**Noname manuscript No.**
(will be inserted by the editor)

# Mixed-Integer Linear Programming for Scheduling Unconventional Oil Field Development

**Akhilesh Soni · Jeff Linderoth · James Luedtke · Fabian Rigterink**

Received: date / Accepted: date

**Abstract** The scheduling of drilling and hydraulic fracturing of wells in an unconventional oil field plays an important role in the profitability of the field. A key challenge arising in this problem is the requirement that neither drilling nor oil production can be done at wells within a specified neighborhood of a well being fractured. We propose a novel mixed-integer linear programming (MILP) formulation for determining a schedule for drilling and fracturing wells in an unconventional oil field. We also derive an alternative formulation which provides stronger relaxations. In order to apply the MILP model for scheduling large fields, we derive a rolling horizon approach that solves a sequence of coarse time-scale MILP instances to obtain a solution at the daily time scale. We benchmark our MILP-based rolling horizon approach against a baseline scheduling algorithm in which wells are developed in the order of their discounted production revenue. Our experiments on synthetically generated instances demonstrate that our MILP-based rolling horizon approach can improve profitability of a field by 4-6%.

## 1 Introduction

Shale oil production in the U.S. has risen from 0.5 million barrels per day to 6.5 million barrels per day between 2008 and 2018. The U.S. Energy Information Administration (EIA) estimates that in 2018, production from tight oil resources accounted to about 61% of total U.S. crude oil production (U.S. Energy Information Administration EIA , 2019). Shale formations are a part of unconventional reservoirs, meaning the wells need to be mechanically stimulated in order to produce oil and gas at economically feasible flow rates (Etherington and McDonald, 2004). The rapid rise of shale oil production can be attributed to innovative advancements in horizontal drilling, hydraulic fracturing, and other well stimulation technologies (Office of Fossil Energy, 2013). However, these advanced production technologies require high level of investments and thus the strategic development of an oil field is important for the profitability of such fields.

The life cycle of a typical shale oil well starts with drilling the vertical part of the well which is followed by horizontal drilling. The next step in the well development process is hydraulic

A. Soni, J. Linderoth, J. Luedtke
University of Wisconsin Madison, Department of Industrial and Systems Engineering
E-mail: soni6@wisc.edu, linderoth@wisc.edu, jim.luedtke@wisc.edu

F. Rigterink
ExxonMobil Upstream Integrated Solutions E-mail: fabian.rigterink@exxonmobil.com

fracturing (fracturing, for short) which consists of pumping a mixture of water, sand, and chemicals to stimulate the rock to allow oil and gas to escape through the rock. After fracturing is completed, a cleaning crew cleans out and turns the well online, after which the well begins producing oil.

We study the problem of scheduling drilling and fracturing of wells in the development of an unconventional oil field. While we focus on oil field development, our results may also be useful for planning shale gas field development, which follows a similar development process. The field can be represented by a collection of pads $P$ as shown in Figure 1. A pad $p \in P$ is a piece of land containing a number of wells. In this work, following common practice, we assume all wells in a pad are first drilled sequentially (by a single drilling crew) and then fractured sequentially (by a single fracturing crew). Thus, for scheduling purposes we treat all the wells on a pad $p$ as a single entity. Although we frame our work in terms of scheduling drilling and fracturing operations for the pads, our work can be directly applied to individual well scheduling by simply considering each well to be on its own pad. We propose a novel MILP formulation for determining a schedule for drilling and fracturing pads in an unconventional oil field which considers capacity, operational, precedence, and interference constraints. We also propose a formulation that uses more decision variables, but which provides a stronger linear programming relaxation. Our results show that this larger formulation can improve solution times by 25-70% on instances with relatively few time periods in the planning horizon, but is not advantageous for instances with more time periods. Due to the large problem size, solving the full MILP model for instances with many pads and a large number of time periods is intractable. Thus, we also derive a MILP-based rolling horizon framework that solves a sequence of limited horizon, coarser-scale MILP instances in a rolling forward fashion to obtain a solution to the full horizon problem on the daily time scale. We benchmark this approach against a baseline scheduling algorithm that approximates current practice, where pads are scheduled in the order of their discounted production revenue with limited lookahead to avoid conflicts. Our results show that our proposed MILP-based rolling horizon approach can improve net present value of a field by 4-6%.

There is significant literature on conventional oil and gas infrastructure planning and development, e.g., (B. Tarhan and Goel, 2009; Iyer et al, 1998; Lin and Christodoulos, 2003; Goel and Grossmann, 2004; Carvalho and Pinto, 2006). The literature on unconventional oil and gas planning and development is more limited. In recent years, some work has been done to determine an optimal structure of a shale gas network. Cafaro and Grossmann (2014) determined the most profitable supply chain design by using a branch-refine-optimize (BRO) strategy to solve a mixed-integer nonlinear programming (MINLP) formulation. Knudsen and Foss (2013) proposed a formulation to solve the scheduling of multi-well shut-ins. Arredondo-Ramirez et al (2016) proposed a method for determining a superstructure with potential wells, gas treatment plants, and distribution networks. Cafaro et al (2016) also presented a superstructure capturing the tree structure of gas gathering systems. They solved a nonconvex MINLP to consider spatial gas quality variations within multiple delivery node gathering systems. Another important aspect of field development is scheduling of different operations. Iyer et al (1998) discussed a discrete-time MILP model for conventional offshore oil field infrastructure development and used a decomposition approach to solve larger instances. Drouven and Grossmann (2016, 2017) introduced optimization frameworks to plan shale gas well refracture treatments of a single well under uncertainty. Rahmanifard and Plaksina (2018) optimized the well placement in a shale gas reservoir and compared the performance of different heuristics for maximizing well production.

