Computers and Chemical Engineering 121 (2019) 556–573

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Multi-timescale, multi-period decision-making model development by combining reinforcement learning and mathematical programming

Joohyun Shin, Jay H. Lee*

Chemical and Biomolecular Engineering Department, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea

ARTICLE INFO

Article history:
Received 20 August 2018
Revised 22 November 2018
Accepted 25 November 2018
Available online 28 November 2018

Keywords:
Multi-timescale decision making
Decision under uncertainty
Markov decision process
Mathematical programming
Reinforcement learning

ABSTRACT

This study focuses on the linkage between decision layers that have different time scales. The resulting expansion of the boundary of decision-making process can provide more robust and flexible management and operation strategies by resolving inconsistencies between different levels. For this, we develop a multi-timescale decision-making model that combines Markov decision process (MDP) and mathematical programming (MP) in a complementary way and introduce a computationally tractable solution algorithm based on reinforcement learning (RL) to solve the MP-embedded MDP problem. To support the integration of the decision hierarchy, a data-driven uncertainty prediction model is suggested which is valid across all time scales considered. A practical example of refinery procurement and production planning is presented to illustrate the proposed method, along with numerical results of a benchmark case study.

1. Introduction

In general, sequential decision making takes the form of a closed loop system which observes the responses of the environment resulting from the decisions taken and uncertainty realized. The typical setting is *dynamic stochastic decision-making* meaning a series of decisions are to be made in interacting with a dynamic and stochastic environment to maximize overall reward. Such sequential decision problems are found in a range of fields such as industrial and manufacturing system (e.g., process control, scheduling and planning, logistics), robotics, financial applications (e.g., optimal investment), medical application (e.g., clinical treatment), computing and communications (e.g., cloud computing, multimedia/radio network), game playing, and power system (e.g., water reservoir control, management of energy consumption).

In some applications, the decision process takes place at different levels of different timescales, which is referred to as a *multi-timescale* decision problem. Specifically, in the standard decision hierarchy adopted by the process system engineering community, the high-level strategic or planning problem (e.g., supply chain design, capacity planning, production targets, and selection of tasks) provides a set of decisions on a longer time scale, i.e., on a yearly, monthly, or weekly basis (McKAY et al., 1995). This information is then fed as input to a lower-level operation decision problem (e.g., scheduling) with the goal of obtaining a complete execution solution on a faster time scale, e.g., daily or hourly (Amaro and Barbosa-Póvoa, 2008).

Since decision layers are interrelated, one-way communication in the hierarchical structure can result in an inconsistent or infeasible operation solution. Currently, the high levels tend to be at best loosely coupled to the information flow and analysis that occur at lower levels in the hierarchy due to the difference in timescale. As a result, a plan or goal can be established based on optimistic estimations that neglect the effects of system dynamics and uncertainty realized on the faster timescale, and this leads to a gap between the plan and execution. Thus, it is recognized that tighter integration of the decision layers is an important research issue (Grossmann, 2005; Grossmann and Guillén-Gosálbez, 2010) in order to achieve a globally optimal and sustainable solution. The economic benefits of the integration of planning and scheduling have been proven by industrial companies (McDonald, 1998; Shobrys and White, 2002).

Solution strategies for dealing with the integration of planning and scheduling are broadly classified into three categories (Maravelias and Sung, 2009): hierarchical, iterative, and full-space methods. In hierarchical and iteration approaches, the master problem for determining high-level decisions employs relaxed and aggregated low-level models to find a feasible region of real execution, but the difference lies in how the disparities resulting from the use of surrogate models are handled. In the one-way hierarchical models, it relies solely on a *reactive* response through rule-based correction of feasible schedule (Bassett et al., 1996; Grunow et al., 2002; Yan et al., 2003) or rolling horizon framework

* Corresponding author.
E-mail addresses: sinnis379@kaist.ac.kr (J. Shin), jayhlee@kaist.ac.kr (J.H. Lee).

https://doi.org/10.1016/j.compchemeng.2018.11.020
0098-1354/

(Honkomp et al., 1999; Lin et al., 2002; Sand and Engell, 2004; van den Heever and Grossmann, 2003). In the iterative models, information from the scheduling sub-models is iteratively fed back to the master problem, i.e., by adding integer cuts, to close the gap between them. Many studies (Dogan and Grossmann, 2006; Papageorgiou and Pantelides, 1996; Stefansson et al., 2006; Wu and Ierapetritou, 2007) have been published in this category, but they still have a structural problem of using the surrogate model and their application is problem-specific.

On the other hand, full-space approaches consider an integrated model containing detailed scheduling sub-models for each planning period. However, the standard mathematical programming formulation for simultaneous planning and scheduling with a fine time grid yields a large-scale optimization problem that quickly becomes computationally infeasible as the decision period expands. One possible approach is to use decomposition techniques (Grossmann, 2012) (e.g., Bender's decomposition, Lagrangian relaxation) to iteratively solve the decomposed subproblems, but existing approaches still allow a narrow range of application due to restrictive formulation and computational infeasibility.

Meanwhile, there are two decision-making policies for sequential decision problems: look-ahead policy via mathematical programming (MP) and Markov decision process (MDP) based policy via dynamic programming (DP). MP-based approaches provides solution over time, and are particularly useful for addressing realistic and complex dynamics and constraints. Meanwhile, a state-oriented formulation and Markov property in MDP allow stage-wise decomposition of the entire multi-period problem, and decision policy can be obtained based on the value function capturing future reward (or cost). However, DP has its own computational issue known as the curse of dimensionality (Powell, 2007). Instead of the exact DP to solve the MDP, reinforcement learning (RL) or approximate dynamic programming (ADP) algorithms have become attractive alternatives to learn a decision policy by interacting with the environment and receiving feedback about its performance. They use stochastic simulations and function approximations to find computationally feasible solutions.

The usefulness of these MP- and MDP-based approaches for addressing multi-period decision-making problems has been demonstrated in many areas, but practical application of them is still limited in large-scale stochastic problems due to the computational complexity (Maravelias and Sung, 2009). Particularly in the multi-timescale decision problems, modeling approaches for efficient integration of the timescales and computationally tractable solution algorithms are lacking so far (Lee, 2014). Here, the limitations of the existing methods are demonstrated through a concrete motivating example.

1.1. Motivating example: Refinery procurement and production planning (Fig. 1)

Refinery is an important production facility for refining or converting petroleum-based raw materials into valuable products. Since refinery contains a complex network of processing units and produces multiple products of desired specs to make a profit, operational planning is of crucial importance. The refinery planning is generally performed in sub-systems divided by temporal or spatial scales (Bengtsson and Nonås, 2010). Crude oil procurement is one of the most important decisions, which affects the rest of the planning and other operation decisions. These days a typical refinery must consider a large number of different types of crude oils, of which price and quality vary significantly (Dunn and Holloway, 2012), and the netback value of each crude selection should be evaluated by considering the subsequent refining process.

There is a time delay between crude purchases and product sales due to the different decision scales between crude procurement (say, every H-days, which is the longer time period indexed as t) and refinery operation (every day, which is the shorter time period indexed as h), as illustrated in Fig. 1, The corresponding price difference results in a gap between the expected netback value of the crude oils at the planning level and the actual margin achieved at the execution level. Thus to reduce this gap, the uncertainty on the prices of products, which are realized after the crude purchasing, should be proactively considered. In addition, to ensure optimal inputs into the refinery at all times under constantly changing prices of crudes and products, one needs to manage the inventories of crude oils.

1.1.1. MP over timescale multiplicity

The standard MP formulation for simultaneous crude procurement and refinery operation over the fine time grids (i.e., of one day) yields a computationally infeasible large-scale optimization problem. In particular, when the problem is expressed as a multi-stage stochastic programming (MSSP) based on a scenario tree, the number of branches increases exponentially, according to $O(M^{(H\times T)})$ where M is the number of possible outcomes per day and T is the decision horizon of the crude procurement planning. The problem will quickly become intractable as the number of the higher-level decision (crude import) times, the number of the lower-level decision (refinery operation) times within each high-level time period, and the number of uncertainty (price) scenarios grow.

1.1.2. DP (or RL) with complex processing model

In order to formulate an MDP representing a multi-period decision problem involving complex dynamic and operational constraints, the size of state and decision space becomes huge. In the illustrative example, to formulate an MDP for the refinery operation problem, all the flowrates of processing and blending materials (represented in Fig. 6) must be decision variables, but it is not easy to define a feasible decision space in a given state (e.g., inventory, price) due to the complex interconnected constraints such as predetermined crude purchases, material balances for all intermediate and final products according to the process network, quality satisfaction of the final products, and capacity restriction of each processing unit. In addition, since most state and decision variables (e.g., inventory, flowrate) are continuous, the size of space grows quickly (which is generally called *curse-of-dimensionality*), and the aggregation of those variables can lead to a suboptimal or infeasible solution due to the tight constraints.

Meanwhile, there is a view that the MP- and MDP-based approaches are complementary as having different pros and cons, rather than being competitive (Barro and Canestrelli, 2016; Dupačová and Sladký, 2002). Each one is specialized in efficiently handling a particular type of problems. It is noteworthy that the favorable and unfavorable features of each approach are related to the characteristics of the high- and low-levels in the decision hierarchy. The high-level decisions are made on a relatively longer time span considering exogenous uncertainty (e.g., market condition) and relatively simple system states (e.g., inventories). On the other hand, the low-level decisions should be made within a finite horizon subject to the higher-level decisions and other complex constraints and state representation (e.g., product spec, complex process network).

Based on this observation, we develop a multi-timescale decision-making model that combines MDP and MP in a complementary manner and introduce a computationally tractable solution algorithm based on RL to solve it. Specifically, the MDP is formulated for the high-level planning of which one (planning) period model is determined by the *optimization-embedded simulation* of the low-level model, rather than a coarse-grained surrogate model. The low-level operation model, on the other hand, is constructed

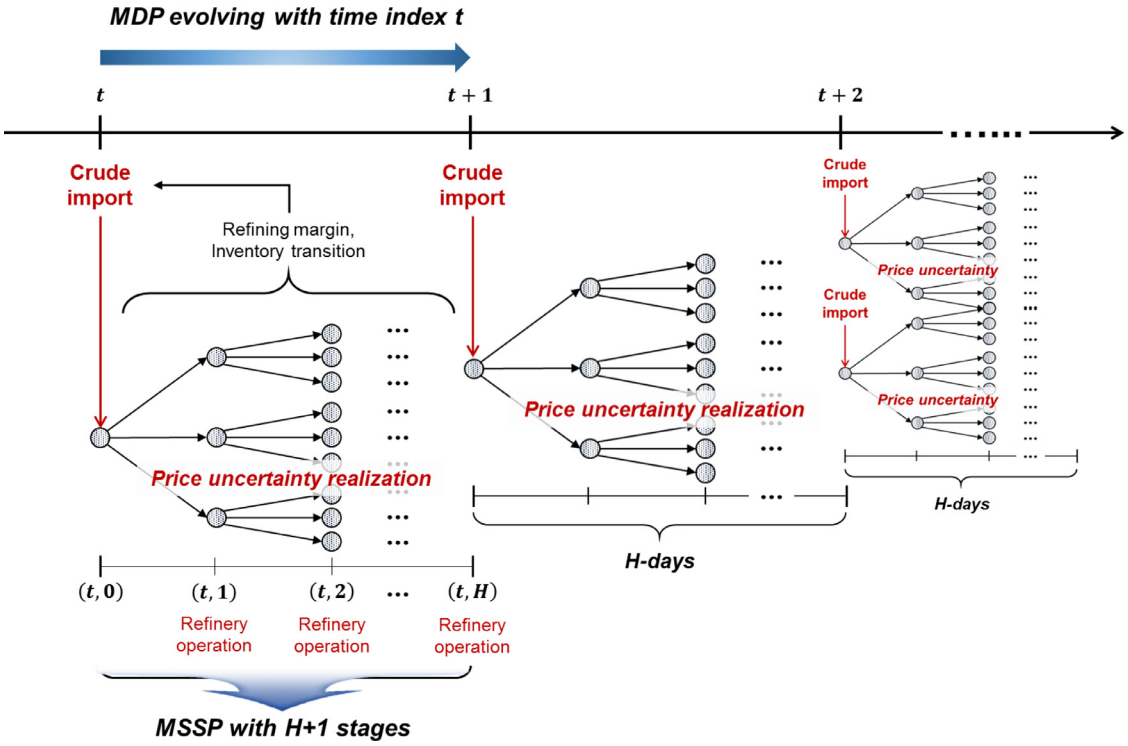

Fig. 1. The motivating example of multi-timescale dynamic stochastic decision problem: refinery procurement and production planning considering crude inventory dynamics and price uncertainty.

as a MP (especially MSSP) containing the value of being in a system state at the end-of-horizon (*value function* evaluated in the upper-layer MDP) to ensure longer-term implications of the decisions within the considered horizon. In addition, to support the integration of decision hierarchy, development of a data based uncertainty prediction model is suggested, which is consistently valid across all the considered time scales.

