$. Thus the total discounted expected return (from all active projects) cannot increase by more than $\nu\bar{\rho}$, i.e., the discounted value of the maximum expected reward obtained at the beginning of period $k + 1$. For all the projects included in state \mathbf{x} , moving from \mathbf{x} to $\mathbf{x} + e_1$ does not change the action space, or any of the transition probabilities for actions then or in the future. We conclude that if $h(\mathbf{x} + e_1) - h(\mathbf{x}) > \nu\bar{\rho}$, it is not optimal to launch a new project in state \mathbf{x} .

(a) Consider the states \mathbf{x} for which $h(\mathbf{x} + e_1) - h(\mathbf{x}) \leq \nu\bar{\rho}$. Notice:

$$\begin{aligned} \sum_{\mathbf{y}} P_{\mathbf{x},\mathbf{y}}(\mathbf{u})w(\mathbf{y}) &= \sum_{\mathbf{y}} P_{\mathbf{x},\mathbf{y}}(\mathbf{u}) \left(\bar{\rho} + \sum_i c_i + h(\mathbf{y}) \right) \\ &= \sum_{\mathbf{y}} P_{\mathbf{x},\mathbf{y}}(\mathbf{u}) \left(\bar{\rho} + \sum_i c_i \right) + \sum_{\mathbf{y}} P_{\mathbf{x},\mathbf{y}}(\mathbf{u})h(\mathbf{y}) \\ &= \bar{\rho} + \sum_i c_i + \sum_{\mathbf{y}} P_{\mathbf{x},\mathbf{y}}(\mathbf{u})h(\mathbf{y}) \\ &\leq \bar{\rho} + \sum_i c_i + h(\mathbf{x} + e_1) \\ &= w(\mathbf{x}) + h(\mathbf{x} + e_1) - h(\mathbf{x}) \\ &\leq w(\mathbf{x}) + \nu\bar{\rho}. \end{aligned}$$

- (b) Consider the states \mathbf{x} for which $h(\mathbf{x} + e_1) - h(\mathbf{x}) > \nu\bar{\rho}$. Without loss of optimality we assume that the system never moves from state \mathbf{x} to $\mathbf{x} + e_1$ under any action \mathbf{u} , i.e., $P_{\mathbf{x}, \mathbf{x} + e_1}(\mathbf{u}) = 0, \forall \mathbf{u}$. Notice:

$$\begin{aligned}
\sum_{\mathbf{y}} P_{\mathbf{x}, \mathbf{y}}(\mathbf{u}) w(\mathbf{y}) &= \sum_{\mathbf{y}} P_{\mathbf{x}, \mathbf{y}}(\mathbf{u}) \left(\bar{\rho} + \sum_i c_i + h(\mathbf{y}) \right) \\
&= \sum_{\mathbf{y}} P_{\mathbf{x}, \mathbf{y}}(\mathbf{u}) \left(\bar{\rho} + \sum_i c_i \right) + \sum_{\mathbf{y}} P_{\mathbf{x}, \mathbf{y}}(\mathbf{u}) h(\mathbf{y}) \\
&= \bar{\rho} + \sum_i c_i + \sum_{\mathbf{y}} P_{\mathbf{x}, \mathbf{y}}(\mathbf{u}) h(\mathbf{y}) \\
&\leq \bar{\rho} + \sum_i c_i + h(\mathbf{x}) \\
&= w(\mathbf{x}) \\
&\leq w(\mathbf{x}) + \nu\bar{\rho}.
\end{aligned}$$

Hence, following Proposition 6.10.5 in Puterman (1994), we conclude that Assumption 6.10.2 holds.

EC.2. Proofs of the Results in Section 3

Proof of Lemma 1. Define \widehat{V} as the set of real-valued functions on \mathbb{N}_0^n that satisfy Properties 1–7. Also, define the operator T on the set of real-valued functions v as follows: $Tv(\mathbf{x}) = \mu_1 T_1 v(\mathbf{x}) + \mu_2 T_2 v(\mathbf{x}) - h(\mathbf{x})$. We below show that $T_1 : \widehat{V} \rightarrow \widehat{V}$, $T_2 : \widehat{V} \rightarrow \widehat{V}$, and $-h \in \widehat{V}$. We will then prove that $T : \widehat{V} \rightarrow \widehat{V}$.

$T_1 : \widehat{V} \rightarrow \widehat{V}$. Assume that v satisfies Properties 1–7. We below show $T_1 v$ satisfies Properties 1–7.

Property 1. We will prove $T_1 v$ satisfies Property 1, i.e., $T_1 v(\mathbf{x} + e_w) \leq T_1 v(\mathbf{x} + e_q), \forall (q, w)$ s.t. $0 < q < w$. We consider the following two scenarios depending on the optimal action at $T_1 v(\mathbf{x} + e_w)$ (if this inequality holds under a suboptimal action of $T_1 v(\mathbf{x} + e_q)$, it also holds under the optimal action of this operator, and thus we do not enforce the optimal action at this operator):

- (1) Suppose that $T_1 v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w)$. As we assume v satisfies Property 1, the following inequalities hold: $T_1 v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w) \leq v(\mathbf{x} + e_q) \leq T_1 v(\mathbf{x} + e_q)$.
- (2) Suppose that $T_1 v(\mathbf{x} + e_w) = \sum_j v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - c$. As we assume v satisfies Property 1, the following inequalities hold: $T_1 v(\mathbf{x} + e_w) = \sum_j v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - c \leq \sum_j v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - c \leq T_1 v(\mathbf{x} + e_q)$.

Therefore $T_1 v$ satisfies Property 1.

Property 2. We will prove $T_1 v$ satisfies Property 2, i.e., $T_1 v(\mathbf{x} + e_w) \leq T_1 v(\mathbf{x}) + \rho_w, \forall w > 0$. We consider the following two scenarios depending on the optimal action at $T_1 v(\mathbf{x} + e_w)$:

- (1) Suppose that $T_1v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w)$. As we assume v satisfies Property 2, the following inequalities hold: $T_1v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w) \leq v(\mathbf{x}) + \rho_w \leq T_1v(\mathbf{x}) + \rho_w$.
- (2) Suppose that $T_1v(\mathbf{x} + e_w) = \sum_j v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - c$. As we assume v satisfies Property 2, the following inequalities hold: $T_1v(\mathbf{x} + e_w) = \sum_j v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - c \leq \sum_j (v(\mathbf{x} + e_j) + \rho_w) f_{0 \rightarrow j} - c = \sum_j v(\mathbf{x} + e_j) f_{0 \rightarrow j} - c + \rho_w \leq T_1v(\mathbf{x}) + \rho_w$.

Therefore T_1v satisfies Property 2.

Property 3. We will prove T_1v satisfies Property 3, i.e., $T_1v(\mathbf{x} + e_q) + \rho_w \leq T_1v(\mathbf{x} + e_w) + \rho_q$, $\forall(q, w)$ s.t. $0 < q < w$. We consider the following two scenarios depending on the optimal action at $T_1v(\mathbf{x} + e_q)$:

- (1) Suppose that $T_1v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q)$. As we assume v satisfies Property 3, the following inequalities hold: $T_1v(\mathbf{x} + e_q) + \rho_w = v(\mathbf{x} + e_q) + \rho_w \leq v(\mathbf{x} + e_w) + \rho_q \leq T_1v(\mathbf{x} + e_w) + \rho_q$.
- (2) Suppose that $T_1v(\mathbf{x} + e_q) = \sum_j v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - c$. As we assume v satisfies Property 3, the following inequalities hold: $T_1v(\mathbf{x} + e_q) + \rho_w = \sum_j v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - c + \rho_w \leq \sum_j v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - c + \rho_q \leq T_1v(\mathbf{x} + e_w) + \rho_q$.

Therefore T_1v satisfies Property 3.

Property 4. We will prove T_1v satisfies Property 4, i.e., $T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \leq T_1v(\mathbf{x} + e_l) - T_1v(\mathbf{x} + e_d)$, $\forall(d, l, q)$ s.t. $0 < d \leq l \leq q$. We consider the following scenarios depending on the optimal actions at $T_1v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_d)$ (if this inequality holds under suboptimal actions of $T_1v(\mathbf{x} + e_q + e_d)$ and/or $T_1v(\mathbf{x} + e_l)$, it also holds under optimal actions of these operators, and thus we do not enforce the optimal actions at these operators):

- (1) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_d) = v(\mathbf{x} + e_d)$. As we assume v satisfies Property 4, $T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \leq v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_d) \leq v(\mathbf{x} + e_l) - v(\mathbf{x} + e_d) \leq T_1v(\mathbf{x} + e_l) - T_1v(\mathbf{x} + e_d)$.
- (2) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_d) = \sum_j v(\mathbf{x} + e_d + e_j) f_{0 \rightarrow j} - c$. As we assume v satisfies Properties 4 and 7, the following inequalities hold:

$$\begin{aligned}
T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) &\leq v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_d) \\
&\leq v(\mathbf{x} + e_l) - v(\mathbf{x} + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_l + e_j) f_{0 \rightarrow j} - c - \sum_j v(\mathbf{x} + e_d + e_j) f_{0 \rightarrow j} + c \\
&\leq T_1v(\mathbf{x} + e_l) - T_1v(\mathbf{x} + e_d)
\end{aligned}$$

- (3) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = \sum_j v(\mathbf{x} + e_q + e_l + e_j)f_{0 \rightarrow j} - c$ and $T_1v(\mathbf{x} + e_d) = v(\mathbf{x} + e_d)$. As we assume v satisfies Properties 4 and 6, the following inequalities hold:

$$\begin{aligned}
& T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \\
& \leq \sum_j v(\mathbf{x} + e_q + e_l + e_j)f_{0 \rightarrow j} - c - v(\mathbf{x} + e_q + e_d) \\
& \leq \sum_j v(\mathbf{x} + e_q + e_l + e_j)f_{0 \rightarrow j} - c - v(\mathbf{x} + e_q + e_l) + v(\mathbf{x} + e_l) - v(\mathbf{x} + e_d) \\
& \leq \sum_j v(\mathbf{x} + e_l + e_j)f_{0 \rightarrow j} - c - v(\mathbf{x} + e_d) \\
& \leq T_1v(\mathbf{x} + e_l) - T_1v(\mathbf{x} + e_d)
\end{aligned}$$

- (4) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = \sum_j v(\mathbf{x} + e_q + e_l + e_j)f_{0 \rightarrow j} - c$ and $T_1v(\mathbf{x} + e_d) = \sum_j v(\mathbf{x} + e_d + e_j)f_{0 \rightarrow j} - c$. As we assume v satisfies Property 4, the following inequalities hold:

$$\begin{aligned}
& T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \\
& \leq \sum_j v(\mathbf{x} + e_q + e_l + e_j)f_{0 \rightarrow j} - c - \sum_j v(\mathbf{x} + e_q + e_d + e_j)f_{0 \rightarrow j} + c \\
& = \sum_j (v(\mathbf{x} + e_q + e_l + e_j) - v(\mathbf{x} + e_q + e_d + e_j))f_{0 \rightarrow j} \\
& \leq \sum_j (v(\mathbf{x} + e_l + e_j) - v(\mathbf{x} + e_d + e_j))f_{0 \rightarrow j} \\
& = \sum_j v(\mathbf{x} + e_l + e_j)f_{0 \rightarrow j} - c - \sum_j v(\mathbf{x} + e_d + e_j)f_{0 \rightarrow j} + c \\
& \leq T_1v(\mathbf{x} + e_l) - T_1v(\mathbf{x} + e_d)
\end{aligned}$$

Therefore T_1v satisfies Property 4.