Little attention has been paid in the literature towards the purpose of identifying a schedule for crews to perform the drilling and fracturing operations in the initial development phase. A key challenge in scheduling these operations is the presence of *conflicts* between different operations. A conflict refers to the restriction that when a pad is being fractured, it is not allowed to perform

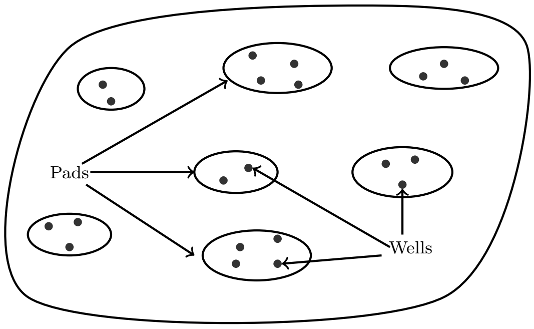

Fig. 1: Diagrammatic view of pads and wells in a shale oil field.

drilling or production on any pads within a specified neighborhood of that pad. There are very few papers in the literature that consider such conflicts. An important exception is Ondeck et al (2019), who provide an optimization framework for selecting and developing gas wells on a single pad. Their model considers conflicts between wells within the pad, in addition to other detailed considerations such as the possibility to curtail gas production and the expenses involved in mobilizing development resources. In contrast, we focus on the scheduling of development operations at the field level, using a model that incorporates less details of the individual well development, but which can be used to schedule development activities of multiple drilling and fracturing crews over the development of the entire unconventional field, with the objective to maximize net present value of the net revenues.

The scheduling problem is related to the flexible flow shop scheduling problem, which has been proven to be $NP$-Hard (Garey, 1979; Gupta, 1988; Xie and Wang, 2005). Flexible flow shop scheduling has been intensively studied in many industries (Lee and Loong, 2019). Gupta (1988) considered a two-stage scheduling problem in which a set of jobs is given, each of which has to undergo two processes in sequence. There are a set of identical resources available to do each of the two processes. The well development scheduling problem has a similar structure, where the wells are the jobs, the first process is drilling, the second process is fracturing, and the resources are the drilling and fracturing crews. Our model extends the flexible flow shop scheduling model by considering the conflicts between these processes (i.e., it is not allowed to drill or produce while fracturing a nearby well). Because of this additional important complication, methods for flexible flow shop scheduling cannot be directly applied to our problem.

The rolling horizon approach has been applied in a variety of applications where a MILP is solved over a smaller number of periods in successive iterations. Some recent work using the rolling horizon approach in various applications are (Marquant et al, 2015; Silvente et al, 2015; Sam et al, 2013; Spratt and Kozan, 2018). A unique feature our rolling horizon strategy is that in addition to solving a sequence of problems with a limited lookahead, the problems we solve have a coarser time-scale than the time-scale of the solution we produce.

This paper is organized as follows. We provide a detailed description of the problem, our MILP formulation, and its use within a rolling horizon framework in Section 2. We provide the alternative, larger MILP formulation in Section 3, and also discuss how the strength of the formulation can be obtained without adding additional variables by using a cutting plane algorithm. In Section 4, we present results of a computational study in which we compare the different MILP formulations, study the effects of period length and lookahead window in the rolling horizon approach, and quantify the value of our MILP-based rolling horizon approach by benchmarking against a baseline scheduling algorithm.

## 2 Problem Description and Solution Approach

### 2.1 Problem Statement

We consider an oil field that has a collection of pads $P$ to be developed as shown in Figure 1. A pad $p \in P$ is a piece of land containing a number of wells. The first step in developing a pad $p \in P$ is drilling, which takes $\hat{\tau}_p^d$ days. The second step is fracturing which takes $\hat{\tau}_p^f$ days. Note that we use ( $\hat{}$ ) to denote the time in days, whereas in section 2.3 we will use, e.g., $\tau_p^d$ to represent the approximate number of *periods* to drill pad $p$ for a given coarser time discretization. Drilling and fracturing operations have a fixed cost $c_p^d$ and $c_p^f$ associated with them, which are assumed to be charged at the beginning of the operation. The final steps in the development of a pad are cleaning and turning in line operations, but as these are typically done after fracturing without delay and cause no conflicts with other operations, we do not consider them in our model.

A fixed number of drilling ($n^d$) and fracturing ($n^f$) crews are available, so that at any point in time at most $n^d$ pads can be in the process of drilling and at most $n^f$ pads can be in the process of fracturing. At most one drilling crew or fracturing crew can be assigned to a pad. Moreover, each pad $p$ is drilled and fractured in a single visit of drilling and fracturing crew respectively, without interruption.

A pad $p$ starts producing oil after fracturing is complete. The amount of oil produced from a pad in a period depends on the amount of time since the pad began fracturing, and is specified by a production curve $\hat{\alpha}_p$, where $\hat{\alpha}_{pk}$ represents the amount of oil production from pad $p$ during day $k$ after fracturing was started. Since production cannot occur while a pad is being fractured and cleaned the production curve is zero for the initial periods until fracturing and cleaning are complete, then increases to the pad's actual initial production level, and typically decreases over time after that. The production curve of a pad is obtained by summing the production curves of the individual wells on the pad.

Since fracturing consists of pumping pressurized liquid into the well, as a safety and production protection measure, when fracturing occurs at a well on a pad, drilling and production at neighboring pads is stopped to prevent damage to equipment. For each pad $p \in P$ the set $N_p \subseteq P$ represents the neighboring pads for which production and drilling are prohibited when pad $p$ is being fractured. Although it is not necessary for our model, we assume the neighborhoods are symmetric so that $p \in N_q$ if and only if $q \in N_p$ for $p, q \in P$. If a pad that has completed fracturing is shut down due to fracturing at a neighboring well, we assume that the production profile of the pad still progresses to the next time period. Since the production rate curves are decreasing, when production resumes it will be at a lower rate. We let $P^{oil}$ denote the net revenue (price less processing costs) per barrel of oil and assume it is known and fixed for the entire time horizon of the field development process. The problem is to determine the drilling and fracturing start time of each pad in the field in order to maximize the net present value (NPV) of net revenues obtained from the field over its production horizon, where NPV is calculated using an annual discount rate of $i^A$. We let $i^D = \sqrt[365]{i^A + 1} - 1$ denote the equivalent daily discount rate.

