A network of options: Evaluating complex interdependent decisions under uncertainty

Takashi Akamatsu a, Takeshi Nagae b,*

a Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan
b Graduate School of Information Systems, The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan

A R T I C L E I N F O

Article history:
Received 3 February 2010
Accepted 17 December 2010
Available online 12 January 2011

JEL classification:
C61
C63
D92
E22
G31
G32

Keywords:
Compound real options
Managerial flexibility
Graph theory
Singular stochastic control problems
Generalized complementarity problems

A B S T R A C T

The present article provides a novel framework for analyzing option network problems, which is a general class of compound real option problems with an arbitrary combination of reversible and irreversible decisions. The present framework represents the interdependent structure of decisions by using a directed graph. In this framework, the option network problem is formulated as a singular stochastic control problem, whose optimality condition is then obtained as a dynamical system of generalized linear complementarity problems (GLCPs). This enables us to develop a systematic and efficient numerical method for evaluating the option value and the optimal decision policy.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Business is a complex chain of decisions for achieving desirable cash flow streams from real investment projects. As uncertainty about future rewards from investments is gradually resolved by the arrival of new information, managerial flexibility with state-dependent decision policy significantly enhances the project values (Kogut and Kulatilaka, 1994). Managerial flexibility in a project is restricted by the project-specific decision making structure, which usually forms a complex combination of irreversible decisions and (fully or partially) reversible decisions: an R&D project can neither move back to earlier stages nor skip inevitable stages; network servers of a large size company can be started and/or expanded within an arbitrary operating system but can only with difficulty be shifted to another operating system after the first choice; constructions of huge infrastructures (e.g. dams, bridges and tunnels) are almost impossible to be redone but their operation and maintenance could be somewhat flexible, and so on. Real investment problems of valuing such managerial flexibility and of finding an appropriate decision policy have been key issues in management science/operations research as well as economics and finance, and there is a vast amount of literature on the subject.

The existing approaches to analyzing complex real investment problems under uncertainty can be roughly classified into two categories: the decision tree/influence diagram (DT/ID) approach and the real option approach. The DT/ID approach is the most...
conventional method for analyzing the real investment problem. It uses a directed graph, referred to as either the decision tree or the influence diagram, to represent the interdependence among underlying random factors, managerial decisions, and rewards embedded in a project, (see, for example, Raiffa, 1968; Howard and Matheson, 1984; Smith and von Winterfeldt, 2004). Although this DT/ID approach provides an intuitive and flexible representation of the decision making problems, it seems difficult to apply it to problems where relevant factors include sequential decisions in conjunction with the resolution of uncertainties over time. The directed graph (and the values of the value and the conditional probabilities) constructed by the DT/ID approach ‘explodes’ when it is applied to problems with dozens of time periods, at each of which new information arrives and decisions can be made. This also implies that the DT/ID approach essentially cannot be incorporated with the financial market models and the theory of financial engineering established in a continuous-time framework, e.g., the dynamic portfolio theory, the term structure model, and others. Furthermore, the DT/ID representation of the timing choice problem can be regarded as the explicit payoff approach explained below, and thus it also inherits the discrete-continuous convergence problem.

The real option approach, on the other hand, was introduced by Brennan and Schwartz (1985) and McDonald and Siegel (1985) as a natural extension of the theory for American options. This approach is thus integrable with continuous-time financial engineering theory and models. The existing real option approaches, however, have critical practical disadvantages. First, the existing real option models are too simple to capture the complex decision making structure in ‘real’ investment projects. Most of the existing real option models are straightforward extensions of ‘plain-vanilla’ American options, in which there is a single option and no future decisions can be made after the option is carried out. Although there are several ‘mixed-flavor’ real option models, as shown in Section 2.2, each is limited to its own specified application and is still insufficiently generalized.

Second, existing approaches for obtaining closed-form solutions of real option problems cannot be used to analyze practical real investment problems of the type that concern us. For example, one of the most widely used such analytical approaches, the so-called value-matching and smooth-pasting (VM-SP) approach, is especially useful for qualitative analyses because it reduces a real option problem to a tractable nonlinear equation system referred to as the value-matching and the smooth-pasting conditions (see, e.g., Dixit, 1993; Dixit and Pindyck, 1994). This reduction, however, is available only for models with an infinite time horizon and some specific underlying stochastic processes and payoff functions. Third and finally, there are no numerical solution methods that are applicable to various real investment problems. In the real option literature, one of the most widely used numerical approaches is to reduce the optimality condition (i.e. the Hamilton–Jacobi–Bellman equation) to a problem of simply taking the maximum of the continuation value and the option payoff. We refer to this as the explicit payoff approach in this paper. Although this approach is intuitive and easy to implement, it causes a serious inconsistency with the optimal condition when it naïvely applied to real option problems with reversible decisions such as the entry–exit options (Dixit, 1989). This inconsistency is inevitable because it approximates the continuation value by using either the Euler explicit finite difference scheme (Brennan and Schwartz, 1977) or the multinomial tree/lattice model (Cox et al., 1979; Trigeorgis, 1991). Furthermore, the explicit payoff approach would be inappropriate even for the plain-vanilla real options as it might face the so-called discrete-continuous convergence problem: The approximate solutions obtained from a discretized model need not converge to their continuous counterpart unless certain conditions are satisfied, but satisfying these conditions can make the computational procedure very demanding. For a more detailed discussion of these deficiencies in the existing approaches, readers are referred to Nagae and Akamatsu (2008).

 Accordingly, to the best of our knowledge, there is neither a systematic framework for modeling real investment problems with complex decision structure nor an efficient and accurate numerical solution method widely applicable to real investment problems without restrictive assumptions.

The present paper provides a novel unified framework for analyzing decision making problems under dynamic uncertainty with a complex combination of reversible and irreversible decisions. It has the following three desirable aspects, each of which is necessary for quantitative analyses of practical real investment problems: (i) The modeling framework can represent in a unified manner a wide variety of decision making problems, ranging from rather simple examples to more complicated and realistic real investment problems. (ii) The analysis method is universally applicable to various models with general settings (e.g. a finite time horizon model with state variables following generalized Itô processes and arbitrary payoff functions). (iii) The solution method is accurate and computationally efficient for large size problems. Surprisingly, no earlier method satisfies even one of these aspects despite the vast literature in the field of management science, operations research, finance and economics. The major contribution of the present paper is to provide a universal tool for practical implementation of real options with complex structures that are much more general than hitherto could be dealt with, exploiting the directed graph not only for systematic representation of compound option structure but also for development of an efficient solution algorithm.