The main contribution of this work is to propose a modeling framework for sequential decision-making under uncertainty with the following features: 1) efficient integration of multi-timescale decision hierarchy by complementary use of MP and RL, 2) proactive capture of uncertainty by developing a multi-timescale stochastic model, and 3) reduction of end-effects, which is caused by the use of finite horizon in optimization, through end-state value evaluation. The strength of this model is its versatility making it useful in a variety of areas where the structures of decision-making and environment are similar to the refinery procurement and production planning problem, which is used to demonstrate the proposed method.

The rest of the paper is organized as follows. Section 2 provides a review of methodologies for sequential decision-making under uncertainty. The model formulation and solution of the suggested multi-timescale decision-making model is introduced in Section 3. Finally, a practical example of refinery management is described in Section 4 to demonstrate the proposed method. Section 5 concludes the paper.

2. Methodology

Powell claims that there appear to be four fundamental classes of decision policy (Powell, 2014): policy function approximations (PFAs), optimizing a cost function approximation (CFA), policies that depend on a value function approximation (VFA), and look-ahead policies. According to the tutorial article, PFAs work best for low-dimensional actions, where the structure of the policy is fairly obvious, whereas CFAs work best for cases when a deterministic model works well or when the impact of uncertainty is easy to recognize. In other cases of large-sized or time-dependent problems, VFA-based approaches and the various forms of look-ahead policies have received considerable attention (Powell, 2014).

Look-ahead policies are often expressed under names such as model predictive control (MPC) and rolling horizon procedures for deterministic models, or robust optimization and multi-stage stochastic programming (MSSP) for stochastic models (Powell and Meisel, 2016). In MSSP, stochastic behavior is expressed as a scenario tree where the realization of each uncertainty corresponds to one node in the stage and each node has a unique ancestor (Fig. 2, left), and the resulting solution is a decision tree. The value function naturally appears in solving an MDP as the expected sum of all future consequences of the decision to solve the entire multi-period problem in a period-wise recursive manner (Fig. 2, right). Instead of a look-up table form of the value function, some sort of function approximation methods (e.g., user-specified parametric model, neural network, nearest neighbor averager, Taylor series) can be used to lower the computation.

2.1. Look-ahead policy via mathematical programming (MP)

Look-ahead policies are particularly useful for time-dependent decision problems, especially when forecasts (scenarios) of uncertainty over the time spans are available in advance. One can typically obtain a look-ahead policy using a mathematical programming (LP, MILP, MINLP etc.) algorithm designed to handle constraints. The generality of such formulations allowed for consideration of the specific condition and circumstance of each case (Chen et al., 2013; Hawkes and Leach, 2009; Ren and Gao, 2010).

Particularly in process operation and management, it is important to forecast uncertainties due to external market/supply changes or internal system variations and to make robust decisions about these uncertainties. The technique of MPC has been widely

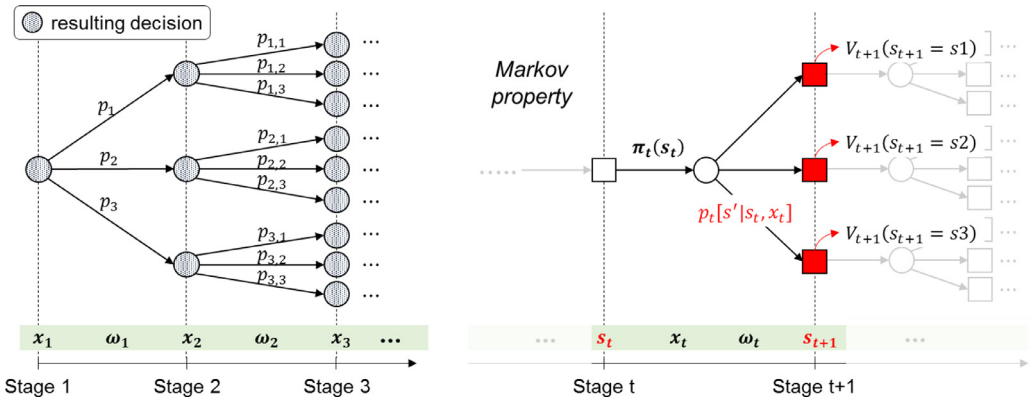

Fig. 2. Scenario tree based MSSP structure (left), and value function based stage-wise structure of state-driven MDP model (right).

applied in the problems of supply chain management, planning and scheduling of manufacturing system (Bose and Pekny, 2000; Braun et al., 2003; Mestan et al., 2006; Perea-Lopez et al., 2003) with efficient consideration of system dynamics within some time horizon. The look-ahead decision policy optimizes the overall costs within a chosen prediction horizon in deciding all decisions in the horizon but typically only the decisions for the current time instance are implemented and the whole plan is revised through a new optimization once feedback containing additional disturbance information is received.

Employing a stochastic formulation instead of deterministic one can provide advantages as it can explicitly incorporate uncertainty information into the decision making. One of the most general modeling approaches widely used for stochastic look-ahead policies is MSSP, in which recourse decisions are made after the uncertainty is realized over a pre-specified scenarios with discrete probabilities of occurrence (Birge and Louveaux, 2011).

Let the sequence of decisions $x=(x_1, \ldots, x_T)$ and observations of stochastic data $\xi=(\xi_1, \ldots, \xi_{T-1})$ for stage $t=1,\ldots, T$ is given like $(x_1, \xi_1, \ldots, \xi_{T-1}, x_T)$. With the practical requirement that decisions taken at any stage of the process do not depend on future realizations of the stochastic process or on future decisions (which is called *nonanticipativity*), the decision at stage t is limited by explicit constraints that may depend on the previous decisions $x^t=(x_1, \ldots, x_t)$ and on past observations of $\xi^t=(\xi_1, \ldots, \xi_t)$. The overall T-stage stochastic program to find a decision over the all times is thus written as Eq. (1), where q_t is a reward function of stage t:

$$\begin{aligned}\max_x q(x) = {} & q_1(x_1) + E_{\xi_1}[q_2(x_2,\xi^1) \\ & + E_{\xi_2|\xi^1}[\ldots E_{\xi_{T-1}|\xi^{T-2}}[q_T(x_T,\xi^{T-1})]]] \\ & \text{s.t.} \quad f_{1i}(x_1) \le 0, i=1,\ldots,m_1 \\ & f_{ti}(x^t,\xi^{t-1}) \le 0, i=1,\ldots,m_t, t=2,\ldots,T. \end{aligned} \tag{1}$$

For purpose of applications, one approximates the true probability distribution of ξ by a discrete probability distribution concentrated on a finite number of scenarios (Birge and Louveaux, 2011), noted as *scenario tree* (Fig. 2, left). Consequently, by calculating the expectations of the recourse functions by generating a numbers of scenario tree (e.g., by Monte Carlo simulation) and taking sample averages, the stochastic problem Eq. (1) can be reformulated as a large-scale deterministic equivalent form.

MP-based approaches are efficient and flexible for modeling time-dependent decision problems, esp. those involving various constraints. However, a MILP or MINLP model over a large number of stages while accounting for multiple scenarios can often become extremely large and computationally challenging due to the exponential nature of branching (Pratikakis, 2009). Since the MSSP formulation is limited to the problem of only a modest number of scenarios and stages, it is often infeasible to apply to multi-period decision problems of practical significance (Lee, 2014; Sahinidis, 2004). Furthermore, the use of a finite horizon in multi-period decision making can be limiting, since what is an optimal solution within a short horizon may be highly suboptimal over the long run beyond the considered time horizon, which is known as *end-effects* (Fisher et al., 2001).

2.2. Markov decision process (MDP) based policy via dynamic programming (DP) or reinforcement learning (RL)

Markov decision process (MDP) represents another general modeling approach for addressing the problem of multi-period decision making under uncertainty. MDP is naturally solved by dynamic programming (DP). A formal Markov decision process formulation requires the following specifications with time index t (Puterman, 2014): 1) state variable, s_t to compute all future dynamics of the system, 2) decision variable, a_t, 3) exogenous information variable, ω_t, which is expressed in the form of random variables governed by probability distributions, 4) stage-wise reward (or cost) function, $R_t(s_t, a_t)$, and 5) state transition probability $P_t(s_{t+1} \mid s_t, a_t)$ from state s_t and decision a_t to the next state, s_{t+1}.

The objective is to find the best policy π, which is a map indicating which action to take for any given state, that maximizes the sum of all (discounted) future reward. Since a decision policy is constructed by estimating the long-term consequences of action, it can be applied to control problems involving temporally extended behavior. The infinite horizon problem, of which the objective function is expressed as Eq. (2) with discount factor $0<\gamma\le 1$, is thought of as a steady-state (or stationary) problem where the system dynamics do not vary over time (Powell, 2007). The MDP formulated problem is solved by dynamic programming (DP), in which the value function evaluating the expected sum of total future costs from given state s_t is introduced in the following temporally recursive form (Bellman optimality equation), as shown in Eq. (3) (Fig. 2, right). Iterative methods are developed based on a fixed-point estimation of Eq. (3) such as value iteration, policy iteration, or generalized policy iteration.

$$\max_{\pi(\cdot)} E\left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} R_\tau(s_\tau, \pi(s_\tau)) | s_t\right] \tag{2}$$

$$V(s_t) = \max_a \{R_t(s_t,a) + \gamma E[V(s_{t+1})|s_t,a]\} \tag{3}$$

The exact solution approach is thought to be inapplicable to most practical problems due to the computational challenge brought by the need to solve Eq. (3) for every state, which is

referred to as the *curse of dimensionality*. Approximate dynamic programming (ADP) (Powell, 2007) or reinforcement learning (RL) (Sutton and Barto, 1998) has been developed as a practical method for evolving a decision policy towards the optimal (or near-optimal) one online, forward in time, by using measured performance feedbacks along the system trajectories visited during simulations or real applications.

The main strategy of ADP is computing the sampled observation of a value $\hat{v}^n$ of being in state s^n, and using the pair $(s^n,\hat{v}^n)$ to estimate (or update the estimate) of the value function. General way for this is based on the time varying prediction error, which is referred to as temporal difference (TD), by replacing the original Bellman equation Eq. (3) into the recursion equation Eq. (4). That is, the sampled value is estimated using the feedback data and the value function obtained up to the previous iteration, and the value function is thus iteratively updated based on TD error Eq. (5). This is a forward-in-time algorithm, which can construct an on-line incremental policy.

$$\hat{v}^n(s_n) = \max_{a\in A^n}\left\{R(s_n,a) + \gamma E\left[\bar{V}^{n-1}(s')|s_n,a\right]\right\} \tag{4}$$

$$\delta^n = \hat{v}^n(s_n) - \bar{V}^{n-1}(s_n) \tag{5}$$

Another important ingredient of ADP is that some sort of function approximation is employed to represent the value function, instead of a lookup-table. One of the most powerful approximation strategies is to use a linear model with a basis function ϕ parameterized by state variables, as shown in Eq. (6). This is particularly useful when the value of the future given a state is easy to approximate (e.g., in multidimensional resource allocation problems), in which large problems can be easily approximated using separable approximations (Powell, 2014). In general, the dimension of separable features used for value function approximation (VFA) is much smaller than the dimension of state space, so a reliable value function can be obtained with a much smaller number of samples.

$$\bar{V}(s|\theta) = \sum_{f\in F}\theta_f\phi_f(s) \tag{6}$$

In more complex situations, popular choices of value function approximator have been artificial neural networks, esp. deep neural networks.

2.3. Linkage between MSSP and DP

There are some studies on making a bridge between MSSP and DP (DP is referred to as *optimal control* in some studies). A MSSP problem can be reformulated in the DP framework by temporal decomposition methods and defining state and decision variables from the original problem (Barro and Canestrelli, 2016). On the other hand, a formulation of DP can be converted into the equivalent MSSP model by switching from a true model of the future to an approximate lookahead model (e.g., estimating the value function with the sum of sampled future costs and replacing expectation with an average over a set of sample paths) and replacing time-dependent policy with a vector-valued decision (Powell, 2012).

Comparing the aforementioned aspects of the two modeling approaches (Barro and Canestrelli, 2016; Dupačová and Sladký, 2002), MSSP can avoid the requirement of the Markov structure and efficiently formulate realistic constraints, but it is difficult to be extended to the multi-period setting since the number of scenarios tends to increase exponentially. On the other hand, MDP can efficiently treat a distant or infinite horizon problem, but the applicable problem structure is limited. Especially, for optimization problems that have complex dynamics with many state variables and many constraints, it is difficult to formulate a computationally feasible MDP due to the need for a high-dimensional state space. In conclusion, rather than being competing methodologies, they are more complementary in nature with different favorable and unfavorable features, and thus each one can be specialized to efficiently treat particular classes of problems.

Recognizing the distinctive strengths of MSSP and DP and their complementary nature, new modeling approaches have been explored with one-way information exchange between them. In the study (Cheng et al., 2003), a problem of design and planning under uncertainty is formulated as a multi-objective MDP, in which the single-period problems are 2SSPs. That is, model information of the MDP is provided from a 2SSP model. In other studies (Cheng et al., 2004; Konicz et al., 2015), MSSP and MDP models are combined in a sequential manner. MSSP is implemented in the first subhorizon with realistic constraints, with the value function added at the end of subhorizon, which is evaluated by solving the MDP for the remaining period characterized by relaxed constraints or simple scenarios. That is, the end-effects of a look-ahead policy are resolved by adding the value function capturing all future reward as a terminal penalty.