*Illustrative Example* We consider a field consisting of 20 pads with three drilling crews and one fracturing crew. The pads are distributed on a rectangular grid with three rows and seven columns as shown in Figure 2. In this example, the neighbors of a pad $p$ consist of the pads lying immediate adjacent to $p$ horizontally or vertically. The drilling duration of pads is between 40 and 210 days and the fracturing duration is between 14 and 42 days. A sample schedule for this instance is shown in Figure 3. Day 0 in this schedule is set as Jan 1, 2019. The schedule demonstrated here is generated using a baseline scheduling algorithm (Algorithm 2) which is discussed in Section 4.3. We see that on every day, there are no more than three drilling and

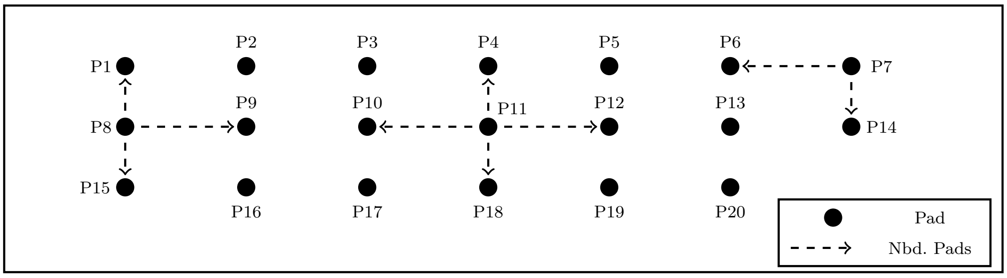

Fig. 2: Layout of pads in the illustrative example. The neighbors of pads P8, P11, and P7 are illustrated with dashed arrows.

one fracturing operations happening in parallel. The order of operations on each pad is drilling, followed by fracturing, and then production. We also observe that fracturing a pad leads to a shut down in the production of neighboring pads (these show up as gaps in the production bars). For instance, production on P2 is halted when P1 is fractured. Another observation is that sometimes a crew may need to idle due to interference constraints. For instance, a drilling crew becomes free after drilling at P2 is completed. However, drilling at P1 isn't initiated right away. P2 is first fractured after which drilling at P1 begins. This is because P1 and P2 lie in the same neighborhood and hence they cannot be drilled and fractured simultaneously. Schedules which have higher NPV tend to limit the number of production shutdowns due to fracturing conflicts, limit idling of resources, and begin production of high volume wells earlier.

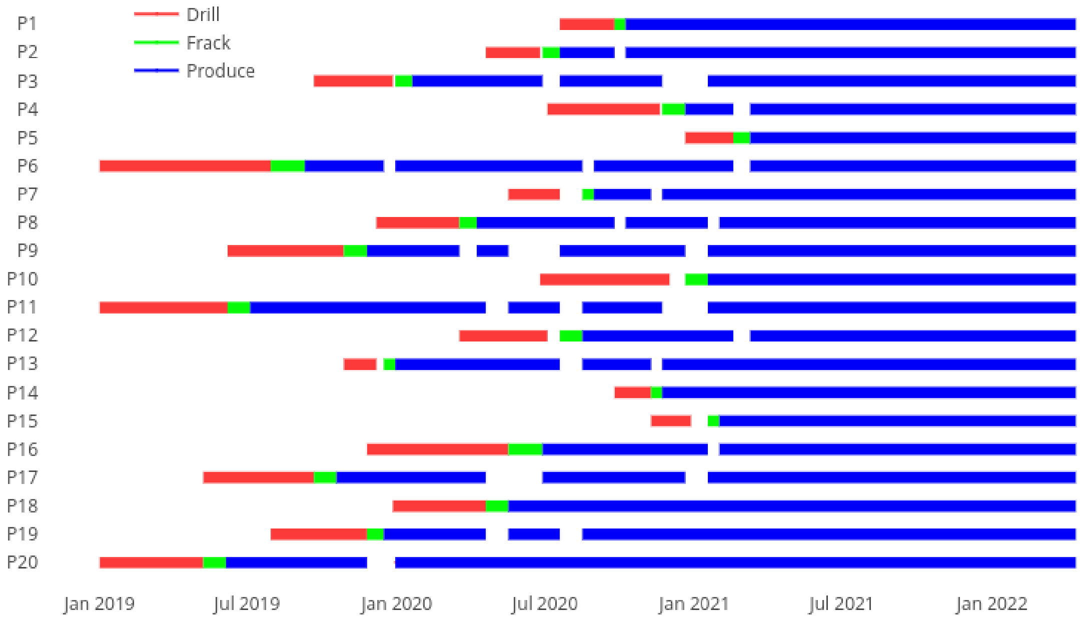

Fig. 3: Sample schedule for the illustrative example, obtained from the baseline scheduling algorithm. The NPV of this schedule is \$5.499*10$^9$.

### 2.2 Coarse-Time Approximation

We formulate the pad drilling and fracturing scheduling problem as a MILP problem using a discrete-time model consisting of a set of time periods $T := \{0, 1, \ldots, |T|\}$. In order to obtain a more compact model, we assume a period consists of $D$ days. For a pad $p$ which takes $\hat{\tau}_p^d$ days to drill, we approximate its drilling time in periods, $\tau_p^d$, by rounding $\hat{\tau}_p^d/D$ to the nearest integer. Similarly, the fracturing duration in periods, $\tau_p^f$, is approximated by rounding $\hat{\tau}_p^f/D$ to the nearest integer. The parameter $D$ provides a trade-off in model accuracy and complexity. A larger value of $D$ leads to a problem with a shorter time horizon which is hence more compact, but also leads to more inaccuracy due to rounding.

Given the annual discount rate $i^A$, the periodic discount rate $i$ is given by the formula

$$i = \sqrt[N]{i^A + 1} - 1,$$

where $N$ represent the number of periods in a year.

For a pad $p \in P$, the amount of oil produced in the $t^{\text{th}}$ period after fracturing was complete is computed as

$$\alpha_{pt} = \sum_{k=tD}^{(t+1)D-1} \hat{\alpha}_{pk}. \tag{1}$$

s If a pad begins production during the planning horizon, then our model needs to account for all production of the pad from the end of the planning horizon until the pad no longer produces. To do so, for each pad $p \in P$ and $0 \le t \le |T| - \tau_p^f$, we define $\beta_{pt}$ to be the discounted total revenue from oil produced beyond the planning horizon if fracturing of pad $p$ begins at time period $t$, where the revenue is discounted to period $|T|$. Specifically, $\beta_{pt}$ is computed as

$$\beta_{pt} = P^{oil} \sum_{k=|T|-(t+\tau_p^f)}^{\text{TMAX}_p} (1+i)^{-k} \alpha_{pk}. \tag{2}$$

The oil that is produced *within* the planning horizon is accounted for differently because of the possibility of production shut downs due to fracturing operations at neighboring pads, and this is why $\beta_{pt}$ must be calculated separately for each possible fracturing start time $t$.