We first introduce a general class of real option problems referred to as the option network. It is defined as a set of decision opportunities (or ‘options’) to change the economic ‘activities’ in accordance with the resolution of uncertainties. This definition is quite general and can capture a wide variety of decision making problems. We present a framework that represents an option network as a directed graph: each node indicates an ‘activity’ (or an operation mode of the project); each directed link indicates an ‘option’ (or a decision opportunity) to switch from one activity (the predecessor of the link) to another (the successor). This graph representation gives a complete intuitive description of the decision structure. For example, suppose a firm is considering a two-stage investment for a new production line, whose decision structure is depicted in Fig. 1. This figure shows that the firm can choose one of the two products A and B in the first stage, and can then change the production size between the three alternatives, small(s), medium(m) and large(l) in the second stage. This figure also shows that the production type cannot be changed after the first choice, whereas the production size can be changed any number of times. In other words, the firm has an option to ‘wait for the first (and the last) choice of the production type’ in the first stage and has an option to ‘alter the production size’ in the second stage; the former option is killed after the production type is chosen, whereas the latter will never be eliminated.
Second, we provide a universal method for analyzing option network problems with arbitrary decision structure. It is a natural but nontrivial extension of the generalized linear complementarity problem (GLCP) approach introduced by Nagae and Akamatsu (2008), which is the latest and perhaps the most promising for analyzing real/financial option problems with timing choice in terms of accuracy, efficiency and applicability. It achieves both efficiency and accuracy without restrictive assumptions about underlying processes and payoff functions. In addition, to the best of our knowledge, the GLCP approach is the only method that can be applied to (rather simple) real option problems with reversible decisions in a finite time horizon. We first formulate the option network problem as a singular stochastic control problem of finding the optimal activity switching policy that maximizes the expected net present value of the cash flow streams. We then reveal that the optimality condition can be reformulated as a dynamical system of generalized linear complementarity problems (GLCPs).

Finally, we use the above to develop a new numerical method for evaluating the option value and the optimal decision strategy. It is accurate and efficiently applicable to large-size problems. We first approximate the GLCP system in an appropriate discrete time-state space and then decompose the discretized GLCP system with respect to time, where each subproblem is formulated as a finite-dimensional GLCP. We then show that the time-decomposed subproblems can be further decomposed with respect to substructures of the directed graph: we reveal that each time-decomposed subproblem reduces to a problem of solving a succession of much smaller sub-subproblems, which is analogous to the shortest-path finding algorithms. This decomposition property of the option network problems enables us to construct an efficient solution algorithm, which includes the solution method developed in Nagae and Akamatsu (2008) as an inevitable subprocedure.

The rest of the present article is organized as follows: Section 2 presents the modeling framework of option networks. It shows how existing real option problems in the literature can be represented in our framework. In Section 3, a rigorous formulation of the option network problem is given and its optimality condition is obtained. Sections 4 and 5 show how the optimality condition is decomposed with respect to time and the graph substructure. Section 6 sketches the whole algorithm for solving the option network problem. Section 7 uses an example to illustrate the option network and the numerical solution method, also demonstrating its efficiency. Section 8 gives concluding remarks.

2. Model

In this section, we first represent the interdependent decision structure of an option network as a directed graph. We then show how to represent the existing real option models as special cases of our option network. Finally, we formulate the option network problem mathematically as a singular stochastic control problem.

2.1. Graph representation of interdependent structure of options

Suppose a decision maker can choose one of the \( N \) economic activities at some moment of time. Let \( N = \{1, \ldots, N\} \) be the set of activities. We then denote an option to switch the activity from \( n \) to \( m \) by a directed link as in Fig. 2. Let \( L \) be the set of links and \( |L| = m \). We refer to the set of these activities and options as an option network and denote it by a directed graph \( G(N, L) \).

2.2. The option network representation of the existing option problems

The present option network model includes most of the existing option problems in the literature as special cases, each of which corresponds to basic substructures of a directed graph: (a) a single link structure; (b) branch structures; (c) tandem structures; and (d) cyclic structures. This section sketches the basic characteristics of these submodels.

(a) A single link structure (‘plain-vanilla’ options): If there is only a single option to invest (or abandon, expand, contract, etc.) and no future decision making can take place after the option is exercised, it is described as an option network with two activities.
and a single link as depicted in Fig. 2. Most of the literature on real options is in this category, for example, single options to invest (or to defer investing) (Titman, 1985; McDonald and Siegel, 1986; Ingersoll and Ross, 1992), to abandon (Myers and Majd, 1990), to alter operating scale (Trigeorgis and Mason, 1987) and other primitive real option problems explained in introductory textbooks such as Dixit and Pindyck (1994), Trigeorgis (1996), Brennan and Trigeorgis (2000) and Schwartz and Trigeorgis (2001). These real options can be regarded as straightforward conversions of simple plain-vanilla American options that have been studied in the field of financial option pricing (see e.g., McKean, 1965; Merton, 1973, 1990; Brennan and Schwartz, 1977; Bensoussan, 1984; Jaillet et al., 1990; Broadie and Detemple, 2004, and references therein).

(b) Diverging structures (choosing one from many options): Suppose the decision maker can choose one from several alternatives and the decision is irreversible: after one alternative is chosen, no future changes will be made to that decision. This decision opportunity can be described as an option network with a diverging structure as depicted in Fig. 3. The diverging structured option model captures the capacity choice options (Pindyck, 1988; He and Pindyck, 1992; Bar-Ilan et al., 2002) in the real option literature. For financial options, the exchange option (Margrabe, 1978) and the minimum/maximum option (Stulz, 1982) are also covered by this model.

(c) Tandem structures (staged options): When a real investment project consists of a set of sequential opportunities, each of which becomes available after its preceding option is exercised, it is described as an option network with tandem structure shown in Fig. 4. The tandem structured option model covers the (simple) compound options (Geske, 1979) and the sequential exchange options (Carr, 1988) in the financial field, and the sequential investment options (Dixit and Pindyck, 1994, Chap. 10) in the real option field. This model also can deal with the ‘time-to-build’ options (Majd and Pindyck, 1987; Bar-Ilan and Strange, 1996; Bar-Ilan et al., 2002).

It is noteworthy that, just by combining the tandem structure and the diverging structure, we can model various decision structures, e.g., the multi-option model studied by Trigeorgis (1991, 1993) as shown in Fig. 5.

(d) Cyclic structures (reversible options): In practice, it is often necessary to analyze real investment projects having partially (or costly) reversible decisions. One of the simplest options involving reversible decisions is the entry–exit option introduced by Dixit (1989) and Dixit and Pindyck (1994, Chap. 7), where the production project can be shut down if market conditions are less favorable, and can be restarted if market conditions are more favorable. This infinitely repeatable decision problem can be modeled using an option network with cyclic structure as shown in Fig. 6.
As (partial) irreversibility is one of the key features of real option problems, by combining the cyclic option with the above irreversible options we can model a wide variety of interesting applications. First, the entry–exit option with costly lay-up and reactivation Dixit and Pindyck (1994, Chap. 7) can be depicted as in Fig. 7. Second, when we suppose a 'complete graph,' in which each node is connected to every other node (i.e. it is possible to switch from any activity to any other activity), the option network becomes equivalent to the (complete) managerial flexibility models studied by Kulatilaka (1995) and Triantis and Hodder (1990).