In these studies, MDP is solved by DP without taking computational advantage of ADP or RL. As a result, they could examine only a small-sized example to demonstrate the proposed approaches and conclude that scalability of the rigorous algorithm is limited, suggesting the need for approximating approaches to solve realistic problems.

3. Model development

At the high level in decision hierarchy, strategic plans (e.g., production or procurement goals) are set to maximize margin (or minimize cost) that meets the market demand and supply constraints. Typically, the decisions are made based on known state and simple recurring dynamics of physical state and information flow of the system. In addition, such a plan should be made as a long-term strategy capturing high-level future uncertainty evolving over long-time. All these argue that the planning problem is naturally expressed as an MDP with a longer time span.

Meanwhile, the detailed schedule or operational decision within each single planning period is needed for execution of the plan. The decisions are made to minimize the operational cost while respecting the complex processing dynamics and constraints (e.g., capacity, product spec, and demand satisfaction) and the predetermined plan in a faster time span for the (finite) corresponding planning period. Thus they are better suited for look-ahead policies using MP. In particular, rolling horizon procedures or MSSPs can be used to ensure timely response to frequently realized uncertainties. However, the end-effects by the use of finite horizon need to be resolved from the perspective of the long-term strategy.

Based on this realization, a multi-timescale decision-making model is developed in which the high-level planning and the low-level scheduling are separately modeled as an MDP and MP, respectively, and are efficiently combined by bi-directional information exchange in a complementary way. The MDP model of state transition and reward for one planning period is determined by the *optimization-embedded simulation* of the low-level MP model, instead of a coarse-grained approximate model which can lose relevant fast time-scale dynamics and uncertainty. On the other hand, the long-term value of being in a system state (which is called *value function*) is evaluated in the high-level (capturing the high-level future uncertainty), and fed back to the look-ahead scheduling model for resolving the end-effects. Before the detailed model formulation, the time index used in the proposed model and the

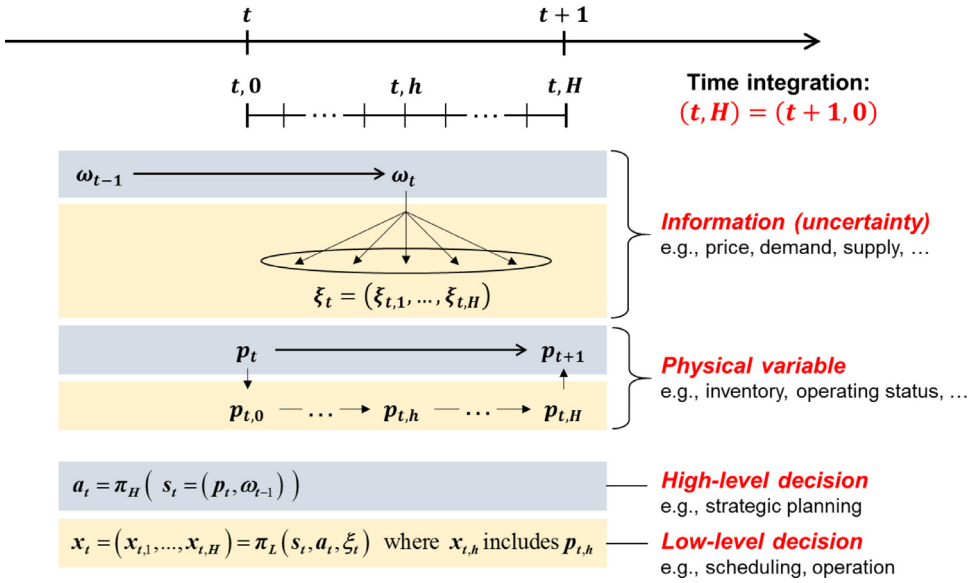

Fig. 3. Temporal multiplicity of time-varying variables and decision-making process.

temporal multiplicity of time-varying variable and decision-making process are described first (Fig. 3).

3.1. Time index

A single time period at the high level is the whole decision horizon at the low level; a planning period t is divided into H equal length time segments indexed as h on a faster time scale. In the integrated model, temporal integrity is maintained by $(t, H) = (t+1, 0)$.

3.2. Time-varying variable

Recurring dynamics of both physical state of system (e.g., inventory, commitment status of units) and information on exogenous uncertainty (e.g., price, demand, supply), which is provided as an essential parameter of the decision models from the external model or data, should be considered. The physical state p_t at the beginning of period t is considered in the planning level, and the state at the next period $t+1$ is updated by the actual operational decisions (through stochastic simulation). In the case of informational variable, the representative value ω_t (e.g., average) during the horizon $(t,1),\ldots,(t,H)$ is taken into account at the higher-level, and a number of look-ahead scenarios $\xi_t = (\xi_{t,1},\ldots,\xi_{t,H})$ realized on the faster timescale are considered in the decisions at the lower-level, where the scenario realization is associated with ω_t.

3.3. Decision-making process

The high-level decision a_t is made based on the information available at the beginning of period t, defined by state variable $s_t = (p_t, \omega_{t-1})$. The low-level decision $x_t = (x_{t,1},\ldots,x_{t,H})$ is thus determined by the current state, higher-level decision, and newly realized uncertainty scenarios, where $x_{t,h}$ includes $p_{t,h}$.

$$a_t = \pi_H(s_t), \quad \text{where } s_t = (p_t, \omega_{t-1}) \tag{7}$$

$$x_t = (x_{t,1}, \ldots, x_{t,H}) = \pi_L(s_t, a_t, \xi_t), \quad \text{where } x_{t,h} = \left(p_{t,h}, x'_{t,h}\right) \tag{8}$$

3.4. Model formulation

The proposed multi-timescale decision-making model comprises three sub-modules: an MDP-based high-level planning, an MP-based look-ahead operation, and a multi-timescale uncertainty model. Each element of the model is efficiently integrated through information exchange between them, as illustrated in Fig. 4.

3.4.1. Multi-timescale uncertainty model

There exists a high level of variations in information essential to decision making (demand, price, yield, etc.) along the time, and these uncertainties are generally stochastic and irreducible because they are caused by inherent variation of the physical system or uncontrollable external environment. Thus such uncertainties should be forecasted and proactively captured in decision-making process for sustainable management and operation of the system. For this, various statistical tools and machine learning techniques can be used to construct valuable models and knowledge from the extensive historical or frequently-updated data.

Variations of the uncertainty can have different characteristics (e.g., nonstationary, deterministic, periodic, stochastic) across the timescales. In order to integrate decision layers having different timescales, a multi-timescale uncertainty model that is consistently valid for all the considered timescales should be constructed. That is, the model should be able to provide both the transition probabilities of the stochastic parameters evolving with the timescale of the high level (Eq. (9)) and the forecasting look-ahead scenarios of the uncertainty for the low-level decision horizon with the fine timescale (Eq. (10)), in a consistent way. Here, a set of possible look-ahead scenarios $\xi_t = (\xi_{t,1},\ldots,\xi_{t,H})$ realized for time t is associated with the high-level state ω_t representing the corresponding horizon (e.g., average or initial value).

$$\Pr(\omega_{t+1}|\omega_t) \text{ for } \forall \omega_t, \omega_{t+1} \in \Omega_t \tag{9}$$

$$\xi_t = \left(\xi_{t,h=1}, \ldots, \xi_{t,h=H}\right) \in \Xi(\omega_t) \tag{10}$$

3.4.2. Markov decision process (MDP) based high-level planning

The high-level planning problem is formulated as a MDP with time index, $t = (t, 0)$. The fundamental components of the upper-level MDP $< s, a, R, P >$ are described as follows: State variable s_t and decision variable a_t at the beginning of time t are defined as Eq. (7). One (planning) period reward (or cost) function is defined by the planning-level reward r, determined by s_t and a_t, as well as the operation-level reward q, which is evaluated from the simulation of the low-level decision x_t for the corresponding time period:

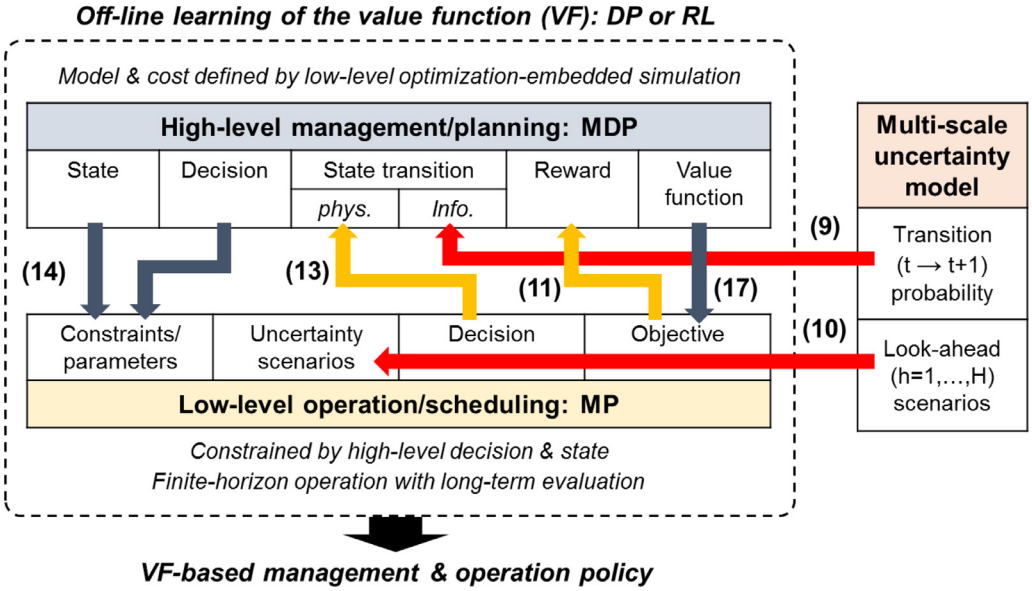

Fig. 4. Diagram of main elements of the proposed multi-timescale decision-making architecture and information flow between the elements (the number in the arrow represents the corresponding equation).

$$R(s_t, a_t) = r_H(s_t, a_t) + r_L(s_t, a_t) \tag{11}$$

The overall state transition probability from the current state and decision to the next state is defined by Eq. (12). The transition probability of exogenous information ω_t is given by Eq. (9), and the physical state transition is determined by the low-level solution $x_{t,H}$, stated as Eq. (13).

$$P(s_{t+1}|s_t, a_t) = P(p_{t+1}|p_t, a_t, \omega_t)P(\omega_t|\omega_{t-1}) \tag{12}$$

$$P(p_{t+1}|p_t, a_t, \omega_t) = \begin{cases} 1 & \text{if } p_{t+1} \in x_{t,H} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

3.4.3. Mathematical programming (MP) based look-ahead operation

The low-level operational decisions or schedules are derived from MP-based approaches to address complex processing dynamics and constraints. Here, the optimization model for the low-level is described in the form of MP without loss of generality. In MP, note that a stage defined by the point of decision-making must be distinguished from a period referring just to a point of time. Here, the stage and period are assumed to be the same, but they may differ depending on how you define the problem.

The detailed schedule or operational decision x_t in every single planning period t is made for the horizon $(t,1),\ldots,(t,H)$, capturing the fine physical dynamics of the system, as defined in Eq. (8). Some of initial conditions or constraints in the MP-based operation model may be determined by the higher-level state and decision.

$$\begin{aligned} &f_{1i}(x_{t,1}, (p_{t,0} = p_t), a_t) \le 0, i = 1, \ldots, m_1 \\ &f_{hi}(x^{t,h}, (\xi^{t,h-1} \in \Xi(\omega_t)), (p_{t,0} = p_t), a_t) \le 0, i = 1, \ldots, m_h, \\ &h = 2, \ldots, H \end{aligned} \tag{14}$$

Expectation over a set of look-ahead uncertainty scenarios $\Xi(\omega_t)$, given by the uncertainty model Eq. (10), is to be maximized, as shown in Eq. (15). In general form, the objective function q of MP can be expressed as Eq. (1). The effect of business-level uncertainty (price, supply or demand) on the process-level (in terms of margin or cost) is evaluated through the system's operating model. As a result, the resulting optimal objective value and the corresponding solution are used to estimate the knowledge required for the planning problem, i.e., for computing the reward in Eq. (11) and the state transition in Eq. (13).

$$\begin{aligned} &r_L(s_t, a_t) = E_{\xi \in \Xi(\omega_t)}[\max_x q(x|s_t, a_t, \xi_t)] \\ &\text{where } q(x|s_t, a_t, \xi_t) = q_1(x_1) + E_{\xi_1}[q_2(x_2, \xi^1) \\ &+ E_{\xi_2|\xi^1}[\ldots E_{\xi_{T-1}|\xi^{T-2}}[q_T(x_T, \xi^{T-1})]]] \end{aligned} \tag{15}$$

3.5. Model solution

In the proposed model, the value function is computed offline through an optimization embedded simulation using a number of random samples, capturing recurring physical dynamics, uncertainties, and decisions. That is, simulated sample for (physical) state transition and reward are provided from solving the lower-level optimization problem with realized uncertainty. The exact DP algorithm can be applied to small-size problems, but learning or function approximation based algorithms are required for large-size problems to ensure computational feasibility. Since RL (or ADP) solves MDP by simulating the stochastic environment and iteratively improving a decision policy by learning, it is a very fitting approach for integrating the two layers. Therefore, with a set of training samples generated by the uncertainty model (via Algorithm 1) the value function is learned by the optimization-embedded RL algorithm (Algorithm 2), as shown in Fig. 5.