The notation used in our problem definition is summarized in Table 1.

### 2.3 MILP Formulation

The decision variables in the MILP model are as follows:

- $x_{pt}$ : Binary variable that takes the value 1 if drilling at pad $p \in P$ starts at the beginning of time period $t \in T$, 0 otherwise.
- $y_{pt}$: Binary variable that takes the value 1 if fracturing at pad $p \in P$ starts at the beginning of time period $t \in T$, 0 otherwise.
- $\overline{x}_{pt}$: Binary variable that takes the value 1 if drilling for pad $p \in P$ has been completed by the beginning of time period $t \in T$, 0 otherwise.
- $\overline{y}_{pt}$: Binary variable that takes the value 1 if fracturing for pad $p \in P$ has been completed by the beginning of time period $t \in T$, 0 otherwise.
- $w_{pt}$: Binary variable that takes the value 1 if pad $p \in P$ is in production mode in period $t \in T$, 0 otherwise.
- $v_{pt}$: Amount of oil produced from pad $p \in P$ during period $t \in T$.

| Parameter | Description | Units |
|---|---|---|
| $P$ | Set of pads | - |
| $N_p$ | Neighboring pads of $p$ | - |
| $D$ | Length of a period | days |
| $\hat{\tau}^d$ | Drilling duration | days |
| $\tau^d$ | Rounded drilling duration | Periods |
| $\hat{\tau}^f$ | Fracturing duration | days |
| $\tau^f$ | Rounded fracturing duration | periods |
| $c^d$ | Drilling cost | \$ |
| $c^f$ | Fracturing cost | \$ |
| $n^d$ | Number of drilling crews | - |
| $n^f$ | Number of fracturing crews | - |
| $\hat{\alpha}_{pk}$ | Pad production on $k^{\text{th}}$ day since fracturing began | barrels |
| $\alpha_{pt}$ | Pad production on $t^{\text{th}}$ period since fracturing began | barrels |
| $i^A$ | Annual discount rate | 1/year |
| $i$ | Periodic discount rate | 1/period |
| $i^D$ | Daily discount rate | 1/day |
| $P^{oil}$ | Net revenue from a barrel of oil | \$/barrel |

Table 1: A summary of parameters

The objective is to maximize net present value (NPV) of net revenue:

$$NPV = \sum_{t\in T}\sum_{p\in P}\Big[(1+i)^{-t}\big(P^{oil}v_{pt} - c_p^d x_{pt} - c_p^f y_{pt}\big) + (1+i)^{-|T|}\beta_{pt}y_{pt}\Big]. \tag{3}$$

Note that the first term is discounted using the period discount rate $i$ since the expression $P^{oil}v_{pt} - c_p^d x_{pt} - c_p^f y_{pt}$ represents the net revenue in period $t$. The term $\beta_{pt}y_{pt}$ is discounted $|T|$ periods since the computation of $\beta_{pt}$ discounts the revenues of oil from beyond the planning horizon to time period $|T|$.

Next, we introduce the constraints in the model.

**Relationship Constraints:** The first set of constraints relates the decision variables for determining when drilling starts for a pad to the decision variables that indicate whether or not drilling has been completed:

$$\overline{x}_{pt} = 0, \qquad \forall p \in P, \quad t = 0, 1, \ldots, \tau_p^d - 1, \tag{4a}$$

$$\overline{x}_{pt} = \overline{x}_{p,t-1} + x_{p,t-\tau_p^d}, \qquad \forall p \in P, \ t = \tau_p^d, \tau_p^d + 1, .., |T|. \tag{4b}$$

Equations (4a) record the fact that for each pad $p \in P$ it is not possible to have completed drilling within the first $\tau_p^d - 1$ periods. Equations (4b) are equivalent to the equations

$$\overline{x}_{pt} = \sum_{k=0}^{t-\tau_p^d} x_{pk}, \quad \forall p \in P, \ t = \tau_p^d, \tau_p^d + 1, \ldots, |T|, \tag{5}$$

and thus correctly capture the relationship that drilling at a pad $p$ is complete if and only if drilling was started at time period $t - \tau_p^d$ or earlier. Note that we use (4b) in our formulation rather than (5) because the number of constraints is the same, and the set of constraints (4b) has significantly fewer nonzero coefficients. The constraints (4b) also imply the equation $\sum_{t=0}^{|T|} x_{pt} = \overline{x}_{p|T|} \leq 1$, which thus enforces the condition thatt each pad is drilled at most once. Observe that the model allows a pad $p$ to not be selected for drilling at all (i.e., $\overline{x}_{p|T|} = 0$). This is necessary because, as we discuss in Section 2.4, for large-scale instances the formulation will be used within a rolling horizon framework in which the problem is solved over a limited lookahead horizon. Due to the limited length of the lookahead horizon it may note be feasible to drill all the wells within the horizon.

A similar set of constraints relates the decision variables for determining when fracturing starts for a pad to the decision variables that indicate whether or not fracturing has been completed:

$$\overline{y}_{pt} = 0, \qquad \forall p \in P, \quad t = 0, 1, \dots, \tau_p^f - 1, \tag{6a}$$

$$\overline{y}_{pt} = \overline{y}_{p,t-1} + y_{p,t-\tau_p^f}, \qquad \forall p \in P, \quad t = \tau_p^f, \tau_p^f + 1, .., |T|. \tag{6b}$$

**Capacity Constraints:** The following constraints ensure that the number of pads being simultaneously drilled or fractured doesn't exceed the number of drilling or fracturing crews available at any period:

$$\sum_{p \in P} \sum_{k=(t-\tau_p^d+1)_+}^{t} x_{pk} \leq n^d, \qquad \forall t \in T, \tag{7}$$

$$\sum_{p \in P} \sum_{k=(t-\tau_p^f+1)_+}^{t} y_{pk} \leq n^f, \qquad \forall t \in T. \tag{8}$$

Here we use the notation $(z)_+ = \max\{0, z\}$ for any integer $z$. Note that a pad $p \in P$ is being drilled at time $t$ if it has begun drilling in one of the periods $\tau_p^d$ before $t$, thus the expression on the left-hand side of (7) computes the number of pads being drilled at time $t$, and similarly for fracturing in (8).