Property 5. We will prove T_1v satisfies Property 5, i.e., $T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \leq T_1v(\mathbf{x} + e_w + e_l) - T_1v(\mathbf{x} + e_w + e_d)$, $\forall(d, l, q, w)$ s.t. $0 < d \leq l \leq q \leq w$. We consider the following scenarios depending on the optimal actions at $T_1v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_w + e_d)$:

- (1) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_w + e_d) = v(\mathbf{x} + e_w + e_d)$. As we assume v satisfies Property 5, $T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \leq v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_d) \leq v(\mathbf{x} + e_w + e_l) - v(\mathbf{x} + e_w + e_d) \leq T_1v(\mathbf{x} + e_w + e_l) - T_1v(\mathbf{x} + e_w + e_d)$.
- (2) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_w + e_d) = \sum_j v(\mathbf{x} + e_w + e_d + e_j)f_{0 \rightarrow j} - c$. As we assume v satisfies Properties 5 and 7, the following inequalities hold:

$$T_1v(\mathbf{x} + e_q + e_l) - T_1v(\mathbf{x} + e_q + e_d) \leq v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_d)$$

$$\begin{aligned}
&\leq v(\mathbf{x} + e_w + e_l) - v(\mathbf{x} + e_w + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_w + e_l + e_j) f_{0 \rightarrow j} - \sum_j v(\mathbf{x} + e_w + e_d + e_j) f_{0 \rightarrow j} \\
&\leq T_1 v(\mathbf{x} + e_w + e_l) - T_1 v(\mathbf{x} + e_w + e_d)
\end{aligned}$$

(3) Suppose that $T_1 v(\mathbf{x} + e_q + e_l) = \sum_j v(\mathbf{x} + e_q + e_l + e_j) f_{0 \rightarrow j} - c$ and $T_1 v(\mathbf{x} + e_w + e_d) = v(\mathbf{x} + e_w + e_d)$. As we assume v satisfies Properties 5 and 7, the following inequalities hold:

$$\begin{aligned}
&T_1 v(\mathbf{x} + e_q + e_l) - T_1 v(\mathbf{x} + e_q + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_q + e_l + e_j) f_{0 \rightarrow j} - c - v(\mathbf{x} + e_q + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_q + e_l + e_j) f_{0 \rightarrow j} - c - v(\mathbf{x} + e_q + e_l) + v(\mathbf{x} + e_w + e_l) - v(\mathbf{x} + e_w + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_w + e_l + e_j) f_{0 \rightarrow j} - c - v(\mathbf{x} + e_w + e_d) \\
&\leq T_1 v(\mathbf{x} + e_w + e_l) - T_1 v(\mathbf{x} + e_w + e_d)
\end{aligned}$$

(4) Suppose that $T_1 v(\mathbf{x} + e_q + e_l) = \sum_j v(\mathbf{x} + e_q + e_l + e_j) f_{0 \rightarrow j} - c$ and $T_1 v(\mathbf{x} + e_w + e_d) = \sum_j v(\mathbf{x} + e_w + e_d + e_j) f_{0 \rightarrow j} - c$. As we assume v satisfies Property 5, the following inequalities hold:

$$\begin{aligned}
&T_1 v(\mathbf{x} + e_q + e_l) - T_1 v(\mathbf{x} + e_q + e_d) \\
&\leq \sum_j v(\mathbf{x} + e_q + e_l + e_j) f_{0 \rightarrow j} - c - \sum_j v(\mathbf{x} + e_q + e_d + e_j) f_{0 \rightarrow j} + c \\
&= \sum_j (v(\mathbf{x} + e_q + e_l + e_j) - v(\mathbf{x} + e_q + e_d + e_j)) f_{0 \rightarrow j} \\
&\leq \sum_j (v(\mathbf{x} + e_w + e_l + e_j) - v(\mathbf{x} + e_w + e_d + e_j)) f_{0 \rightarrow j} \\
&= \sum_j v(\mathbf{x} + e_w + e_l + e_j) f_{0 \rightarrow j} - c - \sum_j v(\mathbf{x} + e_w + e_d + e_j) f_{0 \rightarrow j} + c \\
&\leq T_1 v(\mathbf{x} + e_w + e_l) - T_1 v(\mathbf{x} + e_w + e_d)
\end{aligned}$$

Therefore $T_1 v$ satisfies Property 5.

Property 6. We will prove $T_1 v$ satisfies Property 6, i.e., $\sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x})$, $\forall (l, q)$ s.t. $0 < l \leq q$. We consider the following scenarios depending on the optimal actions at $T_1 v(\mathbf{x} + e_q + e_l)$, $T_1 v(\mathbf{x} + e_q + e_j)$ for $j > l$, and $T_1 v(\mathbf{x})$ (if this inequality holds under suboptimal actions of $T_1 v(\mathbf{x} + e_q)$, $T_1 v(\mathbf{x} + e_l)$, and/or $T_1 v(\mathbf{x} + e_j)$ for $j > l$, it also holds under optimal actions of these operators, and thus we do not enforce the optimal actions at these operators):

- (1) Suppose that $T_1v(\mathbf{x}) = v(\mathbf{x})$. As we assume v satisfies Property 6, $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_q + e_j) = v(\mathbf{x} + e_q + e_j)$ for $j > l$. As we assume v satisfies Property 6, the following inequalities hold:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_j)f_{0 \rightarrow j} - v(\mathbf{x}) \\
& \leq \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x})
\end{aligned}$$

- (2) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$, $T_1v(\mathbf{x} + e_q + e_j) = v(\mathbf{x} + e_q + e_j)$ for $j^* > j > l$, $T_1v(\mathbf{x} + e_q + e_j) = \sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c$ for $j \geq j^*$, and $T_1v(\mathbf{x}) = \sum_{j'} v(\mathbf{x} + e_{j'})f_{0 \rightarrow j'} - c$. As we assume v satisfies Properties 4 and 6, the following inequalities hold:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{l < j < j^*} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
& - \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{l < j < j^*} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + c \\
& = \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
& - \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + c \\
& = \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_q + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j} \\
& \leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_l) - v(\mathbf{x} + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_q + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j} \\
& \leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_l) - v(\mathbf{x} + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j}
\end{aligned}$$

$$\begin{aligned}
&= \sum_{1 \leq j \leq l} v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{1 \leq j \leq l} v(\mathbf{x} + e_j) f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_j) f_{0 \rightarrow j} + c \\
&= \sum_{1 \leq j \leq l} v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{l < j < j^*} v(\mathbf{x} + e_j) f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{1 \leq j \leq l} v(\mathbf{x} + e_j) f_{0 \rightarrow j} - \sum_{l < j < j^*} v(\mathbf{x} + e_j) f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_j) f_{0 \rightarrow j} + c \\
&\leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x})
\end{aligned}$$

(3) Suppose that $T_1 v(\mathbf{x} + e_q + e_l) = \sum_{j'} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j'} - c$. As we assume v satisfies Property 7, $T_1 v(\mathbf{x} + e_q + e_j) = \sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j'} - c$ for $j > l$. As we assume v satisfies Property 6, $T_1 v(\mathbf{x}) = \sum_{j'} v(\mathbf{x} + e_{j'}) f_{0 \rightarrow j'} - c$. As we assume v satisfies Property 6, the following inequalities hold:

$$\begin{aligned}
&\sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_q) \\
&\leq \sum_{1 \leq j \leq l} \left(\sum_{j'} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} + \sum_{n \geq j > l} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{j'} v(\mathbf{x} + e_q + e_{j'}) f_{0 \rightarrow j'} + c \\
&= \sum_{j'} \left(\sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j} - v(\mathbf{x} + e_q + e_{j'}) \right) f_{0 \rightarrow j'} \\
&\leq \sum_{j'} \left(\sum_{1 \leq j \leq l} v(\mathbf{x} + e_l + e_{j'}) f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_j + e_{j'}) f_{0 \rightarrow j} - v(\mathbf{x} + e_{j'}) \right) f_{0 \rightarrow j'} \\
&= \sum_{1 \leq j \leq l} \left(\sum_{j'} v(\mathbf{x} + e_l + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} + \sum_{n \geq j > l} \left(\sum_{j'} v(\mathbf{x} + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{j'} v(\mathbf{x} + e_{j'}) f_{0 \rightarrow j'} + c \\
&\leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x})
\end{aligned}$$

Therefore $T_1 v$ satisfies Property 6.

Property 7. We will prove $T_1 v$ satisfies Property 7, i.e., $\sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_w)$, $\forall (l, q, w)$ s.t. $0 < l \leq q < w$. We consider the following scenarios depending on the optimal actions at $T_1 v(\mathbf{x} + e_q + e_l)$, $T_1 v(\mathbf{x} + e_q + e_j)$ for $j > l$, and $T_1 v(\mathbf{x} + e_w)$:

- (1) Suppose that $T_1v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w)$. As we assume v satisfies Properties 6 and 7, $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$ and $T_1v(\mathbf{x} + e_q + e_j) = v(\mathbf{x} + e_q + e_j)$ for $j > l$. As we assume v satisfies Property 7, the following inequalities hold:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_w + e_j)f_{0 \rightarrow j} - v(\mathbf{x} + e_w) \\
& \leq \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_w + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_w + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x} + e_w)
\end{aligned}$$

- (2) Suppose that $T_1v(\mathbf{x} + e_q + e_l) = v(\mathbf{x} + e_q + e_l)$, $T_1v(\mathbf{x} + e_q + e_j) = v(\mathbf{x} + e_q + e_j)$ for $j^* > j > l$, $T_1v(\mathbf{x} + e_q + e_j) = \sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c$ for $j \geq j^*$, and $T_1v(\mathbf{x} + e_w) = \sum_{j'} v(\mathbf{x} + e_w + e_{j'})f_{0 \rightarrow j'} - c$. As we assume v satisfies Properties 5 and 7, the following inequalities hold:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_1v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_1v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{l < j < j^*} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
& - \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{l < j < j^*} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + c \\
& = \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
& - \sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} + c \\
& = \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q + e_l) - v(\mathbf{x} + e_q + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_q + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j} \\
& \leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_w + e_l) - v(\mathbf{x} + e_w + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_q + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j} \\
& \leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_w + e_l) - v(\mathbf{x} + e_w + e_j)) f_{0 \rightarrow j} \\
& + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_w + e_j + e_{j'})f_{0 \rightarrow j'} - v(\mathbf{x} + e_w + e_j) \right) f_{0 \rightarrow j} + c \sum_{j < j^*} f_{0 \rightarrow j}
\end{aligned}$$

$$\begin{aligned}
&= \sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_w + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} + c \\
&= \sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{l < j < j^*} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} + \sum_{j^* \leq j} \left(\sum_{j'} v(\mathbf{x} + e_w + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - \sum_{l < j < j^*} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - \sum_{j^* \leq j} v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} + c \\
&\leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_w)
\end{aligned}$$

- (3) Suppose that $T_1 v(\mathbf{x} + e_q + e_l) = \sum_{j'} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j'} - c$. As we assume v satisfies Property 7, $T_1 v(\mathbf{x} + e_q + e_j) = \sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j'} - c$ for $j > l$. As we assume v satisfies Properties 6 and 7, $T_1 v(\mathbf{x} + e_w) = \sum_{j'} v(\mathbf{x} + e_w + e_{j'}) f_{0 \rightarrow j'} - c$. As we assume v satisfies Property 7, the following inequalities hold:

$$\begin{aligned}
&\sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_q) \\
&\leq \sum_{1 \leq j \leq l} \left(\sum_{j'} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} + \sum_{n \geq j > l} \left(\sum_{j'} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{j'} v(\mathbf{x} + e_q + e_{j'}) f_{0 \rightarrow j'} + c \\
&= \sum_{j'} \left(\sum_{1 \leq j \leq l} v(\mathbf{x} + e_q + e_l + e_{j'}) f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_q + e_j + e_{j'}) f_{0 \rightarrow j} - v(\mathbf{x} + e_q + e_{j'}) \right) f_{0 \rightarrow j'} \\
&\leq \sum_{j'} \left(\sum_{1 \leq j \leq l} v(\mathbf{x} + e_w + e_l + e_{j'}) f_{0 \rightarrow j} + \sum_{n \geq j > l} v(\mathbf{x} + e_w + e_j + e_{j'}) f_{0 \rightarrow j} - v(\mathbf{x} + e_w + e_{j'}) \right) f_{0 \rightarrow j'} \\
&= \sum_{1 \leq j \leq l} \left(\sum_{j'} v(\mathbf{x} + e_w + e_l + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} + \sum_{n \geq j > l} \left(\sum_{j'} v(\mathbf{x} + e_w + e_j + e_{j'}) f_{0 \rightarrow j'} - c \right) f_{0 \rightarrow j} \\
&- \sum_{j'} v(\mathbf{x} + e_w + e_{j'}) f_{0 \rightarrow j'} + c \\
&\leq \sum_{1 \leq j \leq l} T_1 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_1 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_1 v(\mathbf{x} + e_w)
\end{aligned}$$

Therefore $T_1 v$ satisfies Property 7. Hence $T_1 v$ satisfies Properties 1–7, i.e., $T_1 : \widehat{V} \rightarrow \widehat{V}$.

$T_2 : \widehat{V} \rightarrow \widehat{V}$. Assume that v satisfies Properties 1–7. We below show $T_2 v$ satisfies Properties 1–7.

Property 1. We will prove $T_2 v$ satisfies Property 1, i.e., $T_2 v(\mathbf{x} + e_w) \leq T_2 v(\mathbf{x} + e_q)$, $\forall (q, w)$ s.t. $0 < q < w$. As we assume v satisfies Properties 2 and 3, it is always optimal to launch a new product by choosing a project with highest expected reward: $T_2 v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w - e_b) + \rho_b$ where b is the smallest j such that $x_j + I_{j=w} \geq 1$ ($I_{j=w} = 1$ if $j = w$, and $I_{j=w} = 0$ otherwise). Suppose that $b = w$.

Then it is easy to verify that $T_2v(\mathbf{x} + e_w) = v(\mathbf{x}) + \rho_w \leq v(\mathbf{x}) + \rho_q \leq T_2v(\mathbf{x} + e_q)$. Now suppose that $b \neq w$. As we assume v satisfies Property 1, $T_2v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w - e_b) + \rho_b \leq v(\mathbf{x} + e_q - e_b) + \rho_b \leq T_2v(\mathbf{x} + e_q)$. Therefore T_2v satisfies Property 1.