To perform the stepping forward simulation for learning the value function, multi-timescale random samples are generated by Algorithm 1. A set of random samples representing the high-level uncertain information (called *sample path*) $\omega = (\omega_1,\ldots,\omega_n,\ldots,\omega_N)$ and a set of look-ahead scenarios for each iteration $\Xi_n = \Xi(\omega_n)$ are generated by the multi-timescale uncertainty model. The high-level decision is made over the scenario set Ξ_n before realizing the low-level uncertainty, and the low-level decision is made with one of the scenarios $\xi_n \in \Xi_n$ (which is thought as a realized uncertainty) for stepping forward in the simulation.

A general ADP algorithm for learning the (approximated) value function in the proposed multi-timescale decision-making model is shown in Algorithm 2. Here, Algorithm 2 introduces a value iteration (VI)-based algorithm that recursively updates the value function (and consequently the policy) for every iteration, but batch-

Algorithm 1 Multi-timescale random sample generation by the uncertainty model.

Input: ω_1, N
Step 1. Generate a path of high-level uncertainty $\omega = (\omega_1, \ldots, \omega_N)$ by the transition probability of Eq. (9).
For $n = 1, \ldots, N$
Step 2. Generate a set of low-level uncertainty scenarios $\Xi_n = \Xi(\omega_n)$.
Step 3. Choose a scenario $\xi_n \in \Xi_n$ to realize.
Output: ω, Ξ_n, ξ_n

Algorithm 2 A general RL algorithm for TD-based value function learning in the proposed multi-timescale decision-making model.

Input: N random samples for training (ω, Ξ_n, ξ_n) generated by Algorithm 1
Step 0a. Construct an approximation model for the value function, and initialize the parameter vector θ^0.
Step 0b. Choose an initial physical state p_1.
For $n = 1, \ldots, N$
Step 1. Decision-making
  Step 1a. Make a high-level decision a_n (s_n, $\Xi_n \mid \theta^{n-1}$) over the scenario set Ξ_n.
  Step 1b. Make a low-level decision x_n (s_n, a_n, $\xi_n \mid \theta^{n-1}$) with the realized scenario ξ_n.
Step 2. Observe the corresponding reward $R(s_n, a_n)$, and transitioned state $s_{n+1} = T(s_n, a_n, x_n, \omega_n, \xi_n)$.
Step 3. Value Function update
  Step 3a. Compute the TD error δ_n
  Step 3b. Update the value function parameter vector θ^n.
Output: θ^N

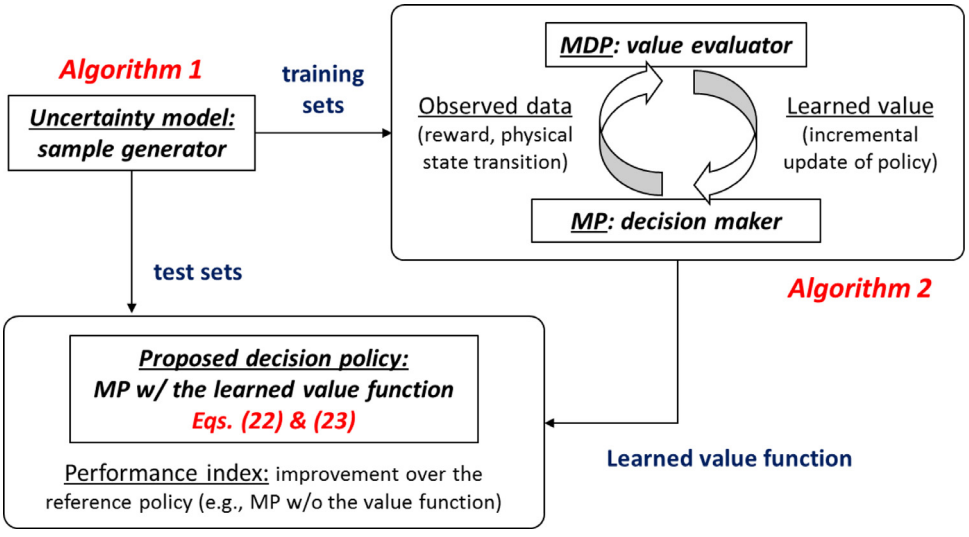

Fig. 5. Schematic diagram of the off-line value function training and performance evaluation steps performed in this study.

data based policy evaluation (value function estimation) algorithms can be used, which is referred to as policy iteration (PI).

3.5.1. Step 0. Initialization

An approximation model for the value function is constructed with an initial parameter vector θ^0. When the value of the future given a state has an explicit and linear relationship with some descriptors (i.e., state variables or combinations of state variables), it can easily be approximated using a linear regression model. If the relationship is nonlinear and complex, a neural network (NN), which is capable of representing general nonlinearities, may be employed. In NNs, however, use of too many hidden layers or neurons can lead to 'overfitting', so determining a proper structure of the NN is very important and the update algorithms need to be fined-tuned. In many strategic decision-making problems (e.g., planning, resources allocation), the relationship between the future value and the state variables (i.e., inventory or price) is fairly intuitive, which facilitates the value function approximation. Thus, a linear model with a basis function ϕ parameterized by the state variables, as shown in Eq. (6), is employed in this study.

3.5.2. Step 1. Decision-making

At each iteration given the current state and uncertainty scenarios, the high-level decision is made according to the MDP solution structure:

$$a_n(s_n, \Xi_n | \theta^{n-1}) = \arg\max_{a \in A^n} \{ r(s_n, a) + E_{\xi \in \Xi_n} [\max_x q(x | s_n, a, \xi) + \gamma \tilde{\theta}^{n-1} \tilde{\phi}(p^{\xi}_{n,H})] \} \quad (16)$$

where $\tilde{\phi}$ is the subvector of the basis function of which components are related to the physical state p, and $\tilde{\theta}$ is the coefficient vector corresponding to the components. Note that the transition of exogenous information variable is independent on decision. This procedure contains the low-level optimization (MP) in which the objective function is defined by adding the value function of being in an end-horizon (physical) state to Eq. (15) and including the low-level constraints Eq. (14). After observing the realized low-

level uncertainty ξ_n, the low-level decision is made:

$$x_n\left(s_n, a_n, \xi_n|\theta^{n-1}\right) = \underset{(x_{n,1},\ldots,x_{n,H})}{\arg\max}\, q(x|s_n, a_n, \xi_n) + \gamma\tilde{\theta}^{n-1}\tilde{\phi}\left(p_{n,H}^{\xi_n}\right) \tag{17}$$

3.5.3. Step 2. Feedback data observation

With the optimization-embedded simulation of the environment (system), the corresponding reward and transitioned state are observed:

$$R(s_n, a_n) = r(s_n, a_n) + q(x_n|s_n, a_n, \xi_n) \tag{18}$$

$$s_{n+1} = (p_{n+1}, \omega_n), \quad \text{where } p_{n+1} \in x_{n,H} \in x_n(s_n, a_n, \xi_n) \tag{19}$$

3.5.4. Step 3. Value function update

The TD error δ_n defined by Eq. (5) is calculated from the simulated feedback data of reward and state transition as shown in Eq. (20), and the coefficient vector of the value function is then updated based on this error using an updating rule U^θ (e.g., gradient-based approaches, or least-squares techniques):

$$\delta^n = R(s_n, a_n) + \gamma\theta^{n-1}\phi_{n+1} - \theta^{n-1}\phi_n, \text{ where } \phi_n = \phi(s_n) \tag{20}$$

$$\theta^n \leftarrow U^\theta(\theta^{n-1}, \delta^n, s_n) \tag{21}$$

Note that TD-based updates are *sample backups*[1] that update the value estimate for the visited state on the basis of the one sample transition from it to the immediately following state (Sutton and Barto, 1998). Stochastic gradient method minimizing the mean squared error (MSE) of TD yields the update rule in the following basic form:

$$\theta^n = \theta^{n-1} - \alpha_n\nabla\left\{\frac{1}{2}\left(\bar{V}^{n-1}(s^n) - \hat{v}^n\right)^2\right\} = \theta^{n-1} + \alpha_n\phi_n\delta^n$$

Tuning the stepsize α_n is important for the sample-based adaptive estimation approach and is generally set to decrease as learning progresses to ensure convergence of the estimates. The degree of stepsize decrease should be adjusted depending on the level of noise and nonstationarity of the sampled data, and the initial estimates. In the recursive least squares (Wang, 1986), the scaling matrix H^n is used instead of the scalar stepsize, and is systematically updated as follows

$$H^n = \frac{1}{\gamma^n}B^{n-1}$$
$$\gamma^n = 1 + {\phi_n}^T B^{n-1}\phi_n$$
$$B^n = B^{n-1} - \frac{1}{\gamma^n}\left(B^{n-1}\phi_n{\phi_n}^T B^{n-1}\right)$$

3.5.5. Proposed decision policy

For each iteration, the observation of a value of being in a state is computed through Eq. (4) assuming that the determined decision from the iterated value function is optimal. As iteration increases, the *learning policy* (the policy we are trying to learn) is improved based on the value function that is approximated more closely to the original one. Finally, the high- and low-level decision policies, as defined in Eqs. (16) and (17) respectively, can be constructed with the trained value function θ^N with N numbers of iterations:

$$\pi_H(s, \Xi|\theta^N) = \arg\max_a\{r(s, a) + E_{\xi\in\Xi}[\max_x q(x|s, a, \xi) + \gamma\tilde{\theta}^N\tilde{\phi}(p_{\cdot,H}^{\xi})]\} \tag{22}$$

$$\pi_L(s, a, \xi|\theta^N) = \underset{(x_{\cdot,1},\ldots,x_{\cdot,H})}{\arg\max}\, q(x|s, a, \xi) + \gamma\tilde{\theta}^N\tilde{\phi}(p_{\cdot,H}^{\xi}) \tag{23}$$

The proposed multi-timescale decision-making model enables to introduce more sustainable and flexible management and operation strategies by addressing the following challenges posed by temporal multiplicity and uncertainty. Real operation and faster-scale uncertainty scenarios can be reflected directly to create a more realistic, hierarchically optimal plan. By developing a multi-timescale uncertainty model that is consistently valid for both timescales, future disruptions can be captured proactively in the decision-making process, resulting in more robust plans and schedules that proactively account for the effects of uncertainty. In addition, with the end-state evaluative information in the look-ahead policy calculation, the end-effects of using a finite decision horizon can be resolved by systematically quantifying the balance between the cost for the considered decision horizon and the future cost thereafter. The strength of this model is its versatility making it useful in a variety of areas where the structures of decision-making and environment are similar.

4. Practical example

The studied refinery flow chart is described in Fig. 6 (Favennec, 2001); it contains typical processing units such as the crude distillation unit (D), reformer (R), cracker (Cr), isomerization (I), desulphurization (Des), refinery fuel (RF), and blending units for the final products (PG98, ES95, JF, DSL, HF). The products are separated into two groups; final products are produced for sales; all other products are categorized as intermediate products. The intermediate products are internally used to make final products, with an exception of LG which can be sold outside. It is assumed that the inlet products for the reformer and the cracker can be imported from outside.

The studied refinery procurement and production planning problem is illustrated in Fig. 1. Decision epoch for the crude procurement is assumed to be H-days (indexed as t), and the one for refinery operation and product export is a day (indexed as h). That is, crude oils are selected and purchased with the price at time $t=(t, 0)$, and the produced products are sold with the price at time $t+h=(t, h)$.

According to the proposed method, an MDP is formulated with time index t for evaluating the value function given crude inventory (physical) and price (information) state, in which refining margin (reward) and inventory transition are computed by the refinery optimization model at finer time scale h for $(t,0),\ldots,(t,H)$. The overall optimization for the decision period t can be formulated as an MSSP where the first stage decisions are crude imports followed by operational decisions including product exports based on price uncertainty realization. For this, a daily price model is developed based on real data. The formalized optimization-embedded MDP problem is solved by value iteration with a number of samples and a linear approximation model for the value function (Algorithm 1), and the resulting value function is integrated into the operation model as an evaluative measure of crude oil storage at a given price scenario.

4.1. Problem formulation

4.1.1. One-day model of refinery management and operation

One-day refinery operation model is established referring to the model and parameters specified in (Favennec, 2001). Following

[1] In the model-based approaches, the expectation in Eq. (4) is computed by a one-step transition matrix (called *full backups*) or is approximated using a set of randomly generated outcomes. In the model-free approaches, we can use simple substitution to replace Eq. (4) with $R(s_n, a_n) + \gamma\bar{V}^{n-1}(s_{n+1})$ observing a reward $R(s_n,a_n)$ and the next state s_{n+1} (called *sample backups*).