**Precedence Constraints:** We next consider constraints that enforce that drilling must be done before fracturing. Specifically, the following constraints ensure that if drilling for a pad $p \in P$ has not yet been completed by a time $t$, then fracturing cannot be completed by time $t + \tau_p^f$:

$$\overline{y}_{p,t+\tau_p^f} \leq \overline{x}_{pt}, \quad \forall p \in P, \ t = 0, 1, ..., |T| - \tau_p^f. \tag{9}$$

Similarly, production can occur only after a pad has completed fracturing. The following constraint therefore enforces that if fracturing is not yet complete on a pad $p \in P$ by time period $t$, then time period $t$ cannot be a production period:

$$w_{pt} \leq \overline{y}_{pt}, \qquad \forall p \in P, \ t \in T. \tag{10}$$

**Operational Constraints:** Since drilling must be done before fracturing, and a pad cannot be producing until fracturing is complete we add the following constraints that prohibit a pad from having more than one operation (drilling, fracturing, or producing) occur at any time period $t$:

$$w_{pt} + \sum_{k=t-\tau_p^d+1}^{t} x_{pk} + \sum_{k=t-\tau_p^f+1}^{t} y_{pk} \leq 1, \qquad \forall p \in P, \ t = \tau_p^d, \tau_p^d + 1, ..., |T|. \tag{11}$$

**Vicinity Constraints:** For each pad $p$ and time period $t$, if any pad $q \in N_p$ is being fractured at time $t$, then $p$ cannot be in the process of drilling during that period, nor can it be producing during that period:

$$w_{pt} + \sum_{k=(t-\tau_p^d+1)_+}^{t} x_{pk} \leq 1 - \sum_{k=(t-\tau_q^f+1)_+}^{t} y_{qk}, \qquad \forall p \in P, \ q \in N_p, \ t \in T. \tag{12}$$

**Oil Production Constraints:** The volume of oil that can be produced from pad $p$ in time period $t$ ($v_{pt}$) is bounded above based on the production curve and when fracturing of the pad began:

$$v_{pt} \leq \sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} y_{pk}, \qquad \forall p \in P, t = \tau_p^f, \tau_p^f + 1, .., |T|. \tag{13}$$

If $y_{pk} = 0$ for all $k \leq t - \tau_p^f$, then fracturing is not yet complete by time $t$ and hence (13) correctly records that no production can occur in period $t$. Otherwise, if $y_{pk} = 1$ for some $k \leq t - \tau_p^f$, then (13) bounds the production to not exceed $\alpha_{p,t-k}$, which is the limit in period $t$ since in this case fracturing began $t - k$ periods before period $t$.

In addition, the production amount from a pad must be zero if the pad is shut down due to fracturing at a neighboring pad. A shut down of pad $p$ in time period $t$ due to fracturing in a neighboring pad will cause $w_{pt} = 0$ due to constraints (12). Thus the following constraint then ensures the volume produced is zero in this case:

$$v_{pt} \leq \bar{\alpha}_p w_{pt}, \qquad \forall p \in P, \quad t \in T, \tag{14}$$

where $\bar{\alpha}_p$ is an upper bound on the maximum possible production from pad $p$ in a period (e.g., $\bar{\alpha}_p = \max\{\alpha_{pk} : k \geq 0\}$).

Note that when $w_{pt} = 1$ for a pad $p \in P$ in a time period $t \in T$, the production amount $v_{pt}$ will be exactly equal to the expression in the right-hand side of (13) due to the objective function. We must use inequality in the constraint (13) in order to allow $v_{pt} = 0$ in the case that $w_{pt} = 0$.

In summary, the MILP formulation is to maximize the objective (3), subject to the constraints (4), (6) - (14), and with binary restrictions on the decision variables $x_{pt}, y_{pt}, \overline{x}_{pt}, \overline{y}_{pt}, w_{pt}$ for $p \in P, t \in T$, and non-negativity on the oil production variables, $v_{pt} \geq 0$ for $p \in P, t \in T$. We refer to this formulation as MILP1.

### 2.4 Rolling Horizon Implementation

We next present a rolling horizon approach which is designed to obtain solutions for the problem for significantly larger instances. The basic idea with a rolling horizon framework is to solve a model over a limited planning horizon, fix the initial decisions, then move the window of the planning horizon forward in time and repeat. We let $\zeta = |T|D$ be the number of days in the planning horizon of the optimization model. In addition to limiting the planning horizon, we also use time periods of length $D$ days to limit the size of the MILP formulation being solved at each step. However, the rolling forward is done at the daily level, so that in the end the algorithm produces a schedule that is feasible to the problem using a daily time discretization. Figure 4 illustrates the basic idea of the approach.

The details of the MILP-based rolling horizon approach are given in Algorithm 1. The status of each pad is maintained throughout the algorithm, which is initialized as ‘idle’. The status of a pad is updated to ‘drilling’ when it is in the process of drilling, and changes to ‘drilled’ after drilling is complete. When fracturing begins the status is updated to ‘fracturing’ and finally it is updated to ‘fractured’ when that is complete, after which all operations for the pad are done. The number of available drilling and fracturing crews at the current day is updated in the variables $A^d$ and $A^f$, respectively. At the beginning of processing each day, we first check to see if there are any free drilling crews and pads that need to be drilled, or there are any free fracturing crews and pads that need to be fractured. If so, the limited horizon, aggregate time-period MILP model is solved (line 6). In this MILP, the decision variables for any drilling or fracturing operations currently in progress are fixed to require them to begin in the initial time period, and their durations are adjusted according to the remaining duration of these operations. After solving the MILP, the pads that are assigned to begin drilling in the first period of the model horizon are stored in $M_t^{\text{drill}}$, and likewise the pads that are assigned to begin fracturing in the first period of the MILP model are stored in $M_t^{\text{fracture}}$. Then, for each pad $p \in M_t^{\text{drill}}$ we first check to see if starting drilling on pad $p$ is feasible with respect to conflicts between pads currently being fractured in its neighborhood, and if it is feasible we update the pad status, store its start day in drillstart$[p]$, and update the number of available drilling crews (line 12). The conflict check is necessary because the MILP formulation uses aggregate time periods, and so could potentially miss conflicts when creating the schedule at the daily basis. A similar process is performed for assigning fracturing operations for each pad $p \in M_t^{\text{fracture}}$. At the end of processing each day, we determine whether any pads with status of 'drilling' or 'fracturing' will complete that process at the end of the day, and if so, update their status and the number of drilling or fracturing crews available. For pads that have status 'drilling' or 'fracturing' and which are not completing that day, we update the remaining time of these operations in terms of the number of periods. We ensure that drilling or fracturing duration is at least one period to prevent rounding down the duration to zero periods. Finally, we check the termination condition of the algorithm, which occurs when all pads have status 'fractured', indicating that all drilling and fracturing operations have been scheduled.