2.3. Formulation

Suppose an option network has duration \([0,T]\), where \(T\) is a given constant, referred to as the maturity of the option network. We assume that the economic environment causing the dynamic uncertainty of the cash flow streams is described by a \(K\)-tuple state variable vector and denote its value at time \(t \in [0,T]\) by \(P(t) = [P_1(t), \ldots, P_k(t)]^T\). Components of the state variable may represent the exchange rate, the output price, the daily operation cost of the plant, the demand for the output, and so on. The state variable gives a stochastic process \(P : [0,T] \times \Omega \rightarrow \mathbb{R}^K\), which is defined on the probability space \((\Omega, \mathcal{F}, \mathbb{P})\); \(\Omega\) is the state space, \(\mathcal{F}\) is a \(\sigma\)-field on \(\Omega\), and \(\mathbb{P}\) is the probability measure on \((\Omega, \mathcal{F})\). Let us suppose that the activity is \(n(t)=n\) and the state variable is \(P(t) = P\) at time \(t \in [0,T]\). The \(k\)th component of the state variable is then assumed to follow an Itô diffusion process:

\[
dP_k(t) = z_k(t, P, n)dt + \sum_{z=1}^Z \sigma_{k,z}(t, P, n)dW_z(t), \quad P_k(0) \text{ is given,}
\]

where \(z_k : [0,T] \times \mathbb{R}^K \times N \rightarrow \mathbb{R}^d\) and \(\sigma_{k,z} : [0,T] \times \mathbb{R}^K \times N \rightarrow \mathbb{R}^d_+\) are known functions that satisfy appropriate Lipschitz conditions. \(W_z(t), \ldots, W_n(t)\) are \(Z\)-dimensional standard Brownian motions defined on \((\Omega, \mathcal{F}, \mathbb{P})\) with independent increments, i.e. \(\mathbb{E}[dW_z(t)dW_z(t)] = 0, \forall z \neq z'\), where \(\mathcal{F}(t)\) is an appropriate filtration on \((\Omega, \mathcal{F})\). It should be noted that the present framework admits that the drift and the variation of the underlying processes, \([z_k]\) and \([\sigma_{k,z}]\), can be determined not only by the current time and state but also the current activity. This implies that the state variables could be indirectly controlled by switching activities.

The cash flow streams of the option network are classified into three categories: (a) instantaneous profits; (b) terminal payoff and (c) switching costs. At each time \(t\), the decision maker yields an instantaneous profit per unit time. When the state variable is \(P(t) = P\) and the activity is \(n(t) = n\) at \(t\), the instantaneous profit is denoted by \(\pi(t, P, n)\), where \(\pi : [0,T] \times \mathbb{R}^K \times N \rightarrow \mathbb{R}\) is a known function. The instantaneous profit could be negative. The decision maker also yields a lump-sum profit (or cost if it is negative) at the maturity \(T\). We denote a decision policy during \([t,T]\) by

\[
f^T(t) \triangleq \{n(t, P, n^-))(t, P, n^-) \in [t,T] \times \mathbb{R}^K \times N\}.
\]

We say that a decision policy \(f^T(t)\) is feasible if and only if

\[
n(t, P, n^-) \in \pi(n^-) \cup [n^-], \quad \forall n(t, P, n^-) \in f^T(t),
\]

where \(\pi(n^-) \triangleq \{m(n^-,m) \in L\}\) is a set of alternatives subsequent to activity \(n^-\).

Suppose that the state variable is \(P(t) = P\) and activity \(n(t) = n\) is chosen at time \(t \in [0,T]\). The net present value (NPV) of cash flow streams during \([t,T]\) under a feasible decision policy \(f^T(t)\) with respect to a sample path \(\omega \in \Omega\) is then defined as

\[
\mathcal{F}(t, f^T(t), \omega) \triangleq \int_t^T \beta(t,s)\pi(s, P, n, s, \omega; f^T(s))ds - \int_t^T \beta(t,s)\sum_{n, m}C_{n,m}\delta_{n,m}(s, \omega; f^T(s))ds + \beta(t,T)\Pi[P(T), n(T, \omega; f^T(T))].
\]
where $n(s, o; \phi^T)$ is the activity chosen at $s$ along sample path $o$ under $\phi^T$; $\delta_{n,m}(s, o; \phi^T) = 1$ if the activity is switched from $s$ to $m$ during $[s, s + ds]$ on sample path $o$ under the strategy $\phi^T$, and 0 otherwise; $\beta(t, s) = e^{f_n(t, s) ds}$ is a (deterministic) discount factor from time $s$ to $t$ and $r : [0, T] \rightarrow R_+$ is an instantaneous discount rate, which is a given function with respect to time. In Eq. (4), the terms on the right hand side represent the present value of the instantaneous profits, the switching cost, and the terminal payoff, respectively.

The option network problem is formulated as the following stochastic control problem:

\[
\max_{\phi_t \in \Phi_{nt}} \mathbb{E}[J(0, T, \phi^T, o)|(0, P_0, n_0)]
\]

subject to Eq. (1),

where $\Phi_{nt}$ is a set of feasible decision policies for duration $[t, T]$. $\mathbb{E}[:|(t, P, n)]$ is an expectation conditional to the information set, $P(t) = P$ and $n(t) = n$, available at $t$. $P_0$ and $n_0$ are the initial values of the state variable and the initial activity, respectively.

### 3. The optimality condition

This section derives the optimality condition of the option network problem (P) as a dynamical system of GLCPs. We first define the value function of (P) as

\[
V(t, P, n) \triangleq \max_{\phi_t \in \Phi_t} \mathbb{E}[J(t, T, \phi^T, o)|(t, P, n)],
\]

where $J(\cdot)$ is defined by (4). We can interpret $V(t, P, n)$ as the value of the project operating with activity $n(t) = n$ at $t$, when the state variable is $P(t) = P$, and hence we refer to $V(t, P, n)$ as the value of activity $n$ (under the condition $(t, P)$).

In what follows, we first derive the optimality condition of the project operating with activity $n$ at time $t$ upon changing the activity from $n$ to the one of the succeeding alternatives, say, $m$. Since the value $V(t, P, n)$ of activity $n$ depends on those of succeeding activities, $V(t, P, m)|m$, the optimality conditions for all activity must be combined. So, we finally formulate the combined optimal conditions that hold at each moment of time $t$ as an infinite-dimensional GLCP.

Let us suppose that the project is operated with activity $n(t) = n$ and the state variable is $P(t) = P$ at time $t$. By applying the dynamic programming (DP) principle, only one of two actions is taken in this situation: either change the activity from

\[
\Delta V(t, P, n) \triangleq V(t + \Delta, P(t + \Delta), n) - V(t, P, n),
\]

Equality holds in (6) when activity $n$ is maintained for $\Delta$. Taking $\Delta \rightarrow 0$ and using Ito’s lemma, we obtain

\[
F_n(t, P, V) = -L_n V(t, P, n) - \pi(t, P, n) \geq 0,
\]

where $F \triangleq \{V(t, P, n)|(t, P, n) \in [0, T] \times R^K \times N\}$ and $L_n$ is the partial differential operator for activity $n$ defined by

\[
L_n \triangleq \frac{\partial}{\partial t} + \sum_{k=1}^{K} \sum_{l=1}^{K} \sigma_{kl}(t, P, n) \frac{\partial}{\partial P_k} - r(t),
\]

where $\sigma_{kl}(t, P, n) = \sum_{i=1}^{L} \sigma_{ikl}(t, P, n) \sigma_{il}(t, P, n)$ is the covariance of $dP_k(t)$ and $dP_l(t)$. The value function should also satisfy

\[
V(t, P, n) \geq \max_{m \in \Omega(n)} \{V(t, P, m) - C_{n,m}\}
\]

or equivalently,

\[
G_n(t, P, V) \triangleq V(t, P, n) - \max_{m \in \Omega(n)} \{V(t, P, m) - C_{n,m}\} \geq 0,
\]