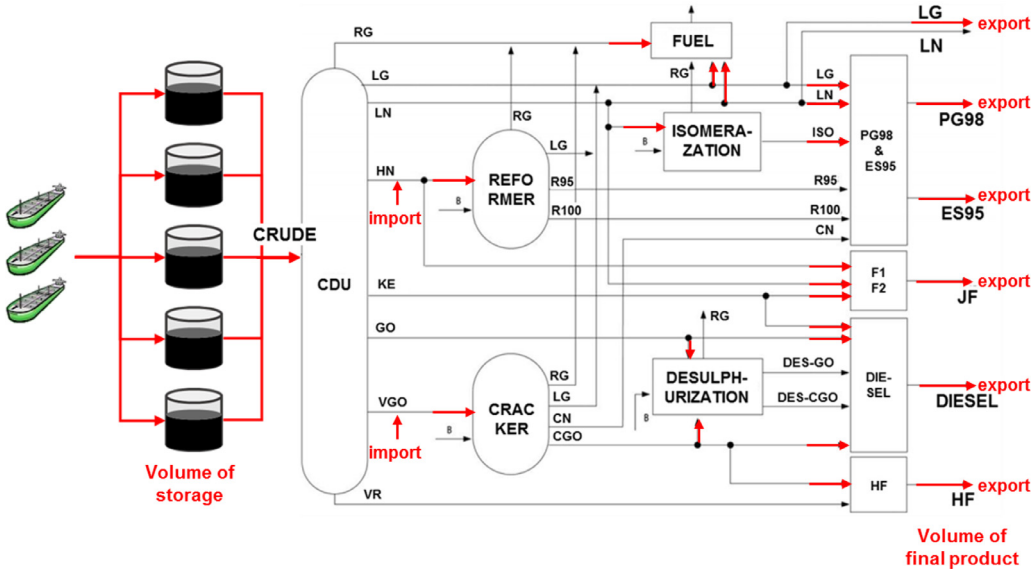

Fig. 6. The studied refinery flow chart; arrows and tank volumes indicate decision variables. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

sets and indices are defined for the formulation: $c \in C = \{\text{crudes}\}$, $u \in U = \{\text{processing units}\}$, $k \in K = \{\text{all products including intermediates}\}$, $f \in F = \{\text{final products}\}$, $i_u \in I_u = \{\text{inlet products supplied to unit } u\}$, $b_f \in B_f = \{\text{blended materials for product } f\}$. Crude procurement and operational decisions represented as red arrows and variables in Fig. 6, should be optimally determined to maximize the refining margin. Here, only the decision variables and objective function are described for explaining the overall decision-making framework, and all the detailed formulation can be found in Appendix A.

In the crude procurement stage, the binary value y denoting the selection of crude to be purchased (among those in C) and the purchase amount x^C of the crudes should be decided.

$$a = \left\{ y_c, x_c^C \right\} \tag{24}$$

The resulting reward is defined as the cost of purchasing the crudes, where P_c is the crude price.

$$r(a, P) = -\sum_c P_c x_c^C \tag{25}$$

Consequently, the following operational decisions are determined: the inventory p_c of crude c, the flowrate x^I of inlet material i_u to unit u, the flowrate x^B of blended material b_f for final product f, the amount of import x^{IMP} and export x^{EXP} product k, and the volume v of final product f to be produced.

$$x = \left\{ p_c, x_{u,i_u}^I, x_{f,b_f}^B, x_k^{IMP}, x_k^{EXP}, v_f \right\} \tag{26}$$

The resulting reward is defined by the profit from product export minus the purchase price of product import and operating cost, where P_i is the price of product i and C_u is the operating cost for each unit.

$$q(x, P) = \sum_k \left(P_k x_k^{EXP} - P_k^{IMP} x_k^{IMP} \right) - \sum_u C_u \sum_{i_u} x_{u,i_u}^I \tag{27}$$

The number of crude oils that can be selected in a day and availability of each crude supply should be constrained. In refinery operation, material balances for all the materials (crude inventory, intermediate and final products, and refinery fuel), quality constraints for the final products, capacity restriction of each unit should be considered (refer to Appendix A for full detailed formulation of these constraints.):

$$Aa \le b_0,\ Ta + Wx \le b_1,\ y \in \{0, 1\},\ a, x \ge 0$$

4.1.2. Price uncertainty model

Daily price data of crude and fuel products from 2014.12 to 2017.3 (about 600 days) available from U.S. Energy Information Administration are used. Since the prices of various crude oils share a same trend, it is common to model them using the price of the marker crude (WTI). Thus, the price data of WTI and three fuel products (GSL: gasoline, JF: jet fuel, DSL: diesel) are modeled together as a first order Markov chain (MC):

$$\xi_{h+1} = \xi_h + \hat{\xi}_{h+1},\ \text{where } \hat{\xi} = (\xi_{\text{WTI}}, \xi_{\text{GSL}}, \xi_{\text{JF}}, \xi_{\text{DSL}}). \tag{28}$$

Each price variable is discretized into eight spaces based on the minimum and maximum values of the data, and the representative value of each space is defined as intermediate value. The total size of price state space is thus 8^4, but a reduced state space of 90 in size is sufficient, as there is a strong correlation between crude oil price and fuel product prices. One-day transition probability of the discretized price space is obtained from the data, where n_{ij} represents the number of transitions from state i to state j during a day.

$$\Pr(\xi_{h+1} = j | \xi_h = i) = \frac{n_{ij}}{\sum_j n_{ij}} \text{for } i, j \in \Omega \tag{29}$$

A set of look-ahead price scenarios Ξ for H-days given a price state $\xi_{t,0}$ can thus be obtained where the realization probability of each scenario $\Pr(\xi_t \mid \xi_{t,0})$ is computed by the product of the daily transition probability Eq. (29) along the scenario path.

$$\Pr(\xi_t | \xi_{t,0}) = \prod_h \Pr(\xi_{t,h} | \xi_{t,h-1}),\ \text{for } \xi_t = (\xi_{t,1}, \ldots, \xi_{t,H}) \in \Xi(\omega_t = \xi_{t,0}) \tag{30}$$

Meanwhile, since a plan of crude purchase is made every H days, which is indexed as t, the price transition probability from time t to $t+1$ can be computed by H-times product of the daily transition matrix P_h, of which component (i, j) is defined by one-day transition probability from state i to state j, $\Pr(\xi_{h+1} = j \mid \xi_h = i)$ in Eq. (29).

$$\Pr(\omega_{t+1} | \omega_t) = (P_h)^H,\ \text{where } \omega_t = \xi_{t,0},\ \omega_{t+1} = \xi_{t,H} \in \Omega \tag{31}$$

Fig. 7 illustrates an example of look-ahead scenario generation and the transition probability at time t given a price state of 1 when $H = 2$. As a result, the overall multi-timescale price uncertainty model is featured by the following quantities: a set of look-

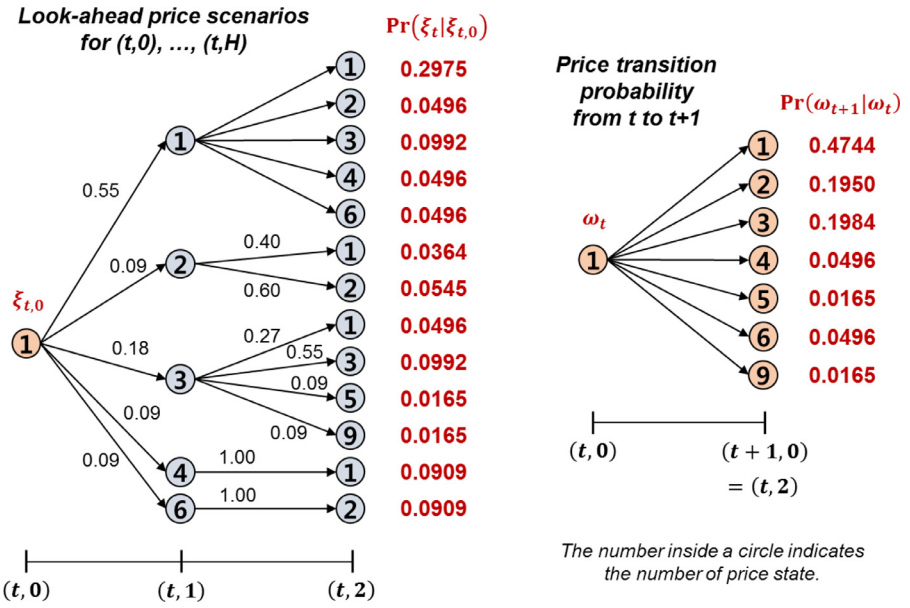

Fig. 7. An example of a set of look-ahead scenarios for time t and the probability of each scenario realization (left) and the transition probability from t to $t+1$ (right) given a price state of 1 when $H=2$.

ahead scenarios $\Xi(\omega_t)$ and their probabilities $\Pr(\xi_t|\omega_t)$, and discrete price space Ω and price transition probability $\Pr(\omega_{t+1}|\omega_t)$ from t to $t+1$.

4.1.3. Multi-timescale decision-making

Based on the state s_t of crude inventory (physical variable) and price (information) at the beginning of time $t=(t,\ 0)$, a plan a_t of crude selection and the amount of crude purchase during the horizon $(t,1),\ldots,(t,H)$ is first made. Then, after establishing a crude procurement plan, all the operational decisions $x_{t,h}$, defined by Eq. (26), are made for all days $(1,\ldots,H)$ as recourses.

$$s_t = (p_c,\ \omega_i \text{ for } \forall c \text{ and } i \in \{\text{WTI, GSL, JF, DSL}\})_t \tag{32}$$

$$a_t = \left(a_{t,h} \text{ for } h = 1, \ldots, H\right) \tag{33}$$

$$x_t = \left(x_{t,h} \text{ for } h = 1, \ldots, H\right) \tag{34}$$

The high-level (crude procurement) reward r is defined as the sum of all crude purchasing cost Eq. (25) for $(t,1),\ldots,(t,H)$, and the subsequent low-level (refinery operation) reward q is the expected sum of all refining margin during the corresponding time horizon, where one-day refining margin q_h is defined by Eq. (27):

$$r(s_t, a_t) = \sum_h r_h\left(a_{t,h}, \omega_t\right) \tag{35}$$

$$q(x_t|s_t, a_t, \xi_t) = E_{\xi_{t,1}}\left[q_1(x_{t,1}, \xi_{t,1}) + E_{\xi_{t,2}}\left[\ldots E_{\xi_{t,H}}[q_H(x_{t,H}, \xi_{t,H})]\right]\right] \tag{36}$$

Consequently, the overall decision-making problem during time t is formulated as a MSSP with $H+1$ stages, in which the first stage is crude procurement and the following stages are refinery operation and product export under the price uncertainty (Fig. 1). Here, all other constraints are daily-independent except for the equation represents inventory dynamics.

$$\begin{aligned} &\max_a \left\{r(s,a) + E_{\xi\in\Xi}\left[\max_x q(x|s,a,\xi)\right]\right\}, \\ &\text{s.t.}\quad Aa \le b_0,\ T_h^{\xi} a + W_h x_h^{\xi} \le b_h^{\xi}, \\ &\qquad y \in \{0,1\},\ a, x_h^{\xi} \ge 0. \end{aligned} \tag{37}$$

Meanwhile, a MDP problem is formulated with time index t for evaluating a value being in a state, in which reward and inventory transition are computed by the refinery optimization model of MSSP. A parametric linear model is constructed for value function approximation, where the basis function ϕ is defined by linear and bilinear terms of state variables:

$$\phi(s) = [1 \quad \omega_i \quad \omega_i^t \quad \omega_i\omega_j \quad \omega_i\omega_j^t \quad \omega_i^t\omega_j^t \quad p_c \quad \omega_{\text{WTI}}^t p_c]^T, \tag{38}$$

where $\omega^t = \sum_{\omega'} \omega' \Pr(\omega'|\omega)$. This linear approximation model is established with the exact DP solution of a small-size problem, in which five types of crude oils are considered and each crude inventory is discretized into three levels, resulting in the state space of size 21,870. Using the converged value function obtained by the exact value iteration, the regression model Eq. (38) is constructed, of which statistical measures are $R^2 = 0.9515$ and normalized MSE $= 2.1296 \times 10^{-2}$. In this approximation model, the subvector of the basis function of which components are related to the physical state p and the corresponding coefficient vector can be defined as follow

$$\tilde{\phi}(p) = [p_c \quad \omega_{\text{WTI}}^t p_c]^T, \text{ and } \tilde{\theta} = [\theta_p \quad \theta_{\omega p}]. \tag{39}$$

The decision-making policy of crude procurement and refinery operation is then obtained in the form of Eqs. (22) and (23) with the off-line trained value function by Algorithm 2. In this study, *the least squares temporal difference* (LSTD) learning method (Bradtke and Barto, 1996) is employed for the value function estimation, which is one of the most prominent least-squares approaches. LSTD minimizes the mean squared projected Bellman error (MSPBE) defined by $\|V_\theta - \Pi T V_\theta\|^2$, where T is the Bellman operator, and Π is a projection operator that projects an arbitrary value function onto the space of parametrized functions. The closed-form least squares solution of the MSPBE with n numbers of data is given by

$$\theta^n = \left(\Phi_n^T\left(\Phi_n - \gamma\,\Phi'_n\right)\right)^{-1} \Phi_n^T R_n, \tag{40}$$

where $\Phi_n = [\phi_1,\ \phi_2,\ \ldots,\ \phi_n]^T$, $\Phi'_n = [\phi_2,\ \phi_3,\ \ldots,\ \phi_{n+1}]^T$, and $R_n = [R(s_1,\ a_1),\ldots,\ R(s_n,\ a_n)]^T$. It is possible to iteratively compute (40) as data become newly available through a method named recursive LSTD (RLSTD) (Dann et al., 2014). That is, the following RL-

Table 1
Crude information.