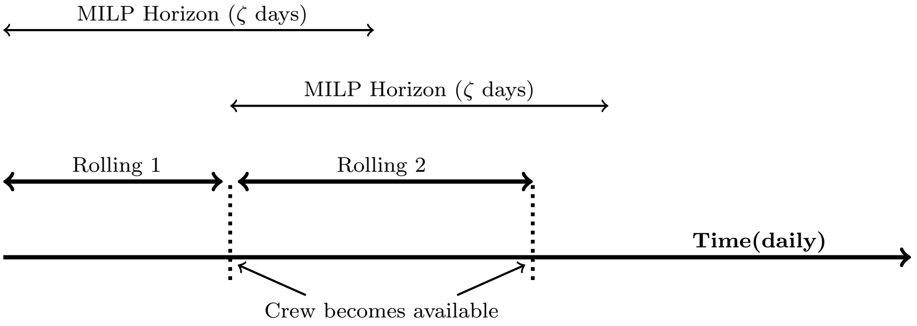

Fig. 4: Illustration of the rolling horizon approach. The limited horizon MILP model is re-solved whenever a crew becomes free, which is checked at the daily level.

## 3 Alternative MILP formulation

In this section we present an alternative MILP formulation and demonstrate that the LP relaxation of this formulation is at least as strong as the LP relaxation of MILP1 presented in Section 2.3. We refer to this alternative formulation as MILP-EF, as it can be considered to be an extended formulation since it uses more decision variables.

For each pad $p \in P$, we introduce a new set of binary decision variable $z_{pkt}$ for $\tau_p^f \leq t \leq |T|$ and for $k \leq t - \tau_p^f$, where $z_{pkt} = 1$ if pad $p$ begins fracturing in period $k$ *and* is producing in period $t$ ($w_{pt} = 1$), and $z_{pkt} = 0$ otherwise.

The first set of new constraints in the MILP-EF formulation relates the new $z_{pkt}$ variables to the $w_{pt}$ variables, enforcing the logic that if $w_{pt} = 0$ for a period $t$, then all the $z_{pkt}$ variables

**Algorithm 1** MILP based rolling horizon algorithm

```
status[$p$]$\leftarrow$ 'idle' for all $p \in P$;
$A^d \leftarrow n^d$, $A^f \leftarrow n^f$;
for $t = 0, 1, 2, \ldots$ do
  Determine if there is possibility to assign an operation to start in time $t$
  if ($A^d > 0$ and $\exists\ p$ s.t. status[$p$]='idle') or ($A^f > 0$ and $\exists\ p$ s.t. status[$p$]='drilled') then
    Solve the MILP model for next $|T|$ periods ($\zeta$ days);
    $M_t^{\text{drill}} \leftarrow$ set of pads $p \in P$ which MILP solution assigns to drill in period 0;
    $M_t^{\text{fracture}} \leftarrow$ set of pads $p \in P$ which MILP solution assigns to fracture in period 0;
    for $p \in M_t^{\text{drill}}$ do
      Check for interference
      if $\nexists q \in N_p$ s.t. status[q]='fracturing' and $A^d > 0$ then
        drillstart[$p$]$\leftarrow$ t, $A^d \leftarrow A^d - 1$, status[$p$]$\leftarrow$'drilling';
      end if
    end for
    for $p \in M_t^{\text{fracture}}$ do
      Check for interference
      if $\nexists q \in N_p$ s.t. status[$q$]='drilling' and $A^f > 0$ then
        fracstart[$p$]$\leftarrow t$, $A^f \leftarrow A^f - 1$, status[$p$]$\leftarrow$'fracturing';
      end if
    end for
  end if
  Update the completed operations and remaining duration for the pads under operation
  for $p \in P$ s.t. status[$p$]='drilling' do
    if $t$ =drillstart[$p$]+$\hat{\tau}_p^d$ then
      $A^d \leftarrow A^d + 1$, status[$p$]$\leftarrow$'drilled';
    else
      $\tau_p^d \leftarrow \max\Big(\text{round}\big(\frac{\hat{\tau}_p^d - (t - \text{drillstart}[p])}{D}\big), 1\Big)$;
    end if
  end for
  for $p \in P$ s.t. status[$p$]='fracturing' do
    if $t$ =fracstart[$p$]+$\hat{\tau}_p^f$ then
      $A^f \leftarrow A^f + 1$, status[$p$]$\leftarrow$'fractured';
    else
      $\tau_p^f \leftarrow \max\Big(\text{round}\big(\frac{\hat{\tau}_p^f - (t - \text{fracstart}[p])}{D}\big), 1\Big)$;
    end if
  end for
  Exit if all pads have been fractured and ready to produce
  if status[$p$]='fractured' for all $p \in P$ then
    break;
  end if
end for
```

for $k \leq t - \tau_p^f$ must also be zero:

$$\sum_{k=0}^{t-\tau_p^f} z_{pkt} \leq w_{pt}, \qquad \forall p \in P,\ t = \tau_p^f, \ldots, |T|. \tag{15}$$

This enforces the desired logic when $w_{pt} = 0$, and when $w_{pt} = 1$, this simply states the redundant constraint that fracturing for pad $p$ can be started at most once in periods 0 to $t - \tau_p^f$.