where $\Omega(n) \triangleq \{m|(n, m) \in L\}$ is the set of alternative activities succeeding $n$. When the activity is switched from $n$ to $m$, equality holds in condition (10) and $m$ should be the maximizer of the right hand side of (9). Since one of the two actions—to switch from $n$ to one of the alternatives $\Omega(n)$ and to defer switching—must be optimal, either Eqs. (7) or (10) holds as an equality. Hence, the optimality condition for switching activity from $n$ at the state $(t, P)$ is

\[
\begin{cases}
F_n(t, P, V) > 0 \quad \text{and} \quad G_n(t, P, V) = 0, \\
F_n(t, P, V) = 0 \quad \text{and} \quad G_n(t, P, V) > 0
\end{cases}
\]

or, equivalently,

\[
\min \{F_n(t, P, V), G_n(t, P, V)\} = 0.
\]
Since \( G_n(\cdot) \) can be rewritten as
\[
G_n(t, P, V) = \min_{m \in O(n)} F_{n,m}(t, P, V),
\]
where
\[
F_{n,m}(t, P, V) = V(t, P, n) - V(t, P, m) + C_{n,m}, \quad \forall m \in O(n),
\]
the optimality condition for activity \( n \) finally can be represented by
\[
\min\left\{ F_{n,n}(t, P, V), \quad \min_{m \in O(n)} F_{n,m}(t, P, V) \right\} = 0.
\] (15)

It should be noted that the optimality condition for the activity value \( V(t, P, n) \) of activity \( n \) involves the value of succeeding activities \( V(t, P, m) | m \in O(n) \), each of which should satisfy its own optimality condition possibly involving \( V(t, P, n) \). This implies that, for any activity \( n \), determination of \( V(t, P, n) \) may require the use of the values of all other activities \( V(t, P, m) \). Therefore, the optimality conditions of all activities must be combined and the values of all activities should be obtained as a solution of the following system of infinite dimensional GLCPs:

Find \( V = \{ V(t, P, n) \} \) such that
\[
\min\left\{ F_{n,n}(t, P, V), \quad \min_{m \in O(n)} F_{n,m}(t, P, V) \right\} = 0, \quad \forall (t, P, n) \in [0, T] \times \mathcal{R}^K \times N.
\]

The terminal condition at the maturity, \( t = T \), is given by
\[
V(T, P(T), n(T)) = \max\left\{ I(P(T), n(T)), \quad \max_{m \in O(n)} \{ V(T, P(T), m(T)) - C_{n,m} \} \right\}, \quad \forall n(T) \in N, V(P(T) \in \mathcal{R}^K.
\] (GLCP0)

For instance, in the option network of Fig. 8, the optimality condition (GLCP0) held at \( (t, P) \) reads:
\[
\begin{align*}
\min\{ -L_1 V_1(\cdot) - \pi_1(\cdot), \quad V_1(\cdot) - V_2(\cdot) + C_{1,2}, \quad V_1(\cdot) - V_3(\cdot) + C_{1,3} \} &= 0, \\
\min\{ -L_2 V_2(\cdot) - \pi_2(\cdot), \quad V_2(\cdot) - V_3(\cdot) + C_{2,3}, \quad V_2(\cdot) - V_4(\cdot) + C_{2,4} \} &= 0, \\
\min\{ -L_3 V_3(\cdot) - \pi_3(\cdot), \quad V_3(\cdot) - V_2(\cdot) + C_{3,2}, \quad V_3(\cdot) - V_4(\cdot) + C_{3,4}, \quad V_3(\cdot) - V_5(\cdot) + C_{3,5} \} &= 0, \\
- \pi_4(\cdot) &= 0, \\
- \pi_5(\cdot) &= 0,
\end{align*}
\]

where \( V_n(\cdot) \) and \( \pi_n(\cdot) \) stand for \( V(t, P, n) \) and \( \pi(t, P, n) \), respectively.

4. Decomposition into finite dimensional GLCPs

Since (GLCP0) does not have closed-form solutions, the solution of the option network problem (P) should be obtained numerically. This section and the subsequent two sections show an efficient numerical method to obtain approximate solutions of the option network problems. In this section we first reformulate (GLCP0) in an appropriate discrete time-state framework, which can be decomposed into a series of subproblems, each of which is formulated as a finite dimensional GLCP by using the DP principle with respect to time; Section 5 shows that each subproblem can also be decomposed further into smaller GLCPs by applying the DP principle with respect to the graph structure. These two decompositions together with recent advances in the theory of complementarity problems as in Nagae and Akamatsu (2008) enable us in Section 6 to develop an efficient algorithm for solving the option network problem (P). In what follows, for notational simplicity we only treat the case of a single state variable (i.e. \( K = 1 \)).

Suppose there is a sufficiently large subspace \([P_{\min}, P_{\max}]\) in the state space \( \mathcal{R} \), and a discrete grid in the time-state space \([0, T] \times [P_{\min}, P_{\max}]\) with increments \( \Delta t \) and \( \Delta P \). Let \((t', P') = (i \Delta t, j \Delta P + P_{\min})\) be points of the grid, where indices \( i = 0, 1, \ldots, I \) and \( j = 0, 1, \ldots, J \) describe the locations of the points with respect to time and space, respectively; \( I \) and \( J \) are set so that \( t = T \) and \( P = P_{\max} \). In this framework, we denote an arbitrary function \( X : [0, T] \times \mathcal{R} \times N \rightarrow \mathcal{R} \) at a grid point \((t', P')\) for activity \( n \) by \( X_n(t', P', n) \), and let \( \mathbf{V}_n \equiv (X_n^1, \ldots, X_n^n) \in \mathcal{R}^J \) be the value of activity \( n \) at time \( t' \). We also use the following \( N \)-tuple vectors \( \mathbf{V} \equiv (\mathbf{V}_1, \ldots, \mathbf{V}_N) \) and \( \mathbf{\pi} \equiv (\mathbf{\pi}_1, \ldots, \mathbf{\pi}_N) \).
In this discretized framework, the partial differential operator $L_n$ defined in (8) can be approximated by using an appropriate finite-difference scheme as follows:

$$L_n V(t^n, P, n) \approx L_n^i V_n^i + M_n^i V_{n+1}^i, \quad \forall n \in N, \quad \forall i = 0, 1, \ldots, I-1,$$

where $L_n^i$ and $M_n^i$ are $J \times J$ matrices determined by the state variable dynamics (1). See, for example Jailet et al. (1990), for more detailed derivations of $L_n^i$ and $M_n^i$. By using this, we can approximate $F_{n,i}(t^n, P, V)$ and $F_{n,m}(t^n, P, V)$ at $t^n$ in (7) and (17) by

$$F_{n,i}^i(V^i, V^{i+1}) \approx -L_n^i V_n^i - M_n^i V_{n+1}^i - \pi_n^i,$$

(19)

$$F_{n,m}^i(V^i) \approx V_n^i - V_{n+1}^i + I_{j} C_{n,m}, \quad \forall m \in O(n)$$

(20)

for any $i=0, 1, \ldots, I-1$ and $n \in N$. In Eq. (20), $I_j$ is a $J$-tuple column vector, each of whose elements is 1.