Property 2. We will prove T_2v satisfies Property 2, i.e., $T_2v(\mathbf{x} + e_w) \leq T_2v(\mathbf{x}) + \rho_w, \forall w > 0$. As we assume v satisfies Properties 2 and 3, $T_2v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w - e_b) + \rho_b$ where b is the smallest j such that $x_j + I_{j=w} \geq 1$. Suppose that $b = w$. Then it is easy to verify that $T_2v(\mathbf{x} + e_w) = v(\mathbf{x}) + \rho_w \leq T_2v(\mathbf{x}) + \rho_w$. Now suppose that $b \neq w$. As we assume v satisfies Property 2, $T_2v(\mathbf{x} + e_w) = v(\mathbf{x} + e_w - e_b) + \rho_b \leq v(\mathbf{x} - e_b) + \rho_b + \rho_w \leq T_2v(\mathbf{x}) + \rho_w$. Therefore T_2v satisfies Property 2.

Property 3. We will prove T_2v satisfies Property 3, i.e., $T_2v(\mathbf{x} + e_q) + \rho_w \leq T_2v(\mathbf{x} + e_w) + \rho_q, \forall (q, w)$ s.t. $0 < q < w$. As we assume v satisfies Properties 2 and 3, $T_2v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_b) + \rho_b$ where b is the smallest j such that $x_j + I_{j=q} \geq 1$. Suppose that $b = q$. Then it is easy to verify that $T_2v(\mathbf{x} + e_q) + \rho_w = v(\mathbf{x}) + \rho_q + \rho_w \leq T_2v(\mathbf{x} + e_w) + \rho_q$. Now suppose that $b \neq q$. As we assume v satisfies Property 3, $T_2v(\mathbf{x} + e_q) + \rho_w = v(\mathbf{x} + e_q - e_b) + \rho_b + \rho_w \leq v(\mathbf{x} + e_w - e_b) + \rho_b + \rho_q \leq T_2v(\mathbf{x} + e_w) + \rho_q$. Therefore T_2v satisfies Property 3.

Property 4. We will prove T_2v satisfies Property 4, i.e., $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) \leq T_2v(\mathbf{x} + e_l) - T_2v(\mathbf{x} + e_d), \forall (d, l, q)$ s.t. $0 < d \leq l \leq q$. Recall that as we assume v satisfies Properties 2 and 3, it is always optimal to launch a new product by choosing a project with highest expected reward. We consider the following scenarios:

- (1) Suppose that $x_j = 0, \forall j \leq l$. Then $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q) + \rho_l - v(\mathbf{x} + e_q) - \rho_d = v(\mathbf{x}) + \rho_l - v(\mathbf{x}) - \rho_d = T_2v(\mathbf{x} + e_l) - T_2v(\mathbf{x} + e_d)$.
- (2) Suppose that $x_j = 0, \forall j \leq d$, and $x_j \geq 1, \exists j \in \{d + 1, d + 2, \dots, l\}$. Define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 4, $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q + e_l - e_b) + \rho_b - v(\mathbf{x} + e_q) - \rho_d \leq v(\mathbf{x} + e_l - e_b) + \rho_b - v(\mathbf{x}) - \rho_d = T_2v(\mathbf{x} + e_l) - T_2v(\mathbf{x} + e_d)$.
- (3) Suppose that $x_j \geq 1, \exists j \leq d$. Again define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 4, $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q + e_l - e_b) + \rho_b - v(\mathbf{x} + e_q + e_d - e_b) - \rho_b \leq v(\mathbf{x} + e_l - e_b) + \rho_b - v(\mathbf{x} + e_d - e_b) - \rho_b = T_2v(\mathbf{x} + e_l) - T_2v(\mathbf{x} + e_d)$.

Therefore T_2v satisfies Property 4.

Property 5. We will prove T_2v satisfies Property 5, i.e., $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) \leq T_2v(\mathbf{x} + e_w + e_l) - T_2v(\mathbf{x} + e_w + e_d), \forall (d, l, q, w)$ s.t. $0 < d \leq l \leq q \leq w$. Recall that as we assume v satisfies Properties 2 and 3, it is always optimal to launch a new product by choosing a project with highest expected reward. We consider the following scenarios:

- (1) Suppose that $x_j = 0, \forall j \leq l$. Then $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q) + \rho_l - v(\mathbf{x} + e_q) - \rho_d = v(\mathbf{x} + e_w) + \rho_l - v(\mathbf{x} + e_w) - \rho_d = T_2v(\mathbf{x} + e_w + e_l) - T_2v(\mathbf{x} + e_w + e_d)$.
- (2) Suppose that $x_j = 0, \forall j \leq d$, and $x_j \geq 1, \exists j \in \{d+1, d+2, \dots, l\}$. Define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 5, $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q + e_l - e_b) + \rho_b - v(\mathbf{x} + e_q) - \rho_d \leq v(\mathbf{x} + e_w + e_l - e_b) + \rho_b - v(\mathbf{x} + e_w) - \rho_d = T_2v(\mathbf{x} + e_w + e_l) - T_2v(\mathbf{x} + e_w + e_d)$.
- (3) Suppose that $x_j \geq 1, \exists j \leq d$. Again define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 5, $T_2v(\mathbf{x} + e_q + e_l) - T_2v(\mathbf{x} + e_q + e_d) = v(\mathbf{x} + e_q + e_l - e_b) + \rho_b - v(\mathbf{x} + e_q + e_d - e_b) - \rho_b \leq v(\mathbf{x} + e_w + e_l - e_b) + \rho_b - v(\mathbf{x} + e_w + e_d - e_b) - \rho_b = T_2v(\mathbf{x} + e_w + e_l) - T_2v(\mathbf{x} + e_w + e_d)$.

Therefore T_2v satisfies Property 5.

Property 6. We will prove T_2v satisfies Property 6, i.e., $\sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x}), \forall (l, q)$ s.t. $0 < l \leq q$. Recall that as we assume v satisfies Properties 2 and 3, it is always optimal to launch a new product by choosing a project with highest expected reward. Taking $q = n$, we consider the following scenarios:

- (1) Suppose that $x_j = 0, \forall j \leq q = n$. As we assume v satisfies Property 2, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q) + \rho_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x} + e_q) + \rho_j)f_{0 \rightarrow j} - v(\mathbf{x}) - \rho_q \\
&= v(\mathbf{x} + e_q) + \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{n \geq j > l} \rho_j f_{0 \rightarrow j} - v(\mathbf{x}) - \rho_q \\
&\leq \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{n \geq j > l} \rho_j f_{0 \rightarrow j} \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x}) + \rho_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x}) + \rho_j)f_{0 \rightarrow j} - v(\mathbf{x}) \\
&= \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x})
\end{aligned}$$

- (2) Suppose that $x_j = 0, \forall j \leq l$, and $x_j \geq 1, \exists j \in \{l+1, l+2, \dots, q\}$. Define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 6, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q) + \rho_l)f_{0 \rightarrow j} + \sum_{b \geq j > l} (v(\mathbf{x} + e_q) + \rho_j)f_{0 \rightarrow j}
\end{aligned}$$

$$\begin{aligned}
& + \sum_{n \geq j > b} (v(\mathbf{x} + e_q + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) - \rho_b \\
& = \sum_{1 \leq j \leq b} v(\mathbf{x} + e_q) f_{0 \rightarrow j} + \sum_{n \geq j > b} v(\mathbf{x} + e_q + e_j - e_b) f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) \\
& + \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{b \geq j > l} \rho_j f_{0 \rightarrow j} - \sum_{1 \leq j \leq b} \rho_b f_{0 \rightarrow j} \\
& \leq \sum_{1 \leq j \leq b} v(\mathbf{x}) f_{0 \rightarrow j} + \sum_{n \geq j > b} v(\mathbf{x} + e_j - e_b) f_{0 \rightarrow j} - v(\mathbf{x} - e_b) + \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{b \geq j > l} \rho_j f_{0 \rightarrow j} - \sum_{1 \leq j \leq b} \rho_b f_{0 \rightarrow j} \\
& = \sum_{1 \leq j \leq l} (v(\mathbf{x}) + \rho_l) f_{0 \rightarrow j} + \sum_{b \geq j > l} (v(\mathbf{x}) + \rho_j) f_{0 \rightarrow j} + \sum_{n \geq j > b} (v(\mathbf{x} + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} - e_b) - \rho_b \\
& = \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x})
\end{aligned}$$

(3) Suppose that $x_j \geq 1$, $\exists j \leq l$. Again define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 6, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \\
& = \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q + e_l - e_b) + \rho_b) f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x} + e_q + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) - \rho_b \\
& \leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_l - e_b) + \rho_b) f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x} + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} - e_b) - \rho_b \\
& = \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x})
\end{aligned}$$

Therefore $T_2 v$ satisfies Property 6 if $q = n$. We below prove that $T_2 v$ also satisfies Property 7. Thus the following inequalities hold:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \\
& \leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+1} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+1} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+1}), \\
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+1} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+1} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+1}) \\
& \leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+2} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+2} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+2}), \\
& \quad \vdots \\
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_n + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_n + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_n) \\
& \leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x}).
\end{aligned}$$

Summation of the above inequalities implies $\sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x})$ for $l \leq q \leq n$. Therefore $T_2 v$ satisfies Property 6.

Property 7. We will prove T_2v satisfies Property 7, i.e., $\sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_w + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_w + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_w)$, $\forall (l, q, w)$ s.t. $0 < l \leq q < w$. Recall that as we assume v satisfies Properties 2 and 3, it is always optimal to launch a new product by choosing a project with highest expected reward. Taking $q = w - 1$, we consider the following scenarios:

- (1) Suppose that $x_j = 0$, $\forall j \leq q = w - 1$. As we assume v satisfies Property 3, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q) + \rho_l)f_{0 \rightarrow j} + \sum_{q \geq j > l} (v(\mathbf{x} + e_q) + \rho_j)f_{0 \rightarrow j} + \sum_{n \geq j \geq w} (v(\mathbf{x} + e_j) + \rho_q)f_{0 \rightarrow j} - v(\mathbf{x}) - \rho_q \\
&= \sum_{1 \leq j \leq q} (v(\mathbf{x} + e_q) - \rho_q)f_{0 \rightarrow j} + \sum_{n \geq j \geq w} v(\mathbf{x} + e_j)f_{0 \rightarrow j} - v(\mathbf{x}) + \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{q \geq j > l} \rho_j f_{0 \rightarrow j} \\
&\leq \sum_{1 \leq j \leq q} (v(\mathbf{x} + e_w) - \rho_w)f_{0 \rightarrow j} + \sum_{n \geq j \geq w} v(\mathbf{x} + e_j)f_{0 \rightarrow j} - v(\mathbf{x}) + \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{q \geq j > l} \rho_j f_{0 \rightarrow j} \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_w) + \rho_l)f_{0 \rightarrow j} + \sum_{q \geq j > l} (v(\mathbf{x} + e_w) + \rho_j)f_{0 \rightarrow j} + \sum_{n \geq j \geq w} (v(\mathbf{x} + e_j) + \rho_w)f_{0 \rightarrow j} - v(\mathbf{x}) - \rho_w \\
&= \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_w + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_w + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_w)
\end{aligned}$$

- (2) Suppose that $x_j = 0$, $\forall j \leq l$, and $x_j \geq 1$, $\exists j \in \{l + 1, l + 2, \dots, q\}$. Define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 7, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_q + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_q + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_q) \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q) + \rho_l)f_{0 \rightarrow j} + \sum_{b \geq j > l} (v(\mathbf{x} + e_q) + \rho_j)f_{0 \rightarrow j} \\
&+ \sum_{n \geq j > b} (v(\mathbf{x} + e_q + e_j - e_b) + \rho_b)f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) - \rho_b \\
&= \sum_{1 \leq j \leq b} v(\mathbf{x} + e_q)f_{0 \rightarrow j} + \sum_{n \geq j > b} v(\mathbf{x} + e_q + e_j - e_b)f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) \\
&+ \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{b \geq j > l} \rho_j f_{0 \rightarrow j} - \sum_{1 \leq j \leq b} \rho_b f_{0 \rightarrow j} \\
&\leq \sum_{1 \leq j \leq b} v(\mathbf{x} + e_w)f_{0 \rightarrow j} + \sum_{n \geq j > b} v(\mathbf{x} + e_w + e_j - e_b)f_{0 \rightarrow j} - v(\mathbf{x} + e_w - e_b) \\
&+ \sum_{1 \leq j \leq l} \rho_l f_{0 \rightarrow j} + \sum_{b \geq j > l} \rho_j f_{0 \rightarrow j} - \sum_{1 \leq j \leq b} \rho_b f_{0 \rightarrow j} \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_w) + \rho_l)f_{0 \rightarrow j} + \sum_{b \geq j > l} (v(\mathbf{x} + e_w) + \rho_j)f_{0 \rightarrow j} \\
&+ \sum_{n \geq j > b} (v(\mathbf{x} + e_w + e_j - e_b) + \rho_b)f_{0 \rightarrow j} - v(\mathbf{x} + e_w - e_b) - \rho_b \\
&= \sum_{1 \leq j \leq l} T_2v(\mathbf{x} + e_w + e_l)f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2v(\mathbf{x} + e_w + e_j)f_{0 \rightarrow j} - T_2v(\mathbf{x} + e_w)
\end{aligned}$$

(3) Suppose that $x_j \geq 1$, $\exists j \leq l$. Again define b as the smallest j such that $x_j \geq 1$. As we assume v satisfies Property 7, the following inequality holds:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \\
&= \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_q + e_l - e_b) + \rho_b) f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x} + e_q + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} + e_q - e_b) - \rho_b \\
&\leq \sum_{1 \leq j \leq l} (v(\mathbf{x} + e_w + e_l - e_b) + \rho_b) f_{0 \rightarrow j} + \sum_{n \geq j > l} (v(\mathbf{x} + e_w + e_j - e_b) + \rho_b) f_{0 \rightarrow j} - v(\mathbf{x} + e_w - e_b) - \rho_b \\
&= \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_w)
\end{aligned}$$

Therefore $T_2 v$ satisfies Property 7 if $q = w - 1$. This implies that $T_2 v$ also satisfies Property 7 for $l \leq q \leq w - 1$:

$$\begin{aligned}
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \\
&\leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+1} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+1} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+1}), \\
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+1} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+1} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+1}) \\
&\leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{q+2} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{q+2} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{q+2}), \\
& \quad \vdots \\
& \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_{w-1} + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_{w-1} + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_{w-1}) \\
&\leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_w).
\end{aligned}$$

Summation of the above inequalities implies $\sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_q) \leq \sum_{1 \leq j \leq l} T_2 v(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} + \sum_{n \geq j > l} T_2 v(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} - T_2 v(\mathbf{x} + e_w)$ for $l \leq q \leq w - 1$. Therefore $T_2 v$ satisfies Property 7. Hence $T_2 v$ satisfies Properties 1–7, i.e., $T_2 : \widehat{V} \rightarrow \widehat{V}$.