	VR yield (wt %)	Sulphur fraction in GO (wt%)	Price variation v_c ($/ton)	Des-cost ($/ton)
C1	11.45	0.19	0.9810	1.13
C2	14.09	0.56	0.9769	1.38
C3	16.55	0.14	0.9718	1.09
C4	18.97	0.23	0.9680	1.16
C5	20.38	1.26	0.9603	1.87
C6	21.08	0.16	0.9639	1.11
C7	23.62	0.21	0.9584	1.14

Table 2
Comparison of the proposed and the reference policies examined in the case study.

	Reference policy	Proposed policy
Decision-making model	MILP (decision horizon: H days)	MSSP (# of stages: $H+1$)
Price uncertainty accounted	Re-optimization (LP) in shrinking-horizon manner	Proactively through the MDP formulation
Value function incorporation	No	Yes (RL)

STD update rule is used for Eq. (21) in Step 3b of Algorithm 2.

$$\begin{aligned}
&\Delta\phi_n = \phi_n - \gamma\phi_{n+1}\\
&\gamma^n = 1 + \Delta\phi_n^T B^{n-1}\phi_n\\
&\theta^n = \theta^{n-1} + \tfrac{1}{\gamma^n} B^{n-1}\phi_n\delta^n\\
&B^n = B^{n-1} - \tfrac{1}{\gamma^n} B^{n-1}\phi_n\Delta\phi_n^T B^{n-1}
\end{aligned} \tag{41}$$

Meanwhile, one of the most important issue in the learning-based decision-making process is to balance the tradeoff between either taking the best decisions according to the current state of knowledge (referred to as *exploitation*), or taking exploratory decisions, which may be less immediately rewarding, but may lead to better reward in the future (referred to as *exploration*). With pure exploitation, learning is 'accidental' and one can get trapped in a local optimum whereas a pure exploration strategy is difficult to use in the case that the budget of off-line training is limited or online learning is required. Therefore, in this study, ε-greedy policy is used as a *sampling policy* in Step 1a of Algorithm 2, where one explores random decisions with the probability of ε and exploits with the probability of 1- ε. Here, the exploratory decision is made by solving Eq. (16) with randomly pre-chosen and fixed crude selection decision y_c.

4.2. Numerical results

The introduced method is examined through a case study. Seven types of crude oils are used in this study, and crude-dependent parameters are specified in Table 1. The price of each crude oil varies by multiplying the price of marker crude (WTI) by v_c, which is a function of key qualities, residual (VR) yield and sulphur content. The unit operating cost of desulfurization is linearly increased with the sulphur content in its feedstock (GO/CGO) and differs from the refining crude. It is assumed that at most two different types of crude oil can be selected at each decision instance, and the lower and upper bounds of crude import amount is set to be 50 and 300 kton, respectively. Crude procurement plan is assumed to be made every two days ($H=2$), and discount factor γ for value iteration is set to be 0.98. All the required model parameters and economic data are specified in Appendix B.

In the case study, the performance of the proposed policy (π_1) is measured by the sum of improved reward (refining margin) over a reference policy (π_2) for 400 days ($T=200$), which is calculated as an average over a set of test random sample E generated for the performance evaluation:

$$\begin{aligned}
&E_{\{\xi,p_1\}\in \mathrm{E}}\left[\frac{R_{\pi_1}(\xi,p_1) - R_{\pi_2}(\xi,p_1)}{R_{\pi_2}(\xi,p_1)} \times 100\right],\\
&\text{where } R_\pi(\xi,p_1) = \sum\nolimits_t R(s_t^\pi, a_t^\pi).
\end{aligned} \tag{42}$$

where the random sample set E includes the price sample path ξ generated by Algorithm 1 for 400 days and randomly chosen initial inventory state p_1 for each crude oil, and $R_\pi(\xi, p_1)$ is the sum of reward obtained by policy π when the corresponding sample is given. In the reference policy, crude procurement decisions are made by an MILP model with decision horizon of H-days and the price uncertainty is reactively accounted by re-optimization (LP) in shrinking horizon manner along the realized uncertainty (Table 2).

The value function is trained in a varying learning environment where the number of iterations (N) and exploration rate (ε) are adjusted, which are the main factors affecting the learning performance. The performance of the value function learning is measured by Eq. (42) with 30 numbers of random samples ($|\mathrm{E}|=30$) where π_1 is the proposed policy using the learned value function. The result shown in Fig. 8 seems reasonable; a certain amount of iterations is required to learn a reliable information, but an appropriate exploration strategy will help due to the limited time (budget) to sample the measurements for learning.

When we analyse the trained value function (with 20,000 iterations and 0.2 exploration rate), the coefficient denoted by $\theta' = \theta_p + \theta_{\omega p}\omega^t_{\mathrm{WTI}}$ can be interpreted as the marginal value of storing the corresponding crude at a given price state. That is, it provides a quantitative assessment of what kind of, and how much crude oil to retain. Fig. 9 plots the difference between the marginal value θ'_c of crude inventory and the price ω_c of Crude 1 and 3 versus price state. The fact that the marginal value of a crude inventory is greater than the price means that it is beneficial to buy more crude oils than is necessary and to keep higher inventory level.

Fig. 10 plots the price state of product (diesel) vs. Crude 1 (above) and 3 (below) of which circle color and size represent the sign and absolute value of the difference ($\theta'_c - \omega_c$). The price state with blue circle satisfies $\theta'_c > \omega_c$. It can be seen that a higher inventory level is preferred when the crude oil is inexpensive (Crude 1), or the price difference from the product is large (Crude 3). That is, the marginal value of storing crude is systematically determined considering the current price of crude oil and products and the probability of its future change, as well as characteristics of each crude oil.

Monte Carlo simulation is implemented with 100 instances of price datasets generated by the MC model for 400 days and ran-

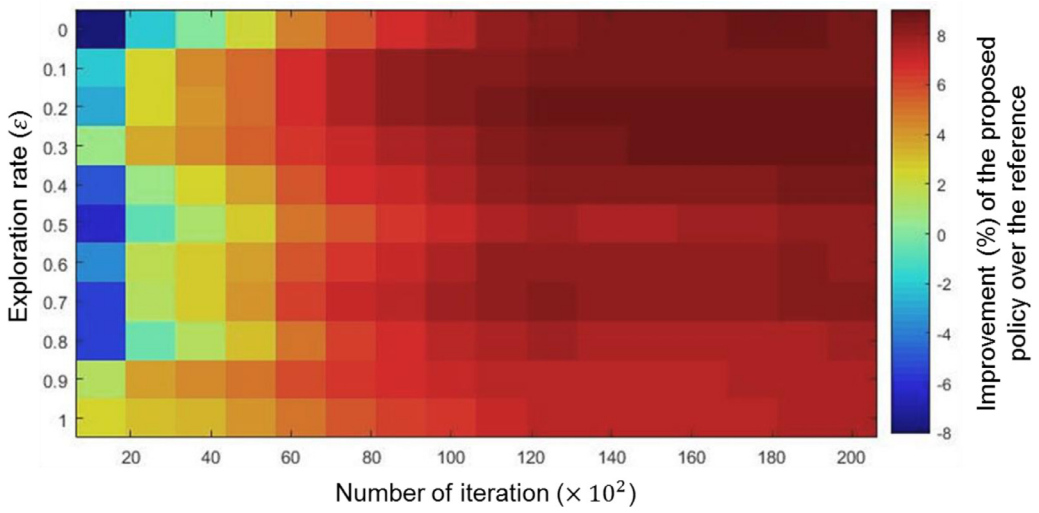

Fig. 8. Improvement (%) of the proposed policy over the reference with respect to the number of iterations and exploration rate for value function learning.

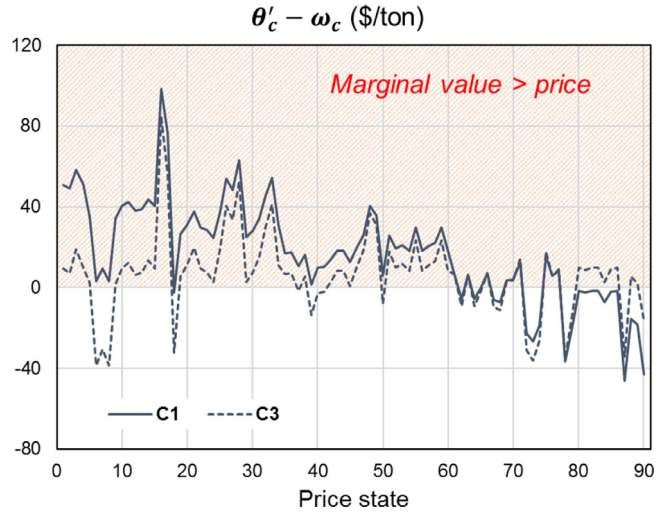

Fig. 9. The difference between the marginal value of crude inventory and price of Crude 1 and 3.

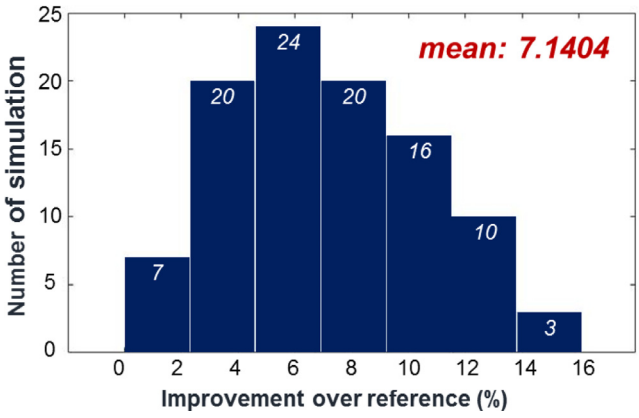

Fig. 11. Histogram of the sampled improvement of the proposed policy over the reference across 100 runs.

domly chosen initial inventory state for each crude oil. Fig. 11 shows the distribution of the sampled improvement of the proposed policy (value function learning with 20,000 iterations and 0.2 exploration rate) over the reference across 100 runs. We see that the proposed method outperforms the reference policy for all 100 runs, and the average improvement is 7.14%.

Table 3 shows the average crude purchase amount and inventory level of each crude oil. More than 80% of crude oil is imported from Crude 1 by the reference, while more widespread crude selection is done by the proposed on considering the price and inventory state. In addition, when the decision is made by the proposed policy, about 10 kton of extra crude oils are purchased on average, but the average inventory level is higher by 643.43 kton

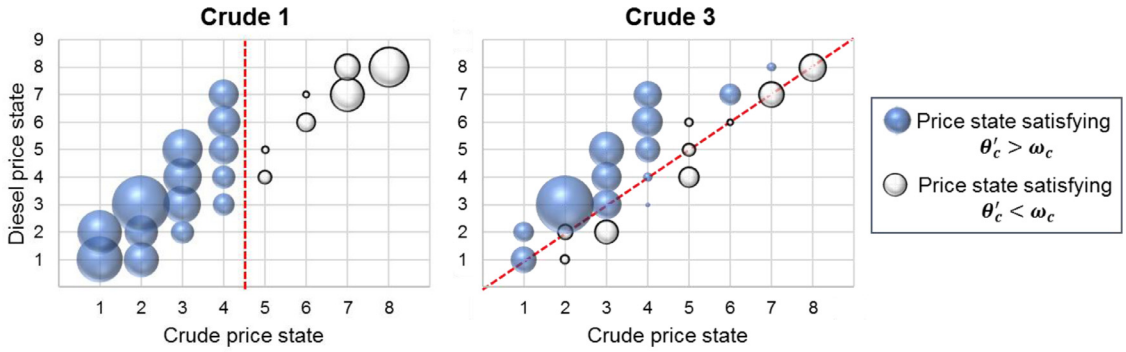

Fig. 10. The price state of diesel vs. Crude 1 (above) and 3 (below), of which circle type and size represent the sign and absolute value of the difference ($\theta'_c - \omega_c$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

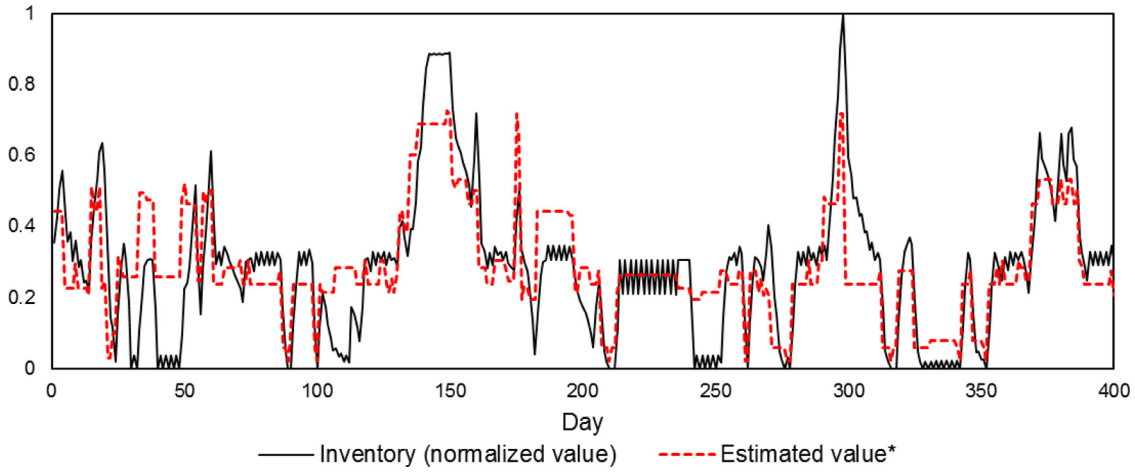

Fig. 12. Illustrative example of dynamic inventory management considering price fluctuations. The solid line represents the actual inventory level (normalized to the maximum value) managed by the proposed policy and the dashed line represents the estimate value regressed by price state. *Estimate values are obtained by Eq. (43) with $\beta_0 = 0.6695$, $\beta_{WTI} = -0.6514$, $\beta_{GSL} = 0.0625$, $\beta_{JF} = -0.2043$, and $\beta_{DSL} = 0.1127$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
The average purchase amount and inventory of each crude.