Accordingly, the optimality condition (GLCP0) can be discretized as a series of finite dimensional GLCPs:

Find $\{V^i \in \mathbb{R}^J \mid i = 0, 1, \ldots, I-1\}$ such that

$$\min \left\{ F_{n,i}(V^i, V^{i+1}), \min_{m \in O(n)} F_{n,m}^i(V^i) \right\} = 0, \quad \forall n \in N, \forall i = 0, 1, \ldots, I-1$$

(GLCP1)

where $\min(F_1, F_2, \ldots)$ is a vector operator, whose $j$th element is defined as $\min(F_j, F_2, \ldots)$; $0_j$ is a $J$-tuple column vector with all elements equal to 0. The terminal condition (16) is also rewritten as

$$V_n^i = \max \left\{ I_n^i, \max_{m \in O(n)} (V_m^i - I_{j} C_{n,m}) \right\} \quad \forall n \in N,$$

(21)

where $I_n^i = (I_{n,1}, \ldots, I_{n,J})$ is a $J$-tuple column vector with components $I_{n,j}^i = I_j(n)$. The $i$th subproblem of (GLCP1) is

Find $V^i \in \mathbb{R}^J$ such that $H^i(V^i, V^{i+1}) = 0$. (GLCP^i)

where $H^i : \mathbb{R}^J \rightarrow \mathbb{R}$ is a vector function defined as

$$H^i(\cdot) \triangleq \begin{bmatrix} H_{1}^i(\cdot) \\ \vdots \\ H_{n}^i(\cdot) \end{bmatrix}, \quad H_n^i(\cdot) \triangleq \min \left\{ F_{n,i}(\cdot), \min_{m \in O(n)} F_{n,m}^i(\cdot) \right\}, \quad \forall n \in N.$$

(22)

$F_{n,i}$ and $F_{n,m}$ are defined by Eqs. (19) and (20). Note that the $i$th subproblem (GLCP^i) can be solved only if the solution of the one-step-ahead subproblem, $V^{i+1}$, is given.

5. Decomposition with respect to graph structure

The previous section shows that the optimality condition approximated in a finite time-state framework can be decomposed into a series of subproblems, each of which is formulated as a finite dimensional GLCP. This section shows that the subproblem (GLCP^i) can further be decomposed into much smaller GLCPs by using the DP principle with respect to the graph structure. In what follows, we first discuss the decomposition of the option networks without cyclic (i.e. partially/costly reversible) structures. We then show a similar decomposition which is also applicable to option networks with cyclic structures.

It is noteworthy that one might be able to solve the subproblem (GLCP^i) as a whole by using our solution algorithm shown in Section 6. However, the graph structure decomposition greatly reduces the computational burden especially in the case with tens of activities.

5.1. The case without cyclic structures

Suppose an option network without cyclic structure as illustrated in Fig. 9, whose $i$th subproblem (GLCP^i) is to be solved given $V^{i+1}$. It is clear that the values of the ‘terminal’ activities, each of which has no succeeding alternatives (i.e. activities 4 and 5 in the example in Fig. 9), can be obtained independently of the values of other activities. Let $N_0^i \triangleq \{ n \mid O(n) = \emptyset \}$ be the set of terminal

Fig. 9. An option network without cyclic structures.
activities. Then the value \( V_i^n \) of each terminal activity can be obtained as a solution of the following linear equation system.

\[
F_{n,i}(V_i^n, V_{i+1}^n) = -L_i^n V_i^n - M_i^n V_{i+1}^n - \pi_i^n = 0, \quad \forall n \in N_0^E.
\]  

(23)

Given the values of the terminal activities \( V_i^n = (V_i^n | n \in N_0^E) \), we are now able to calculate further activity values: There is at least one activity \( n \) succeeded only by terminal activities, i.e. \( O(n) \subseteq N_0^E \). Its value, \( V_i^n \), can be obtained by solving the following finite dimensional GLCP:

Find \( V_i^n \in R^j \) such that

\[
\min \left\{ -L_i^n V_i^n - M_i^n V_{i+1}^n - \pi_i^n, \quad \min_{m \in O(n)} \{ V_m^n - V_{m+1}^n + I_{N,m} \} \right\} = 0_j.
\]

(GLCP\(^n\))

We use the notations \( V_i^n \) to emphasize that these variables are given constants. For the option network depicted in Fig. 9, the value of activity 3 can be obtained by solving (GLCP\(^3\)) for \( n = 3 \) given the terminal activity values \( V_4, V_5 \). The solution method for the subproblems (GLCP\(^n\)) will be shown in Section 6.

Let \( N_0^F \triangleq \{ n | O(n) \subseteq N_0^E \} \) be the set of ‘value determined’ activities, i.e. activities whose succeeding activity set is in \( N_0^E \) and whose values are determined by solving (GLCP\(^n\)) given \( V_i^n \). When we obtain the values \( V_i^n = (V_i^n | n \in N_0^E) \) of these activities, we are then able to obtain further activity values: There is, again, at least one activity \( n \) that is succeeded by the value-determined activities, i.e. \( O(n) \subseteq N_0^E \). Its value \( V_i^n \) can be obtained by solving (GLCP\(^n\)). In the case of Fig. 9, we can obtain the value of activity 2 by solving (GLCP\(^2\)) for \( n = 2 \) given \( V_3, V_4, V_5 \).

Unless the option network is disconnected, when we have the set \( N \) of activities, each value of which has been evaluated, there is at least one activity \( n \) succeeded only by the value determined activities and with unknown value \( V_i^n \), that is, \( O(n) \subseteq N \). Thus we can evaluate \( V_i^n \) by solving the corresponding (GLCP\(^n\)) and we add \( n \) to the value-determined activity set \( \hat{N} \). Repeating this procedure, we eventually obtain all activity values, \( V_i^n \).

The algorithm for solving the \( i \)th subproblem of the option network problem (GLCP1) without cyclic structures can be summarized as follows.

**Algorithm 1.** Solving the \( i \)th subproblem (GLCP\(^i\)) for the cases without cyclic structures

For any \( n \in N \) let \( A(n) \rightleftharpoons |O(n)| \) be the number of succeeding activities.

Obtain the terminal activity values \( (V_i^n | n \in N^F) \) by solving the linear equation system (23);

Let \( \hat{N} \rightleftharpoons N^F; \\
\text{while } \hat{N} \neq 0 \text{ do} \\
\quad \text{let } n \text{ be the first element of } \hat{N} \text{ and } \hat{N} \leftarrow \hat{N} \setminus n; \\
\quad \text{for all } m \text{ succeeded by } n \text{ (i.e. } m \in \{ k | (k,n) \in E \} \text{) do} \\
\qquad A(m) \leftarrow A(m) - 1; \\
\qquad \text{if } A(m) = 0 \text{ then} \\
\qquad \quad \text{Obtain } V_m^n \text{ by solving (GLCP\(^m\))}; \\
\qquad \quad \text{Insert } m \text{ at the end of } \hat{N}; \\
\qquad \text{end if} \\
\quad \text{end for} \\
\text{end while}

5.2. The case with cyclic structures

The decomposition with respect to graph structure is also applicable to option network problems with cyclic structures. Each link embedded in a cyclic structure means that the corresponding switch is (costly) reversible and thus can be repeated an arbitrary number of times.