$-h \in \widehat{V}$. We below show $-h$ satisfies Properties 1–7.

Property 1. As h is constant as long as the total number of projects remains the same, $-h(\mathbf{x} + e_w) = -h(\mathbf{x} + e_q)$.

Property 2. As h is increasing in the total number of projects, $-h(\mathbf{x} + e_w) \leq -h(\mathbf{x}) + \rho_w$.

Property 3. $-h(\mathbf{x} + e_q) + \rho_w = -h(\mathbf{x} + e_w) + \rho_w \leq -h(\mathbf{x} + e_w) + \rho_q$.

Property 4. $-h(\mathbf{x} + e_q + e_l) - (-h(\mathbf{x} + e_q + e_d)) = -h(\mathbf{x} + e_l) - (-h(\mathbf{x} + e_d)) = 0$.

Property 5. $-h(\mathbf{x} + e_q + e_l) - (-h(\mathbf{x} + e_q + e_d)) = -h(\mathbf{x} + e_w + e_l) - (-h(\mathbf{x} + e_w + e_d)) = 0$.

Property 6. As h is convex in the total number of projects, $-\sum_{1 \leq j \leq l} h(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} + h(\mathbf{x} + e_q) = -\sum_{1 \leq j \leq l} h(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_q + e_q) f_{0 \rightarrow j} + h(\mathbf{x} + e_q) = -h(\mathbf{x} + 2e_q) + h(\mathbf{x} + e_q) \leq -h(\mathbf{x} + e_q) + h(\mathbf{x}) = -\sum_{1 \leq j \leq l} h(\mathbf{x} + e_q) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_q) f_{0 \rightarrow j} + h(\mathbf{x}) = -\sum_{1 \leq j \leq l} h(\mathbf{x} + e_l) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_j) f_{0 \rightarrow j} + h(\mathbf{x})$.

Property 7. $-\sum_{1 \leq j \leq l} h(\mathbf{x} + e_q + e_l) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_q + e_j) f_{0 \rightarrow j} + h(\mathbf{x} + e_q) = -h(\mathbf{x} + 2e_q) + h(\mathbf{x} + e_q) = -\sum_{1 \leq j \leq l} h(\mathbf{x} + e_w + e_l) f_{0 \rightarrow j} - \sum_{n \geq j > l} h(\mathbf{x} + e_w + e_j) f_{0 \rightarrow j} + h(\mathbf{x} + e_w)$.

Hence $-h$ satisfies Properties 1–7, i.e., $-h \in \widehat{V}$.

$T: \widehat{V} \rightarrow \widehat{V}$. Assume v satisfies Properties 1–7, i.e., $v \in \widehat{V}$. We proved that $T_1 v$, $T_2 v$, and $-h$ satisfy Properties 1–7. Notice that Tv satisfies Properties 1, 4, 5, 6, and 7, as these properties are preserved by linear transformations. We below show that Tv also satisfies Properties 2 and 3.

- We will prove Tv satisfies Property 2, i.e., $Tv(\mathbf{x} + e_w) \leq Tv(\mathbf{x}) + \rho_w$, $\forall w > 0$. Since (i) h is increasing in the total number of projects, (ii) $T_1 v$ and $T_2 v$ satisfy Property 2, and (iii) $\mu_1 + \mu_2 \leq 1$, Property 2 holds: $Tv(\mathbf{x} + e_w) = -h(\mathbf{x} + e_w) + \mu_1 T_1 v(\mathbf{x} + e_w) + \mu_2 T_2 v(\mathbf{x} + e_w) \leq -h(\mathbf{x}) + \mu_1 (T_1 v(\mathbf{x}) + \rho_w) + \mu_2 (T_2 v(\mathbf{x}) + \rho_w) \leq Tv(\mathbf{x}) + \rho_w$.
- We will prove Tv satisfies Property 3, i.e., $Tv(\mathbf{x} + e_q) + \rho_w \leq Tv(\mathbf{x} + e_w) + \rho_q$, $\forall (q, w)$ s.t. $0 < q < w$. Since (i) $T_1 v$ and $T_2 v$ satisfy Property 3, (ii) $\mu_1 + \mu_2 \leq 1$, and (iii) $\rho_w \leq \rho_q$, Property 3 holds: $Tv(\mathbf{x} + e_q) = -h(\mathbf{x} + e_q) + \mu_1 T_1 v(\mathbf{x} + e_q) + \mu_2 T_2 v(\mathbf{x} + e_q) \leq -h(\mathbf{x} + e_w) + \mu_1 (T_1 v(\mathbf{x} + e_w) - \rho_w + \rho_q) + \mu_2 (T_2 v(\mathbf{x} + e_w) - \rho_w + \rho_q) \leq Tv(\mathbf{x} + e_w) - \rho_w + \rho_q$.

Therefore Tv satisfies Properties 1–7, i.e., $Tv \in \widehat{V}$. Thus $T: \widehat{V} \rightarrow \widehat{V}$. Following Theorem 6.10.4 in Puterman (1994), we verify that $\lim_{k \rightarrow \infty} (T^k v_0)(\mathbf{x}) = v^*(\mathbf{x})$ where v_0 is the zero function, v^* is the optimal reward function, and T^k refers to k compositions of operator T . Since $v_0 \in \widehat{V}$ and $T: \widehat{V} \rightarrow \widehat{V}$, we have $T^k v_0 \in \widehat{V}$, and thus $v^* \in \widehat{V}$. \square

Proof of Theorem 1. By Lemma 1, we know $v^* \in \widehat{V}$. Define, for $v^* \in \widehat{V}$,

$$S_j^*(\mathbf{x}_{-j}) = \min \left\{ z_j : \sum_{1 \leq k \leq n} v^*(\mathbf{z} + e_k) f_{0 \rightarrow k} - v^*(\mathbf{z}) < c, z_{j'} = x_{j'}, \forall j' \neq j, z_j \in \mathbb{N}_0 \right\}, \forall j.$$

Since v^* satisfies Property 6, $\sum_{1 \leq k \leq n} v^*(\mathbf{z} + e_k) f_{0 \rightarrow k} - v^*(\mathbf{z})$ is nonincreasing in z_j . As z_j increases, since the holding cost rate h is strictly increasing, this difference will eventually cross c . Hence the optimal stationary policy is a state-dependent noncongestive promotion policy with state-dependent promote-up-to levels $S_j^*(\mathbf{x}_{-j})$. Next we will prove properties (i)–(v):

- Pick arbitrary j and k such that $j \neq k$. We will show that the optimal promote-up-to level for category j obeys $S_j^*(\mathbf{z}_{-j}) \leq S_j^*(\mathbf{x}_{-j})$, where $z_{j'} = x_{j'}$, $\forall j' \notin \{j, k\}$, and $z_k = x_k + 1$. Suppose that

$S_j^*(\mathbf{z}_{-j}) > S_j^*(\mathbf{x}_{-j})$. By definition, it is optimal to initiate an experiment at \mathbf{z} if $z_j < S_j^*(\mathbf{z}_{-j})$, and it is not optimal to do so at \mathbf{x} if $x_j = S_j^*(\mathbf{x}_{-j}) < S_j^*(\mathbf{z}_{-j})$. But we have a contradiction when $z_j = x_j = S_j^*(\mathbf{x}_{-j})$: If it is optimal to initiate an experiment at \mathbf{z} , it should also be optimal to do so at state \mathbf{x} (due to Property 6). Thus we must have $S_j^*(\mathbf{z}_{-j}) \leq S_j^*(\mathbf{x}_{-j})$.

- ii. Pick arbitrary j , k , and k' such that $j \notin \{k, k'\}$ and $k' < k$. We will show that the optimal promote-up-to level for category j obeys $S_j^*(\mathbf{z}_{-j}) \leq S_j^*(\mathbf{x}_{-j})$, where $z_{j'} = x_{j'}$, $\forall j' \notin \{j, k, k'\}$, $z_k + 1 = x_k$ and $z_{k'} = x_{k'} + 1$. Suppose that $S_j^*(\mathbf{z}_{-j}) > S_j^*(\mathbf{x}_{-j})$. By definition, it is optimal to initiate an experiment at \mathbf{z} if $z_j < S_j^*(\mathbf{z}_{-j})$, and it is not optimal to do so at \mathbf{x} if $x_j = S_j^*(\mathbf{x}_{-j}) < S_j^*(\mathbf{z}_{-j})$. But we have a contradiction when $z_j = x_j = S_j^*(\mathbf{x}_{-j})$: If it is optimal to initiate an experiment at \mathbf{z} , it should also be optimal to do so at \mathbf{x} (due to Property 7). Thus we must have $S_j^*(\mathbf{z}_{-j}) \leq S_j^*(\mathbf{x}_{-j})$.

- iii. Suppose that $\exists k$ s.t. $x_k > 0$. Property 2 implies it is always optimal to launch a new product:

$$T_2 v(\mathbf{x}) = \max \left\{ v(\mathbf{x}), \max_{1 \leq j \leq n} \text{ s.t. } \mathbf{x} \geq e_j v(\mathbf{x} - e_j) + \rho_j \right\} = \max_{1 \leq j \leq n} \text{ s.t. } \mathbf{x} \geq e_j v(\mathbf{x} - e_j) + \rho_j.$$

- iv. Suppose that $\exists k$ s.t. $x_k > 0$. Point (iii) implies that $T_2 v(\mathbf{x}) = \max_{1 \leq j \leq n} \text{ s.t. } \mathbf{x} \geq e_j v(\mathbf{x} - e_j) + \rho_j$.

Property 3 implies that it is always optimal to choose a project with highest expected reward:

$$T_2 v(\mathbf{x}) = v(\mathbf{x} - e_{j^*}) + \rho_{j^*} \text{ where } j^* \text{ is the smallest } j \text{ such that } x_j > 0.$$

- v. Lastly, we will prove it is never optimal to interrupt any experiment. Assume that an experiment is optimally initiated at a given state \mathbf{x} . We then consider the following two cases:

- Suppose that $\exists j$ s.t. $x_j > 0$. Point (iii) implies that it is optimal to launch a new product at state \mathbf{x} . Suppose that a new product is placed on the market before the experiment is complete. Thus the system moves to a state \mathbf{z} where $z_{j^*} = x_{j^*} - 1$ and $z_j = x_j$, $\forall j \neq j^*$ (j^* is the category with highest expected reward among available categories). Property 6 implies that it is optimal to initiate an experiment at state \mathbf{z} : The experiment initiated at \mathbf{x} can be resumed at \mathbf{z} . Now suppose that a new product is placed on the market after the experiment is complete. But then the experiment is not interrupted.
- Suppose that $x_j = 0$, $\forall j$. The system can move to a new state only after the experiment is complete. Therefore the experiment is not interrupted. \square

EC.3. Proofs of the Results in Section 4

We will use the following auxiliary lemma to prove Lemma 2:

LEMMA EC.1. *A real-valued function on \mathbb{N}_0^n satisfying Property 10 also satisfies Properties 11–13.*

Proof of Lemma EC.1. Assuming a real-valued function g on \mathbb{N}_0^n satisfies Property 10, we will prove g satisfies Properties 11–13. First, we will show g satisfies Property 11. Pick arbitrary (l, q, w) s.t. $0 < l \leq q < w \leq m + 1$. As we assume g satisfies Property 10, the following inequalities hold:

$$\begin{aligned} g(\mathbf{x} + e_l + e_q) - g(\mathbf{x} + e_{l-1} + e_q) &\leq g(\mathbf{x} + e_l + e_{q+1}) - g(\mathbf{x} + e_{l-1} + e_{q+1}) \\ g(\mathbf{x} + e_l + e_{q+1}) - g(\mathbf{x} + e_{l-1} + e_{q+1}) &\leq g(\mathbf{x} + e_l + e_{q+2}) - g(\mathbf{x} + e_{l-1} + e_{q+2}) \\ &\vdots \\ g(\mathbf{x} + e_l + e_{w-2}) - g(\mathbf{x} + e_{l-1} + e_{w-2}) &\leq g(\mathbf{x} + e_l + e_{w-1}) - g(\mathbf{x} + e_{l-1} + e_{w-1}) \\ g(\mathbf{x} + e_l + e_{w-1}) - g(\mathbf{x} + e_{l-1} + e_{w-1}) &\leq g(\mathbf{x} + e_l + e_w) - g(\mathbf{x} + e_{l-1} + e_w) \end{aligned}$$

Summation of the above inequalities implies $g(\mathbf{x} + e_l + e_q) - g(\mathbf{x} + e_{l-1} + e_q) \leq g(\mathbf{x} + e_l + e_w) - g(\mathbf{x} + e_{l-1} + e_w)$. Thus g satisfies Property 11.