	Crude purchase (kton)		Inventory (kton)	
	Propose (%)	Reference (%)	Propose	Reference
C1	274.81 (73.36)	293.28 (80.36)	194.82	20.62
C2	43.34 (11.57)	48.82 (13.38)	60.39	3.99
C3	14.51 (3.87)	0	207.07	0.42
C4	11.67 (3.12)	0	58.95	0.41
C5	4.97 (1.33)	0.01 (0.00)	70.58	0.32
C6	24.21 (6.49)	22.84 (6.26)	42.87	5.17
C7	1.08 (0.29)	0	40.04	0.37
	374.59 (100)	364.93 (100)	674.73	31.30

compared to the reference. This can be seen as a result of more flexible adjustment of the timing of crude purchasing and processing using inventory in consideration of price fluctuation.

For more analysis, the inventory value is reversely estimated using the following regression model featured by normalized price state (by least squares):

$$\hat{p}_{t,h} = \beta_0 + \beta_{WTI}\tilde{\omega}_{WTI,t} + \sum\nolimits_i \beta_i \left(\tilde{\xi}_i\right)_{t,h} \text{ for } i \in \{\text{GSL, JF, DSL}\}, \tag{43}$$

Fig. 12 shows that the dynamics of the estimates (red dashed line) are similar to those of actual inventory managed by the proposed policy (black solid line). This means that crude inventory is dynamically managed with price fluctuations. More precisely, the lower the price of crude oil, the higher the profitability of purchasing more crude oil while the higher the price of products, the greater the benefit of processing more crude oil. Therefore, the suggested optimization-embedded MDP can provide a more flexible and robust inventory management policy considering the relative price state of crude and products.

5. Conclusion

In this study, we develop a multi-timescale decision-making model that combines Markov decision process (MDP) and mathematical programming (MP) in a complementary way and introduce a computationally tractable solution algorithm based on reinforcement learning (RL) to solve the optimization-embedded MDP problem. Specifically, the proposed modeling approach can address the following challenges: 1) Efficient integration of multi-timescale decision hierarchy by complementary use of MP and RL, 2) Proactive accounting of future uncertainty by taking advantage of a multi-timescale stochastic model, and 3) Reduction of end-effects with end-state value evaluation. The strength of the proposed decision-making approach is its versatility in a variety of application areas, and this study presents a practical example of refinery procurement and production planning to demonstrate the method. Based on the numerical results of a benchmark case study, we can conclude that the proposed method can provide a more robust and flexible management and operation strategy by considering the future dynamics of physical and information state.

Appendix A. Full formulation of refinery model (one-day)

The studied refinery flow chart is described in Fig. 6; it contains typical processing units such as the crude distillation unit (D), reformer (R), cracker (Cr), isomerization (I), desulphurization (Des), refinery fuel (RF), and blending units for final fuel products. The reformer has two operating modes according to the severity 95 and 100, and the cracker has two different operating modes, named as Mogas and AGO. The products are separated into two groups: gasoline (PG98, ES95), jet fuel (JF), diesel (DSL), and high-sulfur fuel oil (HF) are produced for sales; all other products are defined as intermediate products. The intermediate products are internally used to make the final products, with an exception of liquefied gas (LG) which can be sold outside. The inlet products for reformer (HN) and cracker (VGO) can be imported.

Following sets and indices are defined for the overall mathematical programming formulation: $c \in C = \{\text{crudes}\}$, $u \in U = \{\text{processing units}\}$, $k \in K = \{\text{all products including intermediates; D, R-95, R-100, Cr-Mogas, Cr-AGO, I, Des, RF}\}$, $f \in F = \{\text{final products; PG98, ES95, JF, DSL, HF}\}$, $i_u \in I_u = \{\text{inlet products supplied to unit } u\}$, $b_f \in B_f = \{\text{inlet blended materials for product } f\}$. For each unit u, a set of inlet products supplied to the unit is defined as follows: $I_D = C$, $I_{R\text{-}95} = I_{R\text{-}100} = \{HN\}$, $I_{Cr\text{-}Mogas} = I_{Cr\text{-}AGO} = \{VGO\}$, $I_I = \{LN\}$, $I_{Des} = \{GO_c, CGO\}$, $I_{RF} = \{RG, LG, LN, HF\}$. For each final product f, a set of blended products is defined as follows: $B_{PG98} = B_{ES95} = \{LG, LN, R95, R100, ISO, CN\}$, $B_{JF} = \{LN, HN, KE\}$, $B_{DSL} = \{KE, GO_c, CGO, DES\text{-}GO_c, DES\text{-}CGO\}$, $B_{HF} = \{CGO, VR_c\}$. Here, a subscript c in the product name is to distinguish the produced crude, and such a distinction is required since the products from different crudes have different properties.

A.1. Decision variables ($8|C|+40$)

In the crude procurement stage, the binary value y_c denoting the selection of crude $c \in C$ to be purchased and the purchase amount x_c^C of crude $c \in C$ should be decided. For the refinery operation, the inventory p_c of crude $c \in C$, the flowrate x_{u,i_u}^I of inlet material i_u to unit $u \in$ {D, R, I, RF}, the flowrate x_{f,b_f}^B of blended material b_f for $f \in F$, the amount x_k^{IMP} of import product for $k \in$ {HN, VGO} and the amount x_k^{EXP} of export product for $k \in$ {LG}$\cup F$, and the volume v_f of final product $f \in$ {PG98, ES95, DSL, HF} should be decided.

$$a = \left\{ y_c, x_c^C \right\}$$

$$x = \left\{ p_c, x_{u,i_u}^I, x_{f,b_f}^B, x_k^{IMP}, x_k^{EXP}, v_f \right\}$$

A.2. Constraints for crude procurement process ($2|C|+2$)

The number of crude oils that can be selected in a day is limited due to the limited capacity of vessel unloading capacity, and availability of crude supply are constrained as well according to the crude selection:

$$1 \le \sum_c y_c \le \mathrm{NC}^{\max}$$

$$\mathrm{AV}_c^{\min} y_c \le x_c^C \le \mathrm{AV}_c^{\max} y_c$$

A.3. Constraints for material balance of crude inventory ($|C|$)

Material balances for the inventory of crude $c \in C$ are defined by initial inventory level, the crude purchasing amount and the charging flowrate from storage tank to the distillation unit:

$$p_c = p_{c,0} + x_c^C - x_{D,c}^I$$

A.4. Constraints for material balance of intermediate products ($12+3|C|$)

Material balances for the intermediate product $k \in K \backslash F$ are based on the generalized equation which is expressed as production rate from the distillation and other units minus the internal consumption for the units and the blending of the final products, where $y_{u,i_u,k}$ is the yield of product k from unit u using inlet material i_u, x_k^{IMP} is zero for all intermediate products except HN and VGO and x_k^{EXP} is zero for all intermediate products except LG:

$$\sum_{u \ne \mathrm{RF}} \sum_{i_u} y_{u,i_u,k} x_{u,i_u}^I + x_k^{IMP} - \sum_{u \ne \mathrm{D}} x_{u,k}^I - \sum_f x_{f,k}^B = x_k^{EXP}$$

A.5. Constraints for material balance of final products (5)

Material balances for the final product $f \in F$ are based on the following generalized blending equation with the products in B_{fp} and imported products:

$$\sum_{b_f} x_{f,b_f}^B + x_f^{IMP} = x_f^{EXP}$$

A.6. Constraints for material balance of refinery fuel (1)

Refinery fuel balance is formulated as a similar way: energy is produced from burning the fuel (HF, LN, LG, RG) and fuel consumptions by the units are set to be proportional to the feedstocks while those for the generation of electricity and steam are assumed constant:

$$\sum_{i_{\mathrm{RF}}} x_{\mathrm{RF},i_{\mathrm{RF}}}^I U_{i_{\mathrm{RF}}} = \sum_{u \notin \mathrm{RF}} \sum_{i_u} x_{u,i_u}^I W_u + C_{\mathrm{RF}}$$

where U_k is the calorific value of product k as refinery fuel, W_u is the fuel consumption coefficient of unit u, and C_{RF} is constant refinery fuel consumption for electricity and steam (kton). According to Favennec (2001), the calorific values of RG, LG, and LN are 1.3, 1.2, and 1.1 times, respectively, that of HF.

A.7. Constraints for volume calculation of final products (4)

The volumes of the final products are calculated using the density of each blending material, ρ.

$$\sum_{b_f} \frac{x_{f,b_f}^B}{\rho_{b_f}} + \frac{x_f^{IMP}}{\rho_f} = v_f$$

A.8. Constraints for quality specification of final products (13)

Quality of a blend made up of different components is given by the blending rule, and the minimum or maximum required quality of product to be sold (or demand) should be satisfied:

$$Q = \frac{\sum q_i X_i}{\sum X_i} \begin{matrix} \le \\ \ge \end{matrix} q_f$$

where Q is the quality (e.g. impurity or other characteristics) of the blend obtained, X_i is the (volumetric or weight-based) quantity of each component in the blend, and q_i is the quality of each blend component. The following qualities of final products should be satisfied: butane (C_4) content, vapor pressure (RVP), sensitivity, and octane number (RON) of gasoline (PG98, ES95), sulphur content of DSL, and viscosity blending index of HF.

A.9. Constraints for unit capacity (4)

Capacity restriction of the processing unit $u \in$ {D, R, Cr, Des} is required, where CP_u is the maximum capacity of unit u. The capacity of reformer and cracker is limited to the sum of the two operating modes.

$$\sum_{i_u} x_{u,i_u}^I \le \mathrm{CP}_u$$

A.10. Objective function

The objective function is chosen to maximize the overall margin, which is defined as the profit from product exports minus the purchase price of crudes and material imports and the operating cost:

$$\max_{a,x} r(a,P) + q(x,P),$$

where

$$r(a,P) = -\sum_c P_c x_c^C, \quad q(x,P) = \sum_k \left(P_k x_k^{EXP} - P_k x_k^{IMP} \right) - \sum_u C_u \sum_{i_u} x_{u,i_u}^I.$$

where P_i is the price of product i (including crude, imported product, and exported product) and C_u is the operating cost for each unit. The unit operating cost of desulfurization is linearly increased with the sulphur content in its feedstock (GO or CGO) and differs from the refining crude.

Table B.1
Yield structure for distillation unit (D) by each crude oil (wt%).

	C1	C2	C3	C4	C5	C6	C7
RG	0.21	0.19	0.11	0.09	0.20	0.06	0.11
LG	1.88	1.70	1.01	0.83	1.83	0.56	0.95
LN	6.76	6.82	4.79	4.45	5.72	2.29	3.75
HN	17.94	17.62	14.53	13.89	13.03	9.47	12.29
KE	15.71	14.54	14.85	13.37	13.37	12.46	12.75
GO	18.56	16.90	18.14	17.08	16.14	18.91	16.30
VGO	27.50	28.15	30.03	31.32	29.34	35.16	30.24
VR	11.45	14.09	16.55	18.97	20.38	21.08	23.62

Crude assay source: http://corporate.exxonmobil.com/en/company/worldwide-operations/crude-oils/assays.

Table B.2
Yield for other processing units (wt%) (Favennec, 2001).