Fig. 10 depicts an option network with two distinct cycles, one of which consists of links (1, 2) and (2, 1) and the other involves three links: (3, 5), (4, 3) and (5, 4). This option network allows the decision maker to switch back and forth between activities 1 and 2 as well as to repeat an activity switching cycle \( 3 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow \cdots \) any number of times. The switch between activities 1 and 2 (henceforth represented as \( 1 \leftrightarrow 2 \)) is no longer repeatable after the three-activity cycle (henceforth denoted as \( 3 \leftrightarrow 5 \leftrightarrow 4 \)) is started. In other words, the switch from the repeatable switch \( 1 \leftrightarrow 2 \) to the cyclic switch \( 3 \leftrightarrow 5 \leftrightarrow 4 \) is irreversible. This implies that the values of activities 3, 4, 5 can be determined prior to those of 1, 2. The option

![Fig. 10. An option network with cyclic structures.](image-url)
network problem with cyclic structures is also decomposable with respect to graph structures by taking the resolution with respect to each cycle rather than to each single node.

To discuss this in detail, we first define the cyclic substructures. For an option network \( \mathcal{G}(N,L) \), a subgraph, \( c \triangleq \mathcal{G}(N_c,L_c) \), is referred to as a cyclic substructure, if it consists of a set of links and nodes forming a cycle in which each activity \( m \in N_c \) is reachable from any other activity \( n \in N_c \) by links in \( L_c \). i.e. for any combination of nodes \( (n,m) \in N_c \times N_c \), there exists at least one path from \( n \) to \( m \) in \( Z^+ \). Let \( C = C(N,L) \) denote the set of cyclic subgraphs of the option network \( \mathcal{G}(N,L) \), and \( N^f = \{ n \mid n \notin N \cap N^c, \forall c \in C \} \) be the set of independent nodes that are not in any cyclic substructures. In the example of an option network depicted in Fig. 11, there are two cyclic subgraphs \( C \triangleq \{ c_1, c_2 \} \), where

\[
N_{c_1} \triangleq \{ 6,7 \}, \quad L_{c_1} \triangleq \{ (6,7),(7,6) \},
\]

\[
N_{c_2} \triangleq \{ 3,4,5 \}, \quad L_{c_2} \triangleq \{ (3,4),(3,5),(4,5),(5,3) \}.
\]

For any \( n \in N_c \) in a cyclic subgraph \( c \), it is clear from the definition that no succeeding activity \( m \in O(n) \cap N_c \) is reachable to any node in \( N_c \). The subgraph \( c_1 \) with node set \( N_{c_1} \triangleq \{ 3,5 \} \) and link set \( L_{c_1} \triangleq \{ (3,5),(5,3) \} \) is not a cyclic subgraph, since activity \( 4 \notin N_{c_1} \) that succeeds \( 3 \in N_{c_1} \) is reachable to activity \( 5 \in N_{c_1} \). When an activity \( m \in (\bigcup_{n \in N_c} O(n)) \cap N_c \) is unreachable to \( N_c \), we say that the cyclic subgraph \( c \) is succeeded by \( m \). Nodes or cyclic subgraphs might be succeeded not only by nodes but also by cyclic subgraphs. In Fig. 11, activity 1 is succeeded by both activity 2 and cyclic subgraph \( c_2 \), activity 2 is succeeded by two cyclic subgraphs \( c_1 \) and \( c_2 \), and cyclic subgraph \( c_2 \) is succeeded by cyclic subgraph \( c_1 \) and activity 8, respectively. Cyclic subgraph \( c_1 \) and activity 8 have neither succeeding activities nor cyclic structures. For a cyclic subgraph \( c \), we henceforth use \( O(c) \) to represent the set of its succeeding activities and subgraphs. For the option network depicted in Fig. 11, \( O(1) = \{ 2, c_2 \}, O(2) = \{ c_1, c_2 \}, O(1) = \{ c_1, 8 \}, O(c_1) = O(8) = \emptyset \). Let \( C^F \triangleq \{ c \in C \mid O(c) = \emptyset \} \) be the set of terminal cyclic substructures.

Using the notion of succeeding activities and cyclic subgraphs, we can apply the decomposition with respect to graph structure to an option network problem with cyclic structures. Suppose an option network represented in Fig. 11, whose \( i \)th subproblem (GLCP\(^i\)) is to be solved given \( V^{i+1} \). First, it is clear that the value of activity 8 can be calculated independently of other activity values (as a solution of the corresponding linear equation system (23)), since there is no succeeding activity or cyclic subgraph for this activity. Since cyclic subgraph \( c_1 \) also has no succeeding activities/cyclic subgraphs the values of activities in cyclic subgraph \( c_1 \), which we denote by \( V_{c_1}^i \triangleq \{ V_3^i, V_4^i, V_5^i \} \), can be obtained independently. The problem of evaluating \( V_{c_1}^i \) will be discussed below. Having \( V_{c_1}^i \) and \( V_{c_2}^i \), we can obtain the values of activities in cyclic subgraph \( c_2 \), \( V_{c_1}^i \triangleq \{ V_1^i, V_2^i, V_3^i \} \), as the cyclic subgraph is succeeded by cyclic subgraph \( c_1 \) and activity 8 and their values are known. We can then obtain \( V_{c_2}^i \), given the values of the succeeding cyclic subgraphs, \( V_{c_1}^i \) and \( V_{c_2}^i \). Finally we can obtain \( V_{c_2}^i \) by solving (GLCP\(^{m}c_1\)) given \( V_{c_2}^i \) and \( V_{c_2}^i \).

The problem that should be solved to obtain the values \( V_{c_1}^i \) or \( V_{c_2}^i \) of a cyclic subgraph is the following GLCP:

\[
\text{Find } V_{c}^i \triangleq \{ V_n^i \mid n \in N_c \} \in \mathbb{R}^{N_c} \text{ such that } H_c'(V_c^i; V^{i+1}) = \theta_{N_c}, \quad (\text{GLCP}^{i}_{c})
\]

where \( N_c \triangleq |N_c| \); \( H'_c(\cdot) \triangleq (H'_c(\cdot)) \mid n \in N_c \) and \( H'_n(\cdot) \) is defined by (22). It should be noted that the values of activities in a cyclic substructure should be obtained simultaneously: for any two activities \( n, m \) in a cyclic substructure, \( V_n^i \) cannot be determined without \( V_m^i \) and vice versa. A solution method for (GLCP\(^i\)) will be given in Section 6.

The above procedure for solving (GLCP\(^i\)) can be reduced to the solution procedure for the \( i \)th subproblem of the option network without cycle depicted in Fig. 9, when we replace each of the cyclic subgraphs by an artificial node as shown in Fig. 12, where the cyclic subgraphs \( c_1 \) and \( c_2 \) are replaced by the corresponding artificial nodes.

---

Fig. 11. An example of cyclic subgraphs.

Fig. 12. Reduction to an option network without cyclic structures.
Accordingly, the $i$th subproblem for an option network with cyclic structure can be solved by the following procedure: Take an activity $n$ or a cyclic subgraph $c$ succeeded by only the 'value-determined' activities; solve either a GLCP (GLCP$_n$) or a system of GLCPs (GLCP$_c$), and add $n$ or all activities in $c$ to the set of value-determined activities. This is equivalent to Algorithm 1, but each cyclic substructure is treated as an independent node, which can be summarized as follows.