Second, we will show g satisfies Property 12. Pick arbitrary $\forall(l, q, w)$ s.t. $0 \leq l < q \leq w - 1 \leq m$. As we assume g satisfies Property 10, the following inequalities hold:

$$\begin{aligned} g(\mathbf{x} + e_w + e_l) - g(\mathbf{x} + e_{w-1} + e_l) &\leq g(\mathbf{x} + e_w + e_{l+1}) - g(\mathbf{x} + e_{w-1} + e_{l+1}) \\ g(\mathbf{x} + e_w + e_{l+1}) - g(\mathbf{x} + e_{w-1} + e_{l+1}) &\leq g(\mathbf{x} + e_w + e_{l+2}) - g(\mathbf{x} + e_{w-1} + e_{l+2}) \\ &\vdots \\ g(\mathbf{x} + e_w + e_{q-2}) - g(\mathbf{x} + e_{w-1} + e_{q-2}) &\leq g(\mathbf{x} + e_w + e_{q-1}) - g(\mathbf{x} + e_{w-1} + e_{q-1}) \\ g(\mathbf{x} + e_w + e_{q-1}) - g(\mathbf{x} + e_{w-1} + e_{q-1}) &\leq g(\mathbf{x} + e_w + e_q) - g(\mathbf{x} + e_{w-1} + e_q) \end{aligned}$$

Summation of the above inequalities implies $g(\mathbf{x} + e_w + e_l) - g(\mathbf{x} + e_{w-1} + e_l) \leq g(\mathbf{x} + e_w + e_q) - g(\mathbf{x} + e_{w-1} + e_q)$. Thus g satisfies Property 12.

Lastly, we will show g satisfies Property 13. Since g satisfies Property 11, taking $l = q$ and $w = m + 1$, the following inequality holds: $g(\mathbf{x} + e_l + e_l) - g(\mathbf{x} + e_{l-1} + e_l) \leq g(\mathbf{x} + e_l) - g(\mathbf{x} + e_{l-1})$ for $l \in \{1, \dots, m\}$. Since g satisfies Property 12, taking $l = 0$ and $q = w - 1$, the following inequality holds: $g(\mathbf{x} + e_w) - g(\mathbf{x} + e_{w-1}) \leq g(\mathbf{x} + e_w + e_{w-1}) - g(\mathbf{x} + e_{w-1} + e_{w-1})$ for $w \in \{2, \dots, m\}$. When $l = w \neq 1$, summation of these two inequalities implies $g(\mathbf{x} + e_l + e_l) - g(\mathbf{x} + e_{l-1} + e_l) \leq g(\mathbf{x} + e_l + e_{l-1}) - g(\mathbf{x} + e_{l-1} + e_{l-1})$ for $l \in \{2, \dots, m\}$. Now suppose that $l = 1$. Since g satisfies Property 11, taking $l = q = 1$ and $w = m + 1$, the following inequality holds: $g(\mathbf{x} + 2e_1) - g(\mathbf{x} + e_1) \leq g(\mathbf{x} + e_1) - g(\mathbf{x})$. Thus g satisfies Property 13. \square

Proof of Lemma 2. Define \tilde{V} as the set of real-valued functions on \mathbb{N}_0^n that satisfy Properties 8–13. Also, define the operator T on the set of real-valued functions v as follows: $Tv(\mathbf{x}) = \lambda T_0v(\mathbf{x}) + \sum_{1 \leq i \leq m} \mu_i T_i v(\mathbf{x}) + \mu_{m+1} T_{m+1} v(\mathbf{x}) - h(\mathbf{x})$. We below show that $T_0 : \tilde{V} \rightarrow \tilde{V}$, $T_i : \tilde{V} \rightarrow \tilde{V}$, $T_{m+1} : \tilde{V} \rightarrow \tilde{V}$, and $-h \in \tilde{V}$. We will then prove that $T : \tilde{V} \rightarrow \tilde{V}$.

$T_0 : \tilde{V} \rightarrow \tilde{V}$. Assume that v satisfies Properties 8–13. We below show T_0v satisfies Properties 8–13.

Property 8. We will prove T_0v satisfies Property 8, i.e., $T_0v(\mathbf{x} + e_q) \leq T_0v(\mathbf{x} + e_w)$, $\forall (q, w)$ s.t. $0 < q < w \leq m$. We consider the following two scenarios depending on the optimal action at $T_0v(\mathbf{x} + e_q)$ (if this inequality holds under a suboptimal action of $T_0v(\mathbf{x} + e_w)$, it also holds under the optimal action of this operator, and thus we do not enforce the optimal action at this operator):

- (1) Suppose that $T_0v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q)$. As we assume v satisfies Property 8, $T_0v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q) \leq v(\mathbf{x} + e_w) \leq T_0v(\mathbf{x} + e_w)$.
- (2) Suppose that $T_0v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_l)$ where $l \geq 1$. This scenario is possible only when $\mathbf{x} + e_q \geq e_l$. Also, suppose that $l \neq q$. Thus we should have $x_l > 0$. As we assume v satisfies Property 8, $T_0v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_l) \leq v(\mathbf{x} + e_w - e_l) \leq T_0v(\mathbf{x} + e_w)$. Now suppose that $l = q$. Then it is easy to verify that $T_0v(\mathbf{x} + e_q) = v(\mathbf{x}) \leq T_0v(\mathbf{x} + e_w)$.

Therefore T_0v satisfies Property 8.

Property 9. We will prove T_0v satisfies Property 9, i.e., $T_0v(\mathbf{x} + e_m) \leq T_0v(\mathbf{x}) + \rho$. We consider the following two scenarios depending on the optimal action at $T_0v(\mathbf{x} + e_m)$:

- (1) Suppose that $T_0v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m)$. As we assume v satisfies Property 9, $T_0v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m) \leq v(\mathbf{x}) + \rho \leq T_0v(\mathbf{x}) + \rho$.
- (2) Suppose that $T_0v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m - e_l)$ where $l \geq 1$. This scenario is possible only when $\mathbf{x} + e_m \geq e_l$. Also, suppose that $l \neq m$. Thus we should have $x_l > 0$. As we assume v satisfies Property 9, $T_0v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m - e_l) \leq v(\mathbf{x} - e_l) + \rho \leq T_0v(\mathbf{x}) + \rho$. Now suppose that $l = m$. Then it is easy to verify that $T_0v(\mathbf{x} + e_m) = v(\mathbf{x}) \leq v(\mathbf{x}) + \rho \leq T_0v(\mathbf{x}) + \rho$.

Therefore T_0v satisfies Property 9.

Property 10. We will prove T_0v satisfies Property 10, i.e., $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$, $\forall (q, w)$ s.t. $q \neq w$ and $0 < q, w \leq m + 1$. We consider the following four scenarios depending on the optimal actions at $T_0v(\mathbf{x} + e_q + e_{w-1})$ and $T_0v(\mathbf{x} + e_{q-1} + e_w)$ (if this inequality holds under suboptimal actions of $T_0v(\mathbf{x} + e_{q-1} + e_{w-1})$ and/or $T_0v(\mathbf{x} + e_q + e_w)$, it also holds under optimal actions of these operators, and thus we do not enforce the optimal actions at these operators):

- (1) Suppose that $T_0v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_0v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. As we assume v satisfies Property 10, $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$.
- (2) Suppose that $T_0v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_Av(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_l)$ where $l \geq 1$. This scenario is possible only when $\mathbf{x} + e_{q-1} + e_w \geq e_l$. Also, suppose that $l < w$. Thus we should have $\mathbf{x} + e_{q-1} \geq e_l$. As we assume v satisfies Properties 10 and 12, the following inequalities hold:

$$\begin{aligned}
& T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
& \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_l) \\
& \leq v(\mathbf{x} + e_{q-1} + e_{w-1}) + v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_l) \\
& \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w - e_l) \\
& \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

Now suppose that $l \geq w$. As we assume v satisfies Property 8, if a project is to be terminated, it is optimal to choose this from the earliest possible stage. Thus we should have $l = w$. As we assume v satisfies Property 8, $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1}) \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1}) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$.

- (3) Suppose that $T_0v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_l)$ where $l \geq 1$, and $T_0v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_l$. Also, suppose that $l < q$. Thus we should have $\mathbf{x} + e_{w-1} \geq e_l$. As we assume v satisfies Properties 10 and 12, the following inequalities hold:

$$\begin{aligned}
& T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1} - e_l) - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_l) \\
& \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
& \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \\
& \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

Now suppose that $l \geq q$. As we assume v satisfies Property 8, we should have $l = q$. Again as we assume v satisfies Property 8, $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_{w-1}) - v(\mathbf{x} + e_{w-1}) = 0 \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$.

(4) Suppose that $T_0v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_l)$ where $l \geq 1$, and $T_0v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_d)$ where $d \geq 1$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_l$ and $\mathbf{x} + e_{q-1} + e_w \geq e_d$. Without loss of generality we assume $q < w$. First, suppose that $l \neq q$. As we assume v satisfies Property 8, we should have $l < q$ and $x_l > 0$. Also we should have $l = d$. As we assume v satisfies Property 10, $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1} - e_l) - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_l) \leq v(\mathbf{x} + e_q + e_w - e_l) - v(\mathbf{x} + e_{q-1} + e_w - e_l) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$. Second, suppose that $l = q$. As we assume v satisfies Property 8, we should have $x_i = 0, \forall i < q$. Also we should have $d = q - 1$. Then it is easy to verify that $T_0v(\mathbf{x} + e_q + e_{w-1}) - T_0v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_{w-1}) - v(\mathbf{x} + e_{w-1}) = v(\mathbf{x} + e_w) - v(\mathbf{x} + e_w) \leq T_0v(\mathbf{x} + e_q + e_w) - T_0v(\mathbf{x} + e_{q-1} + e_w)$.

Therefore T_0v satisfies Property 10. By Lemma EC.1, T_0v also satisfies Properties 11–13. Hence we showed that T_0v satisfies Properties 8–13, i.e., $T_0 : \tilde{V} \rightarrow \tilde{V}$.

$T_i : \tilde{V} \rightarrow \tilde{V}$. Assume that v satisfies Properties 8–13. We below show T_iv satisfies Properties 8–13, $\forall i$.

Property 8. We will prove T_iv satisfies Property 8, i.e., $T_iv(\mathbf{x} + e_q) \leq T_iv(\mathbf{x} + e_w), \forall (q, w)$ s.t. $0 < q < w \leq m$. We consider the following two scenarios depending on the optimal action at $T_iv(\mathbf{x} + e_q)$ (if this inequality holds under a suboptimal action of $T_iv(\mathbf{x} + e_w)$, it also holds under the optimal action of this operator, and thus we do not enforce the optimal action at this operator):

- (1) Suppose that $T_iv(\mathbf{x} + e_q) = v(\mathbf{x} + e_q)$. As we assume v satisfies Property 8, $T_iv(\mathbf{x} + e_q) = v(\mathbf{x} + e_q) \leq v(\mathbf{x} + e_w) \leq T_iv(\mathbf{x} + e_w)$.
- (2) Suppose that $T_iv(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_{i-1} + e_i) - c_i$. This scenario is possible only when $\mathbf{x} + e_q \geq e_{i-1}$. Also, suppose that $i \neq q + 1$. Since $\mathbf{x} + e_q \geq e_{i-1}$, we should have $\mathbf{x} \geq e_{i-1}$. As we assume v satisfies Property 8, $T_iv(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_{i-1} + e_i) - c_i \leq v(\mathbf{x} + e_w - e_{i-1} + e_i) - c_i \leq T_iv(\mathbf{x} + e_w)$. Now suppose that $i = q + 1$. As we assume v satisfies Property 8 and $q < w$, $T_iv(\mathbf{x} + e_q) = v(\mathbf{x} + e_{q+1}) - c_{q+1} \leq v(\mathbf{x} + e_w) \leq T_iv(\mathbf{x} + e_w)$.

Therefore T_iv satisfies Property 8.