	R		Cr		Des		I
	Severity 95	*Severity 100*	*Mogas*	*AGO*	*feed: GO*	*feed: CGO*	
RG	8	9	1.5	1.2	2.5	4	3
LG	9	12	5.3	4.6	-	-	-
R95	83	0	–	-	-	-	-
R100	0	79	–	–	–	–	–
CN	–	–	43.6	38.1	–	–	–
CGO	–	–	44.6	51.1	–	–	–
DES-GO	–	–	–	–	97.5	0	–
DES-CGO	–	–	–	–	0	96	–
ISO	–	–	–	–	–	–	97

Reformer and cracker have two different operating modes, and desulfurization has two feedstocks.

Table B.3
The considered quality of final products and properties of intermediate (blended) products (refer to Favennec (2001) and crude assay).

		C_4 content (wt%)	Density (g/cm^3)	Vapor pressure (bar)	RON	Sensitivity	Sulfur (wt%)	VBI	Calorific value (t FOE)
Quality specification of the final product									
PG98		≤ 5	–	$0.5 \leq \leq 0.86$	≥ 98	≤ 10	–	–	–
ES95		≤ 5	–	$0.45 \leq \leq 0.86$	≥ 95	≤ 10	–	–	–
DSL		–	–	–	–	–	≤ 0.05	–	–
HF		–	–	–	–	–	–	$30 \leq \leq 33$	1 (by def.)
Properties of intermediate product									
RG		–	–	–	–	–	–	–	1.3
LG		–	0.58	4.30	94	4	–	–	1.2
LN		–	0.65	0.80	71	3	–	–	1.1
ISO		–	0.665	0.40	91	5	–	–	–
R95		–	0.77	0.50	95	9	–	–	–
R100		–	0.80	0.50	100	9	–	–	–
CN		–	0.75	0.65	93	11	–	–	–
KE		–	–	–	–	–	0.1	–	–
	C1	–	–	–	–	–	0.19	–	–
	C2	–	–	–	–	–	0.56	–	–
	C3	–	–	–	–	–	0.14	–	–
GO	C4	–	–	–	–	–	0.23	–	–
	C5	–	–	–	–	–	1.26	–	–
	C6	–	–	–	–	–	0.16	–	–
	C7	–	–	–	–	–	0.21	–	–
CGO		–	0.95	–	–	–	2	12	–
DES-GO		–	–	–	–	–	*$SLF_{GOc} \times 0.03$	–	–
DES-CGO		–	–	–	–	–	0.06	–	–
	C1	–	0.9885	–	–	–	–	35.16	–
	C2	–	0.9969	–	–	–	–	37.46	–
	C3	–	0.9896	–	–	–	–	39.70	–
VR	C4	–	0.9825	–	–	–	–	37.42	–
	C5	–	1.0244	–	–	–	–	40.58	–
	C6	–	0.9904	–	–	–	–	37.81	–
	C7	–	1.0183	–	–	–	–	42.90	–

* Sulfur content (wt%) in GO depending on the crude oil.

Appendix B. Required parameter and economic data for case study

The yields of all processing units are specified in Tables B.1 and B.2. The yield structure of the distillate unit depends on the type of crude oil, and those of other units are distinguished by the operating modes or feedstocks.

The considered properties of final products for quality specification, and the required properties of the intermediate products (specifically, blended materials for the final products) are appeared in Table B.3.

The maximum capacity, operating cost, and fuel consumption coefficient for each unit are set to be as Table B.4.

Table B.4
Model parameter related to the processing units (Favennec, 2001).

		Maximum capacity, CP_u (kton)	Operating cost, C_u ($/ton)	Fuel consumption coefficient, W_u
D		700	1	0.018
R	R100	60	2.7	0.019
	R95		3.2	0.026
Cr	Mogas	135	3	0.007
	AGO		3	0.007
I		–	0.6	0.04
Des	feed: GO	150	$1 + 0.685 \times SLF_{GO}$*	0.02
	feed: CGO		1.4	0.02

* sulfur content (wt%) in GO depending on the crude oil.

Table B.5
Discretization of price space in the MC model ($/ton).

	ξ_{min}	ξ_{max}	space size
Crude (WTI)	191	556	45.625
Gasoline (GSL)	257	653	49.500
Jet fuel (JF)	236	674	54.750
Diesel (DSL)	236	647	51.375

Table B.6
Economic data of all the import and export products.

	MC model components (i):			
	WTI	GSL	JF	DSL
Export products				
LG	1.005	–	–	–
PG98	–	1.005	–	–
ES95	–	0.995	–	–
JF	–	–	1	–
DSL	–	–	–	1
HF	–	–	0.45	–
Import products				
HN	1.180	–	–	–
VGO	1.092	–	–	–

In the Markov chain (MC) of daily price, each price variable is discretized into eight spaces based on the real data, and the minimum and maximum value and the space size of each price variable are specified as Table B.5.

The prices of other import or export products are set linearly proportional to the price of one of the four products modeled by MC: $P_k = \alpha_k \times P_i$, where $i \in$ {WTI, GSL, JF, DSL}; and the constant α_k is set as Table B.6.

References

Amaro, A., Barbosa-Póvoa, A.P.F., 2008. Planning and scheduling of industrial supply chains with reverse flows: a real pharmaceutical case study. Comput. Chem. Eng. 32 (11), 2606–2625.

Barro, D., Canestrelli, E., 2016. Combining stochastic programming and optimal control to decompose multistage stochastic optimization problems. OR Spectr. 38 (3), 711–742.

Bassett, M.H., Pekny, J.F., Reklaitis, G.V., 1996. Decomposition techniques for the solution of large-scale scheduling problems. AIChE J. 42 (12), 3373–3387.

Bengtsson, J., Nonås, S.-L., 2010. Refinery planning and scheduling: an overview. In: Energy, Natural Resources and Environmental Economics. Springer, pp. 115–130.

Birge, J.R., Louveaux, F., 2011. Introduction to Stochastic Programming. Springer Science & Business Media.

Bose, S., Pekny, J., 2000. A model predictive framework for planning and scheduling problems: a case study of consumer goods supply chain. Comput. Chem. Eng. 24 (2–7), 329–335.

Bradtke, S.J., Barto, A.G., 1996. Linear least-squares algorithms for temporal difference learning. Mach. Learn. 22 (1–3), 33–57.

Braun, M.W., Rivera, D.E., Flores, M., Carlyle, W.M., Kempf, K.G., 2003. A model predictive control framework for robust management of multi-product, multi-echelon demand networks. Annu. Rev. Control 27 (2), 229–245.

Chen, Y.-H., Lu, S.-Y., Chang, Y.-R., Lee, T.-T., Hu, M.-C., 2013. Economic analysis and optimal energy management models for microgrid systems: a case study in Taiwan. Appl. Energy 103, 145–154.

Cheng, L., Subrahmanian, E., Westerberg, A., 2003. Design and planning under uncertainty: issues on problem formulation and solution. Comput. Chem. Eng. 27 (6), 781–801.

Cheng, L., Subrahmanian, E., Westerberg, A.W., 2004. A comparison of optimal control and stochastic programming from a formulation and computation perspective. Comput. Chem. Eng. 29 (1), 149–164.

Dann, C., Neumann, G., Peters, J., 2014. Policy evaluation with temporal differences: a survey and comparison. J. Mach. Learn. Res. 15 (1), 809–883.

Dogan, M.E., Grossmann, I.E., 2006. A decomposition method for the simultaneous planning and scheduling of single-stage continuous multiproduct plants. Ind. Eng. Chem. Res. 45 (1), 299–315.

Dunn, S., Holloway, J., 2012. The pricing of crude oil. RBA Bull. 65–74.

Dupačová, J., Sladký, K., 2002. Comparison of multistage stochastic programs with recourse and stochastic dynamic programs with discrete time. ZAMM J. Appl. Math. Mech./Zeitschrift für Angewandte Mathematik und Mechanik 82 (11–12), 753–765.

Favennec, J., 2001. Petroleum refining V5. *Refinery operation and management*. Technip.

Fisher, M., Ramdas, K., Zheng, Y.-S., 2001. Ending inventory valuation in multiperiod production scheduling. Manag. Sci. 47 (5), 679–692.

Grossmann, I.E., 2005. Enterprise-wide optimization: a new frontier in process systems engineering. AIChE J. 51 (7), 1846–1857.

Grossmann, I.E., 2012. Advances in mathematical programming models for enterprise-wide optimization. Comput. Chem. Eng. 47, 2–18.

Grossmann, I.E., Guillén-Gosálbez, G., 2010. Scope for the application of mathematical programming techniques in the synthesis and planning of sustainable processes. Comput. Chem. Eng. 34 (9), 1365–1376.

Grunow, M., Günther, H.-O., Lehmann, M., 2002. Campaign planning for multi-stage batch processes in the chemical industry. OR Spectr. 24 (3), 281–314.

Hawkes, A., Leach, M., 2009. Modelling high level system design and unit commitment for a microgrid. Appl. Energy 86 (7), 1253–1265.

Honkomp, S., Mockus, L., Reklaitis, G., 1999. A framework for schedule evaluation with processing uncertainty. Comput. Chem. Eng. 23 (4–5), 595–609.

Konicz, A.K., Pisinger, D., Rasmussen, K.M., Steffensen, M., 2015. A combined stochastic programming and optimal control approach to personal finance and pensions. OR Spectr. 37 (3), 583–616.

Lee, J.H., 2014. Energy supply planning and supply chain optimization under uncertainty. J. Process Control 24 (2), 323–331.

Lin, X., Floudas, C.A., Modi, S., Juhasz, N.M., 2002. Continuous-time optimization approach for medium-range production scheduling of a multiproduct batch plant. Ind. Eng. Chem. Res. 41 (16), 3884–3906.

Maravelias, C.T., Sung, C., 2009. Integration of production planning and scheduling: overview, challenges and opportunities. Comput. Chem. Eng. 33 (12), 1919–1930.

McDonald, C., 1998. Synthesizing enterprise-wide optimization with global information technologies: harmony or discord. In: Pekny, J., Blau, G. (Eds.), Foundations of Computer Aided Process Operations, pp. 62–74.

McKAY, K.N., Safayeni, F.R., Buzacott, J.A., 1995. A review of hierarchical production planning and its applicability for modern manufacturing. Prod. Plann. Control 6 (5), 384–394.

Mestan, E., Türkay, M., Arkun, Y., 2006. Optimization of operations in supply chain systems using hybrid systems approach and model predictive control. Ind. Eng. Chem. Res. 45 (19), 6493–6503.

Papageorgiou, L.G., Pantelides, C.C., 1996. Optimal campaign planning/scheduling of multipurpose batch/semicontinuous plants. 1. Mathematical formulation. Ind. Eng. Chem. Res. 35 (2), 488–509.

Perea-Lopez, E., Ydstie, B.E., Grossmann, I.E., 2003. A model predictive control strategy for supply chain optimization. Comput. Chem. Eng. 27 (8), 1201–1218.

Powell, W.B., 2007. Approximate Dynamic Programming: Solving the Curses of Dimensionality, 703. John Wiley & Sons.

Powell, W.B., 2012. AI, OR and Control Theory: A Rosetta Stone for Stochastic Optimization. Princeton University.

Powell, W.B., 2014. Clearing the jungle of stochastic optimization. In: Bridging Data and Decisions. Informs, pp. 109–137.

Powell, W.B., Meisel, S., 2016. Tutorial on stochastic optimization in energy—Part I: modeling and policies. IEEE Trans. Power Syst. 31 (2), 1459–1467.

Pratikakis, N.E., 2009. Multistage Decisions and Risk in Markov Decision Processes: Towards Effective Approximate Dynamic Programming Architectures. Georgia Institute of Technology.

Puterman, M.L., 2014. Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons.
Ren, H., Gao, W., 2010. A MILP model for integrated plan and evaluation of distributed energy systems. Appl. Energy 87 (3), 1001–1014.
Sahinidis, N.V., 2004. Optimization under uncertainty: state-of-the-art and opportunities. Comput. Chem. Eng. 28 (6), 971–983.
Sand, G., Engell, S., 2004. Modeling and solving real-time scheduling problems by stochastic integer programming. Comput. Chem. Eng. 28 (6–7), 1087–1103.
Shobrys, D.E., White, D.C., 2002. Planning, scheduling and control systems: why cannot they work together. Comput. Chem. Eng. 26 (2), 149–160.
Stefansson, H., Shah, N., Jensson, P., 2006. Multiscale planning and scheduling in the secondary pharmaceutical industry. AIChE J. 52 (12), 4133–4149.
Sutton, R.S., Barto, A.G., 1998. Reinforcement Learning: An Introduction, 1. MIT press Cambridge.
van den Heever, S.A., Grossmann, I.E., 2003. A strategy for the integration of production planning and reactive scheduling in the optimization of a hydrogen supply network. Comput. Chem. Engineering 27 (12), 1813–1839.
Wang, H., 1986. Recursive estimation and time-series analysis. Acoust. Speech Signal Process. IEEE Trans. 34 (6) 1678-1678.
Wu, D., Ierapetritou, M., 2007. Hierarchical approach for production planning and scheduling under uncertainty. Chem. Eng. Process. 46 (11), 1129–1140.
Yan, H.-S., Xia, Q.-F., Zhu, M.-R., Liu, X.-L., Guo, Z.-M., 2003. Integrated production planning and scheduling on automobile assembly lines. IIE Trans. 35 (8), 711–725.