**Algorithm 2.** Solving the $i$th subproblem (GLCP$_n$) for the cases with cyclic structures

For any node or cyclic substructure $n \in N \cup C$, let $A(n) = \emptyset(n)$ be the number of succeeding activities/cyclic substructures.

Obtain the terminal activity values ($V_i|n \in N^k$) by solving the linear equation system (23);

Obtain the terminal cyclic substructure values ($V_c|c \in C^k$) by solving (GLCP$_c$);

Let $N \approx N^k \cup C^k$;

while $\hat{N} \neq 0$
do

Let $n$ be the first element of $\hat{N}$ and $\hat{N} = \hat{N} \setminus n$.

for all $m$ succeeded by $n$ (i.e. $m \in \emptyset(n)$) do

$A(m) := A(m) - 1$;

if $A(m) = 0$ then

if $m$ is a node $n$ then

Obtain $V_i$ by solving (GLCP$_n$);

else if $m$ is a cyclic substructure $c$ then

Obtain $V_c$ by solving (GLCP$_c$);

end if

Insert $m$ at the end of $\hat{N}$;

end if

end for

end while

6. Algorithm for solving the subproblems

This section sketches our algorithm for solving the subproblems (GLCP$_n$) and (GLCP$_c$). It is a natural expansion of the algorithm of Nagae and Akamatsu (2008). In the present algorithm, the finite dimensional GLCPs of the previous section, (GLCP$_n$) and (GLCP$_c$), are each rewritten as an equivalent piecewise linear equation system:

$$\text{Find } V \in \mathcal{R}^l \text{ such that } H(V) = 0,$$

where $H(V) := \min_s \{F_1(V), F_2(V), \ldots, F_S(V)\}$

$$\text{where } F_s : \mathcal{R}^l \rightarrow \mathcal{R}, \forall s \in S \text{ is a known map with } j\text{th elements } F_s(V), \text{ and } S \triangleq \{1,2,\ldots,S\}.$$

The present algorithm is based on the smoothing function approach developed by Peng (1999), Qi and Liao (1999), and Peng and Lin (1999). This approach is not only a state-of-the-art technique, but is also well-suited to our problems from the viewpoint of efficiency, as we discuss later. The key idea of the smoothing approach is to transform the original non-smooth equation system (24) into a system of smooth equations via a so-called smoothing function $G(V, \xi)$ with $j$th component

$$G(V, \xi) := -\xi \ln \sum_{s \in S} \exp \left(-\frac{F_s(V)}{\xi}\right),$$

where $\xi \geq 0$ is referred to as the smoothing parameter. For the smoothing function (26), the following two desirable properties are known: (i) $G(V, +0) := \lim_{\xi \rightarrow +0} G(V, \xi) = H(V)$; and (ii) $G(V, \xi)$ is a continuously differentiable function of $V$ for all $\xi > 0$.

This algorithm based on a smoothing approach generates a solution set to the smooth equations system forming a path $\{(V^{(k)}, \xi^{(k)})\}$, where $\xi^{(k)}$ is the smoothing parameter in the $k$th iteration, and $V^{(k)}$ is a solution of the corresponding smooth equation

$$G(V, \xi^{(k)}) = 0.$$  

Based on the first property of the smoothing function, the sequence $\{(V^{(k)}, \xi^{(k)})\}$, which is referred to as the smoothing path, is guaranteed to converge globally to the solution of (24) under reasonably mild conditions that are naturally satisfied in our framework. Also the smoothing equation (27) can be solved efficiently by any Newton-type method since $G(V, \xi)$ is continuously differentiable with respect to $V$ (the second property of the smooth function). For details of the global convergence and guaranteed efficiency of this algorithm, readers are referred to Peng and Lin (1999) and Nagae and Akamatsu (2008).

It is noteworthy that the calculation of the Newton direction for the smoothing equation (27) in the present option network framework is efficient as well since the Jacobian of the smoothing function, $\nabla G(\cdot)$, is sparse (Nagae and Akamatsu, 2008).
This is clear from the fact that $\nabla G(\cdot)$ can be rewritten as a linear combination of tridiagonal matrices, $(\nabla F_j)$:

$$\nabla G(V, \xi) \triangleq \sum_{s \in \mathcal{S}} \lambda_s (V, \xi) \nabla F_s(V),$$

(28)

where $\lambda_s : \mathbb{R}^l \times \mathbb{R}^+ \rightarrow \mathbb{R}^{l \times l}$ is a diagonal matrix whose $(j, j)$ element is

$$\lambda^{jj}_s(V, \xi) \triangleq \frac{\exp[-F_s(V)/\xi]}{\sum_{s' \in \mathcal{S}} \exp[-F_{s'}(V)/\xi]} \quad \forall s \in \mathcal{S}.$$  

(29)

Accordingly, each of the subproblems (GLCP$_s^*$) and (GLCP$_c^*$) can be solved efficiently using our algorithm.

7. Numerical examples

In this section we present an example of an option network to illustrate our numerical solution method.

Suppose a firm is considering an investment in a multinational production project which is represented as the option network depicted in Fig. 13. We assume that the product is sold in a foreign market, and thus the cash flow streams of the project change over time due to the stochastic fluctuation of the exchange rate and the choices of the production location and size.

In this option network, the five activities represent the following five operation modes respectively: idle, export, offshore, large-scale export, and large-scale offshore. The project begins with the idle mode (activity 1), at which neither profit nor loss is made. The firm can switch from the idle mode to either the export mode (activity 2) or the offshore mode (activity 3), both of which are identical in the production size, $D_1$, but are different in the cost structure: In the export mode, the production cost per unit and the export cost, denoted by $c_2$ and $e_2$, are priced in the home country currency (yen for example). In the offshore mode, the production cost per unit $c_3$ is priced in the foreign currency (dollars for example). It is also assumed that the unit product is sold at unit foreign currency. The instantaneous profits for activities 1, 2 and 3 at $t$ are thus defined as

$$\pi(t, P(t), 1) = 0, \quad \pi(t, P(t), 2) = \{P(t) - c_2\}D_1 - e_2, \quad \pi(t, P(t), 3) = \{P(t) - P(t)c_3\}D_1,$$

(30)

where $P(t)$ is the exchange rate for the foreign currency (yen per dollar) at $t$. It is assumed that the firm can switch between the export mode and the offshore mode any number of times. In other words, the firm has managerial flexibility in the production location choice: the firm would choose the export mode in the case of low $P(t)$, whereas the offshore mode would likely be chosen when $P(t)$ is high. We further assume that the firm can also expand the production size by an additional investment: the firm can either switch from the export mode to the large-scale export mode (activity 4) or switch from the offshore mode to the large-scale offshore mode (activity 5). The additional investment increases the production size from $D_1$ to $D_2$ achieving scale economy in the production and the export costs, but restricts the managerial flexibility: no future decision can be made after the expansion. Let $c_4$ and $e_4$ be the production cost per unit and the export cost for the large-scale export mode, and $c_5$ be the production cost per unit for the large-scale offshore mode. The instantaneous profits of activities 4 and 5 are then denoted by

$$\pi(t, P(t), 4) = \{P(t) - c_4\}D_2 - e_4, \quad \pi(t, P(t), 5) = \{P(t) - P(t)c_5\}D_2.$$  

(31)

Finally, we assume that the large-scale export mode is available from each mode in the first stage, whereas the large-scale offshore mode is not available from the (small-scale) export mode, reflecting so-called ‘home country advantages’ in production.