Property 9. We will prove T_iv satisfies Property 9, i.e., $T_iv(\mathbf{x} + e_m) \leq T_iv(\mathbf{x}) + \rho$. We consider the following two scenarios depending on the optimal action at $T_iv(\mathbf{x} + e_m)$:

- (1) Suppose that $T_iv(\mathbf{x} + e_m) = v(\mathbf{x} + e_m)$. As we assume v satisfies Property 9, $T_iv(\mathbf{x} + e_m) = v(\mathbf{x} + e_m) \leq v(\mathbf{x}) + \rho \leq T_iv(\mathbf{x}) + \rho$.
- (2) Suppose that $T_iv(\mathbf{x} + e_m) = v(\mathbf{x} + e_m - e_{i-1} + e_i) - c_i$. This scenario is possible only when

$\mathbf{x} + e_m \geq e_{i-1}$. Since $i \leq m$, we should have $\mathbf{x} \geq e_{i-1}$. As we assume v satisfies Property 9, $T_i v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m - e_{i-1} + e_i) - c_i \leq v(\mathbf{x} - e_{i-1} + e_i) - c_i + \rho \leq T_i v(\mathbf{x}) + \rho$.

Therefore $T_i v$ satisfies Property 9.

Property 10. We will prove $T_i v$ satisfies Property 10, i.e., $T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)$, $\forall (q, w)$ s.t. $q \neq w$ and $0 < q, w \leq m + 1$. We consider the following four scenarios depending on the optimal actions at $T_i v(\mathbf{x} + e_q + e_{w-1})$ and $T_i v(\mathbf{x} + e_{q-1} + e_w)$ (if this inequality holds under suboptimal actions of $T_i v(\mathbf{x} + e_{q-1} + e_{w-1})$ and/or $T_i v(\mathbf{x} + e_q + e_w)$, it also holds under optimal actions of these operators, and thus we do not enforce the optimal actions at these operators):

- (1) Suppose that $T_i v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_i v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. As we assume v satisfies Property 10, $T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)$.
- (2) Suppose that $T_i v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_i v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_{i-1} + e_i) - c_i$. This scenario is possible only when $\mathbf{x} + e_{q-1} + e_w \geq e_{i-1}$. Also, suppose that $i \neq q$. Since $\mathbf{x} + e_{q-1} + e_w \geq e_{i-1}$, we should have $\mathbf{x} + e_w \geq e_{i-1}$. As we assume v satisfies Property 10, the following inequalities hold:

$$\begin{aligned}
& T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
& \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
& \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \\
& \leq v(\mathbf{x} + e_q + e_w - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_{q-1} + e_w - e_{i-1} + e_i) + c_i \\
& \leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

Now suppose that $i = q$. Then it is easy to verify that $T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_q + e_{w-1}) + c_q = v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_q + e_w) + c_q \leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)$.

- (3) Suppose that $T_i v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_{i-1} + e_i) - c_i$ and $T_i v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_{i-1}$. Also, suppose that $i \neq w$. Since $\mathbf{x} + e_q + e_{w-1} \geq e_{i-1}$, we should have $\mathbf{x} + e_q \geq e_{i-1}$. As we assume v satisfies Property 10, the following inequalities hold:

$$T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1})$$

$$\begin{aligned}
&\leq v(\mathbf{x} + e_q + e_{w-1} - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
&\leq v(\mathbf{x} + e_q + e_w - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_q + e_w) + v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
&\leq v(\mathbf{x} + e_q + e_w - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_{q-1} + e_w) \\
&\leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

Now suppose that $i = w$. Then it is easy to verify that $T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_w) - c_w - v(\mathbf{x} + e_{q-1} + e_w) + c_w = v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)$.

(4) Suppose that $T_i v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_{i-1} + e_i) - c_i$ and $T_i v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_{i-1} + e_i) - c_i$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_{i-1}$ and $\mathbf{x} + e_{q-1} + e_w \geq e_{i-1}$. Thus, since $q \neq w$, we should have $\mathbf{x} \geq e_{i-1}$. As we assume v satisfies Property 10, the following inequalities hold:

$$\begin{aligned}
&T_i v(\mathbf{x} + e_q + e_{w-1}) - T_i v(\mathbf{x} + e_{q-1} + e_{w-1}) \\
&\leq v(\mathbf{x} + e_q + e_{w-1} - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_{i-1} + e_i) + c_i \\
&\leq v(\mathbf{x} + e_q + e_w - e_{i-1} + e_i) - c_i - v(\mathbf{x} + e_{q-1} + e_w - e_{i-1} + e_i) + c_i \\
&\leq T_i v(\mathbf{x} + e_q + e_w) - T_i v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

Therefore $T_i v$ satisfies Property 10. By Lemma EC.1, $T_i v$ also satisfies Properties 11–13. Hence we showed that $T_i v$ satisfies Properties 8–13, i.e., $T_i : \tilde{V} \rightarrow \tilde{V}$.

$T_{m+1} : \tilde{V} \rightarrow \tilde{V}$. Assume that v satisfies Properties 8–13. We below show $T_{m+1} v$ satisfies Properties 8–13.

Property 8. We will prove $T_{m+1} v$ satisfies Property 8, i.e., $T_{m+1} v(\mathbf{x} + e_q) \leq T_{m+1} v(\mathbf{x} + e_w)$, $\forall (q, w)$ s.t. $0 < q < w \leq m$. We consider the following two scenarios depending on the optimal action at $T_{m+1} v(\mathbf{x} + e_q)$:

- (1) Suppose that $T_{m+1} v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q)$. As we assume v satisfies Property 8, $T_{m+1} v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q) \leq v(\mathbf{x} + e_w) \leq T_{m+1} v(\mathbf{x} + e_w)$.
- (2) Suppose that $T_{m+1} v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_m) + \rho$. This scenario is possible only when $\mathbf{x} + e_q \geq e_m$. Since $q < w \leq m$, note that $x_m > 0$. As we assume v satisfies Property 8, $T_{m+1} v(\mathbf{x} + e_q) = v(\mathbf{x} + e_q - e_m) + \rho \leq v(\mathbf{x} + e_w - e_m) + \rho \leq T_{m+1} v(\mathbf{x} + e_w)$.

Therefore $T_{m+1} v$ satisfies Property 8.

Property 9. We will prove $T_{m+1} v$ satisfies Property 9, i.e., $T_{m+1} v(\mathbf{x} + e_m) \leq T_{m+1} v(\mathbf{x}) + \rho$. We consider the following two scenarios depending on the optimal action at $T_{m+1} v(\mathbf{x} + e_m)$:

(1) Suppose that $T_{m+1}v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m)$. As we assume v satisfies Property 9, $T_{m+1}v(\mathbf{x} + e_m) = v(\mathbf{x} + e_m) \leq v(\mathbf{x}) + \rho \leq T_{m+1}v(\mathbf{x}) + \rho$.

(2) Suppose that $T_{m+1}v(\mathbf{x} + e_m) = v(\mathbf{x}) + \rho$. Then $T_{m+1}v(\mathbf{x} + e_m) = v(\mathbf{x}) + \rho \leq T_{m+1}v(\mathbf{x}) + \rho$.

Therefore $T_{m+1}v$ satisfies Property 9.

Property 10. We will prove $T_{m+1}v$ satisfies Property 10, i.e., $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$, $\forall (q, w)$ s.t. $q \neq w$ and $0 < q, w \leq m + 1$.

We consider the following four scenarios depending on the optimal actions at $T_{m+1}v(\mathbf{x} + e_q + e_{w-1})$ and $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$:

(1) Suppose that $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. As we assume v satisfies Property 10, $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$.

(2) Suppose that $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$ and $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_m) + \rho$. This scenario is possible only when $\mathbf{x} + e_{q-1} + e_w \geq e_m$. As we assume v satisfies Property 9, it is always optimal to launch a new product if it is feasible. Thus, since $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1})$, we should have $x_m = 0$, $q \neq m$, and $w \neq m + 1$. Since $\mathbf{x} + e_{q-1} + e_w \geq e_m$, we should also have $q = m + 1$ and/or $w = m$. First, suppose that $q = m + 1$ and $w \neq m$. Then the following inequalities hold (recall that e_{m+1} is a zero vector of dimension m):

$$\begin{aligned} T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) &\leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{w-1}) - \rho \\ &= v(\mathbf{x} + e_{w-1}) - v(\mathbf{x} + e_{w-1}) - \rho \\ &= v(\mathbf{x} + e_w) - v(\mathbf{x} + e_w) - \rho \\ &= v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_w) - \rho \\ &\leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) \end{aligned}$$

Second, suppose that $q \neq m + 1$ and $w = m$. Thus $q < m$. As we assume v satisfies Property 11, $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q) + \rho - v(\mathbf{x} + e_{q-1}) - \rho \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$. Lastly, suppose that $q = m + 1$ and $w = m$. Then the following inequalities hold:

$$T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1}) - v(\mathbf{x} + e_{w-1}) - \rho$$

$$\begin{aligned}
&= v(\mathbf{x} + e_{w-1}) - v(\mathbf{x} + e_{w-1}) - \rho \\
&= v(\mathbf{x} + e_w) - v(\mathbf{x} + e_w) - \rho \\
&= v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_w) - \rho \\
&\leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)
\end{aligned}$$

- (3) Suppose that $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_m) + \rho$ and $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_m$. As we assume v satisfies Property 9, it is always optimal to launch a new product if it is feasible. Thus, since $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w)$, we should have $x_m = 0$, $q \neq m + 1$, and $w \neq m$. Since $\mathbf{x} + e_q + e_{w-1} \geq e_m$, we should also have $q = m$ and/or $w = m + 1$. First, suppose that $q = m$ and $w \neq m + 1$. Thus $w < m$. As we assume v satisfies Property 11, $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_{w-1}) + \rho - v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_w) + \rho - v(\mathbf{x} + e_{q-1} + e_w) \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$. Second, suppose that $q \neq m$ and $w = m + 1$. Then $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q) + \rho - v(\mathbf{x} + e_{q-1}) - \rho = v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$. Lastly, suppose that $q = m$ and $w = m + 1$. Then $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q) + \rho - v(\mathbf{x} + e_{q-1}) - \rho = v(\mathbf{x} + e_q + e_w) - v(\mathbf{x} + e_{q-1} + e_w) \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$.
- (4) Suppose that $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) = v(\mathbf{x} + e_q + e_{w-1} - e_m) + \rho$ and $T_{m+1}v(\mathbf{x} + e_{q-1} + e_w) = v(\mathbf{x} + e_{q-1} + e_w - e_m) + \rho$. This scenario is possible only when $\mathbf{x} + e_q + e_{w-1} \geq e_m$ and $\mathbf{x} + e_{q-1} + e_w \geq e_m$. Thus, since $q \neq w$, we should have $\mathbf{x} \geq e_m$. As we assume v satisfies Property 10, $T_{m+1}v(\mathbf{x} + e_q + e_{w-1}) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_{w-1}) \leq v(\mathbf{x} + e_q + e_{w-1} - e_m) + \rho - v(\mathbf{x} + e_{q-1} + e_{w-1} - e_m) - \rho \leq v(\mathbf{x} + e_q + e_w - e_m) + \rho - v(\mathbf{x} + e_{q-1} + e_w - e_m) - \rho \leq T_{m+1}v(\mathbf{x} + e_q + e_w) - T_{m+1}v(\mathbf{x} + e_{q-1} + e_w)$.

Therefore $T_{m+1}v$ satisfies Property 10. By Lemma EC.1, $T_{m+1}v$ also satisfies Properties 11–13.

Hence we showed that $T_{m+1}v$ satisfies Properties 8–13, i.e., $T_{m+1} : \tilde{V} \rightarrow \tilde{V}$.

$-h \in \tilde{V}$. We below show $-h$ satisfies Properties 8–13.

Property 8. As h is constant as long as the total number of projects remains the same, $-h(\mathbf{x} + e_q) = -h(\mathbf{x} + e_w)$.

Property 9. As h is increasing in the total number of projects, $-h(\mathbf{x} + e_m) \leq -h(\mathbf{x}) + \rho$.

Property 10. Pick arbitrary $\forall(q, w)$ s.t. $q \neq w$ and $0 < q, w \leq m + 1$.

- (1) Suppose that $m \geq q \geq 2$: $-h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = 0 = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_{q-1} + e_w)$.

- (2) Suppose that $q \in \{1, m+1\}$ and $m \geq w \geq 2$: If $q = 1$, $-h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = -h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{w-1}) = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_w) = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_{q-1} + e_w)$. If $q = m+1$, $-h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = -h(\mathbf{x} + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = -h(\mathbf{x} + e_w) + h(\mathbf{x} + e_{q-1} + e_w) = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_{q-1} + e_w)$.
- (3) Suppose that $q, w \in \{1, m+1\}$: If $q = 1$ and $w = m+1$, since h is convex in the total number of projects, $-h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = -h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{w-1}) \leq -h(\mathbf{x} + e_q) + h(\mathbf{x}) = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_{q-1} + e_w)$. If $q = m+1$ and $w = 1$, since h is convex in the total number of projects, $-h(\mathbf{x} + e_q + e_{w-1}) + h(\mathbf{x} + e_{q-1} + e_{w-1}) = -h(\mathbf{x}) + h(\mathbf{x} + e_{q-1}) \leq -h(\mathbf{x} + e_w) + h(\mathbf{x} + e_{q-1} + e_w) = -h(\mathbf{x} + e_q + e_w) + h(\mathbf{x} + e_{q-1} + e_w)$.