The next relationship constraint we add enforces the condition that if a pad $p$ does not have fracturing start in a period $k$ ($y_{pk} = 0$), then all of the $z_{pkt}$ variables for $t \geq k + \tau_p^f$ must be zero:

$$z_{pkt} \leq y_{pk}, \qquad \forall p \in P,\ k = 0, \ldots, t - \tau_p^f,\ t = \tau_p^f, \ldots, |T|. \tag{16}$$

Finally, we can replace the constraints that specify the upper bounds of volume produced in each period (inequalities (13) and (14)) with the following constraints:

$$\sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} z_{pkt} = v_{pt}, \qquad \forall p \in P, t \in T, \tag{17}$$

where by convention for $t < \tau_p^f$ the left-hand side sum is zero and thus for such $t$ the constraints simply enforce the condition that no production can occur until after fracturing is complete. For $t \geq \tau_p^f$, the production amount in period $t$ is set to zero when $w_{pt} = 0$, since in this case $z_{pkt} = 0$ for all $k \leq t$ by (15). On the other hand, when $w_{pt} = 1$, the variables $z_{pkt}$ will be equal to $y_{pk}$ for all $k \leq t - \tau_p^f$ in an optimal solution since this is allowed by (16) and because the objective is improved by increasing $v_{pt}$. Thus, if fracturing begins in some period $k' \leq t - \tau_p^f$, then we will have $z_{pk't} = y_{pk'} = 1$, and so the production amount in period $t$ will be set to $v_{pt} = \alpha_{p,t-k'}$, which is the correct amount given that fracturing began $k'$ periods earlier.

In summary, the new formulation MILP-EF maximizes the objective (3), subject to the constraints (4), (6) - (12), and (15) - (17).

**Lemma 1** *The LP relaxation upper bound of MILP-EF is not larger than the LP relaxation upper bound of formulation MILP1.*

*Proof* We show that if $(x, \overline{x}, y, \overline{y}, v, w, z)$ is a feasible solution to the LP relaxation of MILP-EF, then $(x, \overline{x}, y, \overline{y}, v, w)$ is a feasible solution to MILP1. This proves the claim since the objective functions in the two models are the same.

Thus, let $(x, \overline{x}, y, \overline{y}, v, w, z)$ be a feasible solution to the LP relaxation of MILP-EF. We only need to verify that this solution satisfies (13) and (14), since all other constraints in MILP1 are included in MILP-EF.

First, for each $p \in P$ and $t \geq \tau_p^f$ we obtain

$$v_{pt} = \sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} z_{pkt} \leq \sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} y_{pk}$$

where the equality follows from (17) and the inequality follows from (16), and hence the solution satisfies (13).

Next, for each $p \in P$ and $t \geq \tau_p^f$ we obtain

$$v_{pt} = \sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} z_{pkt} \leq \sum_{k=0}^{t-\tau_p^f} \bar{\alpha}_p z_{pkt} \leq \bar{\alpha}_p y_{pt}$$

where the first equality follows from (17), the first inequality follows because $\alpha_{p,t-k} \leq \bar{\alpha}_p$, and the second inequality follows from (15). For $t < \tau_p^f$ (17) implies $v_{pt} = 0$. Thus, in either case this verifies that the solution satisfies (14). □

### 3.1 Valid Inequalities in the Original Variable Space

The number of decision variables and constraints in MILP-EF grows quadratically with the number of time periods $|T|$, and thus for problems with many time periods it may be time-consuming to even solve the LP relaxation of this model. We thus discuss how the strength of this formulation can be obtained in the space of variables of the original model MILP1 by adding valid inequalities as cuts to the LP relaxation. One possible implementation of this would be to add these valid inequalities at the initial LP relaxation before starting the branch-and-bound process for solving MILP1.

*Cut Separating Linear Program.* Given a (partial) solution $(\hat{y}, \hat{v}, \hat{w})$ of the LP relaxation of MILP1, we can determine if there is a solution to the LP relaxation of formulation MILP-EF by solving a small linear program for each $p \in P$ to determine if there are values for the variables $z_{tkp}$ that satisfy constraints (15) - (17) for this pad $p$. Specifically, for each pad $p \in P$ we solve the linear program:

$$\min \sum_{k \in T} \gamma_t \tag{18a}$$

$$\text{s.t.} \sum_{k=0}^{t-\tau_p^f} \alpha_{p,t-k} z_{tk} + \gamma_t = \hat{v}_{pt}, \quad \forall t \in T, \tag{18b}$$

$$\sum_{k=0}^{t-\tau_p^f} z_{tk} \leq \hat{w}_{pt}, \quad \forall t = \tau_p^f, \dots, |T|, \tag{18c}$$

$$0 \leq z_{tk} \leq \hat{y}_{pt}, \quad \forall k = 0, \dots, t - \tau_p^f,\ t = \tau_p^f, \dots, |T|, \tag{18d}$$

$$\gamma_t \geq 0, \quad \forall t \in T. \tag{18e}$$

Given $\hat{w}_{pt}, \hat{y}_{pt}$, and $\hat{v}_{pt} \geq 0$, (18) is feasible because one can set all $z_{tk}$ variables equal zero and $\gamma_t = \hat{v}_{pt}$ for all $t \in T$. This LP is also trivially bounded, and hence it has an optimal solution. The optimal value of (18) is zero if and only if there exists values of $z_{ptk}$ for $k = 0, \dots, t - \tau_p^f,\ t = \tau_p^f, \dots, |T|$ that satisfy the pad $p$ constraints of MILP-EF. Thus, such an extended solution for pad $p$ exists if and only if the dual objective value of every feasible dual solution of (18) is less than or equal to zero.

Let $\pi$, $\rho$, and $\theta$ be the dual variables associated with constraints (18b), (18c), and (18d), respectively, and let $\hat{\pi}$, $\hat{\rho}$, $\hat{\theta}$ be an optimal dual solution. Observe that the dual objective is to maximize

$$\sum_{t \in T} \pi_t \hat{v}_{pt} + \sum_{t=\tau_p^f}^{|T|} \rho_t \hat{w}_{pt} + \sum_{t=\tau_p^f}^{|T|} \hat{y}_{pt} \sum_{k=0}^{t-\tau_p^f} \theta_{tk}.$$

Thus, if the optimal value of (18) is zero for every $p \in P$, then there exists values of the $z_{ptk}$ variables such that appending these values to the partial solution $(\hat{y}, \hat{v}, \hat{w})$ of the LP relaxation of MILP1 is feasible to MILP-EF. On the other hand, if the optimal value of (18) is positive for some $p \in P$, the following inequality is implied by the constraints of MILP-EF, and hence by correctness of that formulation, is valid for formulation MILP-1, i.e., it does not cut off any integer feasible solutions:

$$\sum_{t \in T} \hat{\pi}_t v_{pt} + \sum_{t=\tau_p^f}^{|T|} \hat{\rho}_t w_{pt} + \sum_{t=\tau_p^f}^{|T|} y_{pt} \sum_{k=0}^{t-\tau_p^f} \hat{\theta}_{tk} \leq 0. \tag{19}$$

Moreover, this inequality is violated by the current LP relaxation solution $(\hat{y}, \hat{v}, \hat{w})$. Thus, adding this inequality to the LP relaxation of MILP1 and then re-solving has the potential to improve the LP relaxation value. This process can be repeated in a simple cutting plane algorithm, in which in each iteration the LP relaxation of MILP1 with cuts added is solved, then given the solution the cut separating linear programs (18) are solved for each $p \in P$, and then cuts of the form (19) are added when violated. If the process continues until no more violated cuts are found, the resulting LP relaxation value will be equal to the LP relaxation value of MILP-EF.