In our numerical experiments, the dynamics of the exchange rate $P(t)$ are specified by the following mean reverting process.

$$dP(t) = \mu(\overline{P} - P(t))dt + \sigma P(t)dW(t), \quad P(0) = P_0,$$

(32)

where $\overline{P}$ is the long-term average exchange rate, $\mu$ and $\sigma$ are given positive constants. The base case parameters are follow: The duration of the operation period is $T = 5$ years and the annual discount rate is $\rho = 0.02$. The long-term average exchange rate is $\overline{P} = 1.5$. The annual reverting speed and the annual volatility are $\mu = 0.05$ and $\sigma = 0.4$, respectively. The initial investments for the export mode and the offshore mode are $c_{1,2} = 0.2$ and $c_{1,3} = 0.4$, and the switching costs between these two modes are $c_{2,3} = 0.5$ and $c_{3,2} = 0.7$. The additional investment for the export mode (i.e. the switching cost from activity

![Fig. 13. The multinational production project as an option network, which consists of five activities: (1) the idle mode, (2) the export mode, (3) the offshore mode, (4) the large-scale export mode, and (5) the large-scale offshore mode.](image-url)
The value function for each activity and the optimal decision policy can be obtained as follows. Suppose that we solve the optimality condition held at \( t, (\text{GLCP}) \), with given \( V^{i+1} \). We first calculate each of the terminal activity values, \( V_4 \) and \( V_5 \), as a solution for the linear equation system (23). Given \( (V_4, V_5) \) and \( V^{i+1} \), we then solve (GLCP\(_i\)) for the cyclic subgraph 2 \( \rightarrow \) 3 and obtain \( V_2 \) and \( V_3 \). Note that \( V_2 \) and \( V_3 \) should be calculated simultaneously as the solution of (GLCP\(_i\)), which cannot be solved by using the explicit payoff method. Figs. 14 and 15 depict these values at exchange rate, the optimality condition held at \( t=3 \). We observe that the optimal decision policy is to switch to the large-scale export mode if the exchange rate exceeds the threshold \( P_{C3,4} = 2.5 \) and \( P_{C3,5} = 3.5 \), otherwise–neither the switch to 2 nor the switch to 3 is optimal for any value of \( P \). Finally, we compute the switching thresholds for the offshore mode \( (V_3−C_{3,5}) \), respectively. \( P_{C3,4} \) is the switching threshold from the export mode to the offshore mode, whereas \( P_{C3,4} \) is that to large-scale export mode.

Fig. 14. The value functions and the switching thresholds for the export mode (activity 2) at \( t=3 \). The thick solid curve is the value of the export mode, the dashed and thin solid curves are the values for switching to the offshore mode (activity 3), \( V_1 − C_{3,2} \), and the large-scale export mode (activity 4), \( V_4 − C_{2,4} \), respectively. \( P_{C3,4} \) is the switching threshold from the export mode to the offshore mode, whereas \( P_{C3,4} \) is that to large-scale export mode.

Fig. 15. The value functions and the switching thresholds for the offshore mode (activity 3) at \( t=3 \). The thick solid curve is the value of the offshore mode, the dashed, solid, and dash–dot curves are the switching values to the export mode \( (V_2−C_{3,2}) \), the large-scale export mode \( (V_4−C_{3,4}) \), and the large-scale offshore mode \( (V_5−C_{4,5}) \), respectively. \( P_{C3,4} \) is the switching threshold from the offshore mode to the large-scale export mode.
respectively. There is a single switching threshold \( P_0 \) at \( t = 3 \), which indicates that it is optimal to start the production with the export mode if \( P \geq P_0 \), and otherwise remain idle.

Since the optimal decision policy also depends on time, each of the above switching thresholds can be depicted as a function of time as shown in Figs. 17–19. In Fig. 17, \( P_{2\rightarrow3} \) and \( P_{2\rightarrow4} \) are the optimal thresholds for the export mode, whereas Fig. 18 depicts the optimal thresholds, \( P_{3\rightarrow2} \), \( P_{3\rightarrow4} \) and \( P_{3\rightarrow5} \), for the offshore mode. Fig. 19 illustrates those for the idle mode. These diagrams give complete information about the optimal decision policy. For example, Fig. 19 shows that it is optimal to begin the project with the export mode when the exchange rate \( P(t) \) hits \( P_{2\rightarrow3}(t) \), whereas the project should be started with the offshore mode when \( P(t) \) hits \( P_{3\rightarrow5}(t) \). Otherwise, it is optimal to remain idle. The optimal decision policy represented by these pictures bears no resemblance to that of the existing real option problems. This implies the existing real option rules in the literature might be too simplified and thus hardly applicable to real-world decision making problems.

Finally, we demonstrate the efficiency of our algorithm. Table 1 illustrates the total CPU time for solving the present option network problem in different problem sizes, \( I, J = 100, 200, 300, 800 \) and 1000. We use the following parameters for the solution algorithm of (GLCP): \( \omega = 0.005, \delta_1 = 0.9, \delta_2 = 0.85, \delta_3 = 0.001 \); the initial smoothing parameter is \( \xi^0 = 0.01 \); the convergence criterion is \( \|H(V)\| \leq 10^{-5} \). The solution algorithm was encoded by MATLAB and run on a 2.66 GHz quadcore personal computer. The first column of Table 1 indicates the total CPU time for solving the whole option network problem by using the present method. It shows that the present method can solve a fairly large problem with \( I=J=1000 \) in a moderate CPU time. The second column of Table 1 indicates the total CPU time when we do not decompose each subproblem (GLCP) with respect to substructures and solve it as a large GLCP. We observe that the decomposition with respect to substructure significantly decreases the computational effort.
This article provides a novel framework for analyzing option network problems, which is a general class of real option problems with a complex combination of reversible and irreversible decisions. We first introduce the option network framework that represents interdependent structure of decisions by using directed graphs (explicitly differentiating their reversibility and irreversibility). The option network problem is formulated as a singular stochastic control problem, whose optimality condition is then obtained as a dynamical system of generalized linear complementarity problems (GLCPs). We then develop an efficient numerical algorithm for solving the option network problems exploiting the two decomposable properties of the problems, each of which is revealed by our analyses: (a) in an appropriate discrete time-state framework, the solution of an option network problem can be obtained by successively solving subproblems, each of which is represented as a finite-dimensional GLCP; (b) each of the time-decomposed subproblems further reduces to a problem of solving a succession of much smaller sub-subproblems, which is analogous to the shortest-path finding.
The present modeling framework separates the decision making problem under dynamic uncertainty into two parts: the decision structure represented by a directed graph; and the quantitative properties, such as the underlying processes and the instantaneous profit/payoff functions. This implies that if the present method is implemented as a decision support system enabling us to treat the latter part as a ‘black box,’ it can be used by corporate managers, practitioners, and perhaps some academics. They have only to describe the decision structure as a directed graph, choose suitable sets of the underlying stochastic processes, the instantaneous profit, and the terminal payoff for each activity from a given catalog, and press the ‘calculation’ button. It would be easier and more intuitive than generating a vast decision tree/influence diagram with tens of nodes.

References