Therefore $-h$ satisfies Property 10. By Lemma EC.1, $-h$ also satisfies Properties 11–13. Thus $h \in \tilde{V}$.

$T: \tilde{V} \rightarrow \tilde{V}$. Assume v satisfies Properties 8–13, i.e., $v \in \tilde{V}$. We proved that $T_0v, T_iv, T_{m+1}v$, and $-h$ satisfy Properties 8–13. Notice that Tv satisfies Properties 8, 10, 11, 12, and 13, as these properties are preserved by linear transformations. We below show that Tv also satisfies Property 9, i.e., $Tv(\mathbf{x} + e_m) \leq Tv(\mathbf{x}) + \rho$. Since (i) h is increasing in the total number of projects, (ii) T_0v, T_iv , and $T_{m+1}v$ satisfy Property 9, and (iii) $\lambda + \sum_{1 \leq i \leq m} \mu_i + \mu_{m+1} \leq 1$, Property 9 holds: $Tv(\mathbf{x} + e_m) = -h(\mathbf{x} + e_m) + \lambda T_0v(\mathbf{x} + e_m) + \sum_{1 \leq i \leq m} \mu_i T_iv(\mathbf{x} + e_m) + \mu_{m+1} T_{m+1}v(\mathbf{x} + e_m) \leq -h(\mathbf{x}) + \lambda(T_0v(\mathbf{x}) + \rho) + \sum_{1 \leq i \leq m} \mu_i (T_iv(\mathbf{x}) + \rho) + \mu_{m+1} (T_{m+1}v(\mathbf{x}) + \rho) = Tv(\mathbf{x}) + (\lambda + \sum_{1 \leq i \leq m} \mu_i + \mu_{m+1})\rho \leq Tv(\mathbf{x}) + \rho$. Therefore Tv satisfies Properties 8–13, i.e., $Tv \in \tilde{V}$. Thus $T: \tilde{V} \rightarrow \tilde{V}$. Following Theorem 6.10.4 in Puterman (1994), we verify that $\lim_{k \rightarrow \infty} (T^k v_0)(\mathbf{x}) = v^*(\mathbf{x})$ where v_0 is the zero function, v^* is the optimal reward function, and T^k refers to k compositions of operator T . Since $v_0 \in \tilde{V}$ and $T: \tilde{V} \rightarrow \tilde{V}$, we have $T^k v_0 \in \tilde{V}$, and thus $v^* \in \tilde{V}$. \square

Proof of Theorem 2. By Lemma 2, we know $v^* \in \tilde{V}$. Define, for $v^* \in \tilde{V}$,

$$S_1^*(\mathbf{x}_{-1}) = \min \{z_1 : v^*(\mathbf{z} + e_1) - v^*(\mathbf{z}) < c_1, z_k = x_k, \forall k > 1, z_1 \in \mathbb{N}_0\}, \text{ and}$$

$$S_i^*(\mathbf{x}_{-i}) = \min \{x_{i-1}, \min \{z_i : v^*(\mathbf{z} - e_{i-1} + e_i) - v^*(\mathbf{z}) < c_i, z_k = x_k, \forall k \neq i, z_i \in \mathbb{N}_0\}\}, \forall i \in \{2, \dots, m\}.$$

Since v^* satisfies Property 11, $v^*(\mathbf{z} - e_{i-1} + e_i) - v^*(\mathbf{z})$ is nonincreasing in z_i (take $l = q = i$ and $w = m+1$), $\forall i \in \{1, 2, \dots, m\}$. Consider $S_1^*(\mathbf{x}_{-1})$. As z_1 increases, since the holding cost rate h is strictly increasing, $v^*(\mathbf{z} + e_1) - v^*(\mathbf{z})$ will eventually cross c_1 . Thus there exists a finite $S_1^*(\mathbf{x}_{-1})$. Now consider $S_i^*(\mathbf{x}_{-i})$ for $i \in \{2, \dots, m\}$. Notice that, since the holding cost rate h is strictly increasing in the total number of projects, x_{i-1} should be finite. Thus there exists a finite $S_i^*(\mathbf{x}_{-i})$. Hence

the optimal stationary policy is a state-dependent noncongestive promotion policy with state-dependent promote-up-to levels $S_i^*(\mathbf{x}_{-i})$, $\forall i$. Next we will prove properties (i)-(vi):

- i. Pick arbitrary i and j such that $i < j$. We will show that the optimal promote-up-to level in stage i obeys $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$, where $z_{i'} = x_{i'}$, $\forall i' \notin \{i, j\}$, and $z_j + 1 = x_j$. Suppose that $S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. By definition, it is optimal to promote a project to stage i at \mathbf{x} if $x_i < S_i^*(\mathbf{x}_{-i})$, and it is not optimal to do so at \mathbf{z} if $z_i = S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. But we have a contradiction when $x_i = z_i = S_i^*(\mathbf{z}_{-i})$: If it is optimal to promote a project to stage i at \mathbf{x} , it should also be optimal to do so at \mathbf{z} (due to Property 11 when $l = i$, $q = j$, and $w = m + 1$). Thus we must have $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$.
- ii. Pick arbitrary i and j such that $i > j$. We will show that the optimal promote-up-to level in stage i obeys $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$, where $z_{i'} = x_{i'}$, $\forall i' \notin \{i, j\}$, and $z_j = x_j + 1$. Suppose that $S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. By definition, it is optimal to promote a project to stage i at \mathbf{x} if $x_i < S_i^*(\mathbf{x}_{-i})$, and it is not optimal to do so at \mathbf{z} if $z_i = S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. But we have a contradiction when $x_i = z_i = S_i^*(\mathbf{z}_{-i})$: If it is optimal to promote a project to stage i at \mathbf{x} , it should also be optimal to do so at \mathbf{z} (due to Property 12 when $l = 0$, $q = j$, and $w = i$). Thus we must have $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$.
- iii. Pick arbitrary i and j such that $i \neq j + 1$. We will show that the optimal promote-up-to level in stage i obeys $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$, where $z_{i'} = x_{i'}$, $\forall i' \notin \{i, j, j + 1\}$, $z_j + 1 = x_j$ and $z_{j+1} = x_{j+1} + 1$. Suppose that $S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. By definition, it is optimal to promote a project to stage i at \mathbf{x} if $x_i < S_i^*(\mathbf{x}_{-i})$, and it is not optimal to do so at \mathbf{z} if $z_i = S_i^*(\mathbf{z}_{-i}) < S_i^*(\mathbf{x}_{-i})$. But we have a contradiction when $x_i = z_i = S_i^*(\mathbf{z}_{-i})$: If it is optimal to promote a project to stage i at \mathbf{x} , it should also be optimal to do so at \mathbf{z} (due to Property 10 when $q = i$ and $w = j + 1$). Thus we must have $S_i^*(\mathbf{z}_{-i}) \geq S_i^*(\mathbf{x}_{-i})$.
- iv. Suppose that $x_m > 0$. Since v^* satisfies Property 9, it is always optimal to launch a new product: $T_{m+1}v^*(\mathbf{x}) = \max\{v^*(\mathbf{x}), v^*(\mathbf{x} - e_m) + \rho\} = v^*(\mathbf{x} - e_m) + \rho$.
- v. We will prove it is never optimal to interrupt any experiment. Assume that a project is optimally promoted to stage i at a given state \mathbf{x} , or equivalently, an experiment in stage i is optimally initiated at \mathbf{x} ($x_i < S_i^*(\mathbf{x}_{-i})$). We will then consider the following cases:
 - Suppose that $i \geq 2$. Also, suppose that the system moves to a state \mathbf{z} such that $z_1 = x_1 + 1$ and $z_j = x_j$, $\forall j > 1$. Point (ii) implies that it is optimal to promote a project to stage i at \mathbf{z} : The experiment in stage i , which has been initiated at \mathbf{x} , can be resumed at \mathbf{z} .

- Pick arbitrary $j \geq 1$ such that $j + 1 \neq i$ and $j + 1 \leq m$. Suppose that the system moves to a state \mathbf{z} such that $z_j + 1 = x_j$, $z_{j+1} = x_{j+1} + 1$, and $z_{j'} = x_{j'}$, $\forall j' \notin \{j, j + 1\}$. Point (iii) implies that it is optimal to promote a project to stage i at \mathbf{z} : Again, the experiment in stage i , which has been initiated at \mathbf{x} , can be resumed at \mathbf{z} .
- Suppose that the system moves to a state \mathbf{z} such that $z_m + 1 = x_m$ and $z_j = x_j$, $\forall j < m$. Also, suppose that $i < m$. Point (i) implies that it is optimal to promote a project to stage i at state \mathbf{z} : Once again, the experiment in stage i , which has been initiated at \mathbf{x} , can be resumed at \mathbf{z} . Next suppose that $i = m$: It is optimal to promote a project to stage m at state \mathbf{z} , since $z_m < x_m < S_m^*(\mathbf{x}_{-m}) = S_m^*(\mathbf{z}_{-m})$. Once again, the experiment in stage i , which has been initiated at \mathbf{x} , can be resumed at \mathbf{z} .

Therefore the experiment in stage i is never interrupted.

- vi. As v^* satisfies Property 8, if a project is to be terminated, it is optimal to choose this from the earliest possible stage. Pick arbitrary state \mathbf{x} . Let i^* denote the earliest stage with at least one available project. Thus $x_i = 0$, $\forall i < i^*$. Suppose that it is not optimal to terminate a project from category i^* at state \mathbf{x} , i.e., $v^*(\mathbf{x}) \geq v^*(\mathbf{x} - e_{i^*})$. Then we consider the following scenarios:

- Suppose that the system moves to a state \mathbf{z} such that $z_1 = x_1 + 1$ and $z_j = x_j$, $\forall j > 1$. This implies that $v^*(\mathbf{x} + e_1) - c_1 \geq v^*(\mathbf{x})$. Since $c_1 \geq 0$, we should have $v^*(\mathbf{x} + e_1) \geq v^*(\mathbf{x})$. Thus it is not optimal to terminate a project in stage 1. As v^* satisfies Property 8, we should also have $v^*(\mathbf{x} + e_1) - v^*(\mathbf{x} + e_1 - e_{i^*}) \geq v^*(\mathbf{x} + e_1) - v^*(\mathbf{x}) \geq 0$. Thus it is not optimal to terminate any project in stage i^* .
- Suppose that the system moves to a state \mathbf{z} such that $z_{i^*} + 1 = x_{i^*}$, $z_{i^*+1} = x_{i^*+1} + 1$, and $z_j = x_j$, $\forall j \notin \{i^*, i^* + 1\}$. This implies that $v^*(\mathbf{x}) \leq v^*(\mathbf{z}) - c_{i^*+1}$. Also, suppose that $x_{i^*} \geq 2$. Since $v^*(\mathbf{x} - e_{i^*}) \leq v^*(\mathbf{x})$ and v^* satisfies Property 12 (take $l = 0$ and $q = w - 1$), we should have $v^*(\mathbf{z}) - v^*(\mathbf{z} - e_{i^*}) \geq v^*(\mathbf{x}) - v^*(\mathbf{x} - e_{i^*}) \geq 0$. Thus it is not optimal to terminate any project in stage i^* . Now suppose that $x_{i^*} = 1$. Notice that $z_i = 0$, $\forall i \leq i^*$. If a project is to be terminated at \mathbf{z} , it is optimal to select this from stage $i^* + 1$. But it is not optimal to terminate such a project: Since $v^*(\mathbf{x} - e_{i^*}) \leq v^*(\mathbf{x})$ and $v^*(\mathbf{x}) \leq v^*(\mathbf{z}) - c_{i^*+1}$, we should have $v^*(\mathbf{z} - e_{i^*+1}) = v^*(\mathbf{x} - e_{i^*}) \leq v^*(\mathbf{z})$.
- Pick arbitrary j such that $j > i^*$ and $j < m$. Suppose that the system moves to a state \mathbf{z} such that $z_j + 1 = x_j$, $z_{j+1} = x_{j+1} + 1$, and $z_{j'} = x_{j'}$, $\forall j' \notin \{j, j + 1\}$. Since $v^*(\mathbf{x} - e_{i^*}) \leq v^*(\mathbf{x})$ and v^* satisfies Property 12 (take $l = 0$ and $q < w - 1$), we should have $v^*(\mathbf{z}) - v^*(\mathbf{z} - e_{i^*}) \geq v^*(\mathbf{x}) - v^*(\mathbf{x} - e_{i^*}) \geq 0$. Thus it is not optimal to terminate any project in stage i^* .

- Suppose that the system moves to a state \mathbf{z} such that $z_m + 1 = x_m$ and $z_j = x_j, \forall j < m$. Also, suppose that either $i^* < m$ or $i^* = m$ and $x_m \geq 2$. Since $v^*(\mathbf{x} - e_{i^*}) \leq v^*(\mathbf{x})$ and v^* satisfies Property 12 (take $l = 0$ and $w = m + 1$), we should have $v^*(\mathbf{z}) - v^*(\mathbf{z} - e_{i^*}) \geq v^*(\mathbf{x}) - v^*(\mathbf{x} - e_{i^*}) \geq 0$. Thus it is not optimal to terminate any project in stage i^* . Now suppose that $i^* = m$ and $x_m = 1$. But then $z_i = 0, \forall i$, i.e., there is no project in the system.

Therefore it is never optimal to terminate any project during the NPD process. \square

EC.4. Proofs of the Results in Section 5

In Section 5, as our optimization criterion, we take the average reward rate over an infinite planning horizon. Given a policy $\ell = \tilde{\ell}$, the average reward rate is given by

$$v^{\tilde{\ell}}(\mathbf{x}) = \liminf_{T \rightarrow \infty} \frac{1}{T} \left\{ \sum_{j \in \mathcal{W}_{m+1}} \int_0^T \rho_j dN_j(t) - \sum_{i \in \mathcal{M}} \int_0^T c_i dN^{(i)}(t) - \int_0^T h(\mathbf{X}(t)) dt \right\}. \quad (\text{EC.1})$$

The objective is to identify a policy ℓ^* that yields $v^*(\mathbf{x}) = \sup_{\ell} v^{\ell}(\mathbf{x})$ for all states \mathbf{x} . The following proposition shows that our optimal structural results in Sections 3 and 4 carry over to the average reward case:

PROPOSITION EC.1. *Suppose that Assumption 1 (or Assumption 2) holds. Then there exists a stationary policy that is optimal under the average reward criterion. This policy retains all the properties of the optimal policy under the discounted reward criterion, as introduced in Theorem 1 (or Theorem 2). Also, the optimal average reward is finite and independent of the initial state; there exists a finite constant v^* such that $v^*(\mathbf{x}) = v^*, \forall \mathbf{x}$.*

Proof of Proposition EC.1. Suppose that Assumption 1 holds. We first prove the following conditions: (i) There exists a stationary policy ℓ that induces an irreducible positive recurrent Markov chain with finite average reward v^{ℓ} , and (ii) the number of states for which $-h(\mathbf{x}) \geq \gamma$ is finite for any negative value γ . Consider a policy where a new product idea is tested if and only if the number of projects in the system is less than a fixed threshold, and a new product is always launched from the most promising project available. Note that we have a finite-state Markov chain under this policy. Thus this policy yields a finite average reward, and condition (i) holds. Since the holding cost rate is increasing convex in the number of projects in the system, condition (ii) holds as well. Under conditions (i) and (ii), there exists a constant v^* and a function $f(\mathbf{x})$ such that $f(\mathbf{x}) + v^* = \mu_1 T_1 f(\mathbf{x}) + \mu_2 T_2 f(\mathbf{x}) - h(\mathbf{x})$ (Weber and Stidham 1987). The stationary policy that maximizes the righthand side of the above equation for each state is an optimal policy in

the average reward case, yielding a constant average reward v^* . Thus properties of the optimal policy in the average reward case are determined through $f(\mathbf{x})$, whereas properties of the optimal policy in the discounted reward case are determined through $v^*(\mathbf{x})$. Since the same event operators appear in either case, the optimal policy for the average reward retains the same structure as in Theorem 1.

Now suppose that Assumption 2 holds. We again prove conditions (i) and (ii). Consider a policy where a project is advanced to the next stage if and only if the number of projects in the next stage is less than a fixed threshold, projects are never terminated, and a new product is always launched if feasible. Again, we have a finite-state Markov chain under this policy. Thus this policy yields a finite average reward, and condition (i) holds. Since the holding cost rate is increasing convex in the number of projects in the system, condition (ii) holds as well. Hence there exists a constant v^* and a function $f(\mathbf{x})$ such that $f(\mathbf{x}) + v^* = \lambda T_0 f(\mathbf{x}) + \sum_{1 \leq i \leq m} \mu_i T_i f(\mathbf{x}) + \mu_{m+1} T_{m+1} f(\mathbf{x}) - h(\mathbf{x})$. The stationary policy that maximizes the righthand side of the above equation for each state is an optimal policy in the average reward case, yielding a constant average reward v^* . Thus properties of the optimal policy in the average reward case are determined through $f(\mathbf{x})$, whereas properties of the optimal policy in the discounted reward case are determined through $v^*(\mathbf{x})$. Since the same event operators appear in either case, the optimal policy for the average reward retains the same structure as in Theorem 2. \square

EC.5. Additional Numerical Results

Tables EC.1-EC.3 exhibit our numerical results when the number of projects in the NPD process should never exceed six, i.e., $\bar{x} = 6$.

Table EC.1 Numerical results for various experimentation and product launch rates.

μ_1	μ_2	μ_3	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
				SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.5	0.5	0.5	2.700	> 100	0.074	2.222	3600	10.09	2.05
-	-	1.0	3.371	0.000	0.030	0.326	2228.75	9.43	1.85
-	-	1.5	3.534	0.000	0.000	0.113	1678.26	9.77	1.59
-	1.0	0.5	3.331	0.000	0.120	0.150	2899.61	7.76	1.83
-	-	1.0	4.326	0.000	0.000	0.000	1161.73	10.55	1.85
-	-	1.5	4.628	0.000	0.000	0.000	1623.22	8.49	2.01
-	1.5	0.5	3.509	0.000	0.171	0.171	2603.50	12.78	1.37
-	-	1.0	4.641	> 100	0.000	0.000	3600	8.08	2.28
-	-	1.5	5.009	0.000	0.000	0.000	2217.71	12.84	1.76
1.0	0.5	0.5	3.542	> 100	1.525	21.062	3600	10.83	2.05
-	-	1.0	4.772	2.242	0.000	10.583	3600	8.40	2.05
-	-	1.5	5.054	0.020	0.000	10.051	3600	6.66	1.50
-	1.0	0.5	4.608	> 100	3.060	7.791	3600	8.92	1.35
-	-	1.0	7.293	0.000	0.000	0.000	808.19	7.23	1.96
-	-	1.5	8.051	0.000	0.000	0.000	2330.31	6.10	1.60
-	1.5	0.5	4.893	0.020	3.022	6.193	3169.36	13.75	1.34
-	-	1.0	8.023	> 100	0.000	0.000	3600	6.65	1.79
-	-	1.5	9.037	0.000	0.000	0.000	1950.66	5.56	1.77
1.5	0.5	0.5	3.832	> 100	5.271	35.360	3600	10.09	1.67
-	-	1.0	5.318	34.393	0.094	23.825	3600	8.13	1.80
-	-	1.5	5.621	> 100	0.107	23.305	3600	7.70	1.70
-	1.0	0.5	5.054	> 100	5.639	20.439	3600	14.06	1.75
-	-	1.0	8.974	0.000	0.011	0.501	1923.07	8.00	1.97
-	-	1.5	10.255	> 100	0.000	0.107	3600	7.83	1.81
-	1.5	0.5	5.429	> 100	5.102	18.714	3600	7.38	1.51
-	-	1.0	10.177	0.000	0.000	0.020	1989.07	8.33	1.91
-	-	1.5	12.124	0.000	0.000	0.000	601.81	4.22	1.50
Average				71.036	0.897	6.701	2740.19	8.88	1.76

$h = 1$, $c_1 = c_2 = 4$, $r_1 = 40$, $r_2 = 0$, $\lambda = 100$, $p_{0,1} = p_{0,2} = 0.5$, $\phi^{(1)} = \phi^{(2)} = 0.75$, $\bar{x} = 6$.

Computation times equal to 3600 seconds indicate termination of the algorithm.

Table EC.2 Numerical results for various holding and experimentation costs.

h	c_1	c_2	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
				SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.5	2	2	8.679	0.000	0.000	0.000	1464.32	11.73	2.36
-	-	4	7.631	0.000	0.000	0.000	2268.98	11.57	2.10
-	-	6	6.694	0.015	0.015	0.015	1894.06	9.58	1.52
-	4	2	6.843	0.000	0.000	0.044	1390.73	16.02	2.38
-	-	4	5.913	0.000	0.034	0.085	2448.34	8.66	1.58
-	-	6	5.044	9.358	0.278	0.377	3600	10.22	1.74
-	6	2	5.253	26.156	0.876	1.123	3600	14.84	1.65
-	-	4	4.313	> 100	1.159	1.530	3600	9.71	2.01
-	-	6	3.451	0.000	2.028	2.695	2071.95	10.54	1.79
1.0	2	2	7.154	0.000	0.140	0.405	2931.48	14.29	1.82
-	-	4	6.190	0.000	0.468	1.179	2620.58	9.54	1.35
-	-	6	5.321	> 100	1.522	3.251	3600	8.15	1.00
-	4	2	5.437	> 100	1.471	3.587	3600	18.15	1.51
-	-	4	4.608	> 100	3.060	7.791	3600	8.72	1.60
-	-	6	3.826	> 100	3.764	12.467	3600	14.14	1.18
-	6	2	3.991	> 100	4.711	14.107	3600	16.55	1.46
-	-	4	3.185	> 100	6.750	20.126	3600	11.38	1.67
-	-	6	2.443	> 100	5.649	30.864	3600	10.21	1.50
1.5	2	2	5.979	1.104	1.037	3.596	3600	9.67	1.21
-	-	4	5.087	> 100	1.533	5.740	3600	7.94	0.78
-	-	6	4.348	> 100	3.818	9.890	3600	14.65	0.61
-	4	2	4.420	> 100	4.027	12.240	3600	17.71	1.72
-	-	4	3.666	12.057	5.674	21.004	3600	11.77	0.86
-	-	6	2.995	> 100	5.042	33.923	3600	12.67	0.90
-	6	2	3.087	> 100	7.872	34.532	3600	15.76	1.30
-	-	4	2.363	7.787	4.486	55.650	3600	13.22	1.06
-	-	6	1.769	> 100	5.596	92.482	3600	13.40	0.89
Average				> 100	2.630	13.656	3166.31	12.25	1.46

$r_1 = 40, r_2 = 0, \lambda = 100, \mu_1 = \mu_2 = 1, \mu_3 = p_{0,1} = p_{0,2} = 0.5, \phi^{(1)} = \phi^{(2)} = 0.75, \bar{x} = 6.$

Computation times equal to 3600 seconds indicate termination of the algorithm.

Table EC.3 Numerical results for various levels of accuracy in experimental stages.

$\phi^{(1)}$	$\phi^{(2)}$	Optimal reward	Percentage difference from optimal reward			Computation times (in seconds)		
			SDNCP	FNCP	CP	SDNCP	FNCP	CP
0.55	0.55	2.818	70.085	4.400	61.320	3600	10.20	1.62
-	0.65	2.818	> 100	4.400	57.488	3600	12.81	1.39
-	0.75	3.053	> 100	2.588	35.637	3600	6.36	1.58
-	0.85	3.758	> 100	2.874	18.680	3600	9.60	2.45
-	0.95	4.776	> 100	1.340	9.590	3600	9.05	1.93
0.65	0.55	3.295	33.384	7.284	21.275	3600	11.94	1.19
-	0.65	3.335	> 100	8.006	21.769	3600	10.12	1.36
-	0.75	3.423	> 100	7.216	15.542	3600	12.61	1.71
-	0.85	3.930	> 100	3.893	6.997	3600	8.60	1.55
-	0.95	4.885	> 100	1.965	3.234	3600	9.26	2.18
0.75	0.55	4.594	12.495	2.982	8.663	3600	6.60	0.98
-	0.65	4.597	> 100	3.002	8.680	3600	11.44	1.19
-	0.75	4.608	> 100	3.060	7.791	3600	8.78	1.44
-	0.85	4.831	> 100	1.056	2.256	3600	7.91	1.91
-	0.95	5.553	0.018	0.540	0.882	1958.08	7.54	1.59
0.85	0.55	6.078	2.534	2.353	3.883	3600	9.56	0.94
-	0.65	6.078	0.000	2.353	3.883	2702.42	6.23	0.99
-	0.75	6.078	0.707	2.353	3.883	3600	9.94	1.01
-	0.85	6.136	0.359	0.913	1.679	3600	7.25	1.15
-	0.95	6.574	0.000	0.411	0.578	800.07	7.32	1.61
0.95	0.55	7.640	0.052	1.283	1.702	3600	6.73	1.21
-	0.65	7.640	0.851	1.283	1.702	3600	7.34	1.29
-	0.75	7.640	0.000	1.283	1.702	2939.76	6.39	1.38
-	0.85	7.640	0.000	1.283	1.702	3036.21	7.99	1.16
-	0.95	7.740	0.000	0.736	1.021	1457.61	6.60	1.26
Average			91.340	2.754	12.061	3251.77	8.73	1.44

$h = 1$, $c_1 = c_2 = 4$, $r_1 = 40$, $r_2 = 0$, $\lambda = 100$, $\mu_1 = \mu_2 = 1$, $\mu_3 = p_{0,1} = p_{0,2} = 0.5$, $\bar{x} = 6$.

Computation times equal to 3600 seconds indicate termination of the algorithm.

References

- Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons, New York, NY.
- Weber, R. R., S. Stidham. 1987. Optimal control of service rates in networks of queues. *Adv. Appl. Probab.* **19** 202–218.