*Cut Validation.* Validity of the inequality (19) requires that the dual solution $(\hat{\pi}, \hat{\rho}, \hat{\theta})$ used to construct it be a feasible dual solution. When solving a linear program in practice, the solution given may be slightly infeasible (within numerical tolerances). Using such a solution has the potential to lead to an invalid cut. To ensure validity of the cut (19), we propose to modify the (potentially infeasible) solution returned by the LP solver to make it feasible as follows.

The feasible region to the dual linear program of (18) is given by the inequalities

$$\begin{aligned} \alpha_{p,t-k}\pi_t + \rho_t + \theta_{tk} &\le 0, && \forall k = 0, \ldots, t - \tau_p^f,\ t = \tau_p^f, \ldots, |T|, && \text{(20a)} \\ \pi_t &\le 1, && \forall t \in T, && \text{(20b)} \\ \rho_t &\le 0, && \forall t = \tau_p^f, \ldots, |T|, && \text{(20c)} \\ \theta_{kt} &\le 0, && \forall k = 0, \ldots, t - \tau_p^f,\ t = \tau_p^f, \ldots, |T|. && \text{(20d)} \end{aligned}$$

Given an "approximately" feasible dual solution $(\hat{\pi}, \hat{\rho}, \hat{\theta})$, we propose to adjust it to a guaranteed feasible solution $(\overline{\pi}, \overline{\rho}, \overline{\theta})$ using the following formulae:

$$\begin{aligned} \overline{\pi}_t &= \min\{\hat{\pi}_t, 1\}, \quad \forall t \in T, \\ \overline{\rho}_t &= \min\{\hat{\rho}_t, 0\}, \quad \forall t = \tau_p^f, \ldots, |T|, \\ \overline{\theta}_{tk} &= \min\{0, -(\alpha_{p,t-k}\overline{\pi}_t + \overline{\rho}_t)\}, \qquad \forall k = 0, \ldots, t - \tau_p^f,\ t = \tau_p^f, \ldots, |T|. \end{aligned}$$

## 4 Computational Study

We next report results from a computational study in which we compare the performance of the two proposed MILP formulations, MILP1 and MILP-EF, investigate the effect of the period length and lookahead window parameters of the MILP-based rolling horizon approach, and compare the solutions obtained with the proposed MILP-based rolling horizon approach to those obtained by a baseline scheduling algorithm that mimics current practice.

### 4.1 Test Instances

We randomly generated test instances for our experiments using parameter ranges adopted from Cafaro et al (2016), Drouven and Grossmann (2016) , Cafaro et al (2018), and Ondeck et al (2019). Each instance consists of a set of pads which we assume lie on a regular grid of length $L$ and width $W$. We define pads to be neighbors of each other if they are immediately adjacent vertically or horizontally in the grid. The data for each pad is determined by first randomly choosing how many wells are on the pad, which we generate uniformly as an integer between one and six. We then randomly generate characteristics of each well on the pad and determine the pad data based on the wells. The characteristics of the wells are generated according to the distributions given in Table 2. For a pad $p \in P$, this yields a set of wells $\mathcal{W}_p$ that are on that pad. We then determine the pad parameters from the wells on the pad as follows: $\hat{\tau}_p^d = \sum_{w \in \mathcal{W}_p} \hat{\tau}_w^d$, $\hat{\tau}_p^f = \sum_{w \in \mathcal{W}_p} \hat{\tau}_w^f$, $\text{TMAX}_p = \max\{\text{TMAX}_w : w \in \mathcal{W}_p\}$, $c_p^d = \sum_{w \in \mathcal{W}_p} c_w^d$, $c_p^f = \sum_{w \in \mathcal{W}_p} c_w^f$. For each well $w$, we follow (Arps, 1945) and model the oil production rate function of the well using an exponential decline curve

$$\lambda_w(s) = M_w e^{-a_w s}, \tag{21}$$

where $s \geq 0$ represents the time (in days) from when production begins. The daily production curve parameters for each pad $p$ and day $t \geq 0$ are then computed as

$$\alpha_{pt} = \begin{cases} 0 & t < \hat{\tau}_p^f \\ \sum_{w \in \mathcal{W}_p} \int_{t-\hat{\tau}_p^f}^{t-\hat{\tau}_p^f+1} \lambda_w(s) ds & t \geq \hat{\tau}_p^f. \end{cases}$$

We use an annual discount rate of $i^A = 0.12$ and set $P^{oil} = \$60$ per barrel.

| Parameter | Description | Value | Units |
|---|---|---|---|
| $\hat{\tau}_w^d$ | Time to drill the well | Normal, $\mu = 30, \sigma = 10$ | days |
| $\text{TMAX}_w$ | Life of a well | 40 | years |
| $\hat{\tau}_w^f$ | Time to fracture the well | Normal, $\mu = 7, \sigma = 2$ | days |
| $c_w^d$ | Cost to drill the well | $1.5 + .003\hat{\tau}_p^d$ | MM$ |
| $c_w^f$ | Cost to fracture the well | $3.5 + .01\hat{\tau}_p^d$ | MM$ |
| $M_w$ | Initial production rate of well | Uniform$(400, 3300)$ | Barrels/day |
| $a_w$ | Decay rate constant for well | Uniform(.0003-0.0007) | 1/day |

Table 2: Well-specific instance parameters.

| Inst. | Pads | $\zeta$ (Periods) | $\zeta$ (Days) | Period len. | $n^d, n^f$ |
|---|---|---|---|---|---|
| I-10-67 | 10 | 67 | 1000 | 15 days | 3,1 |
| I-10-71 | 10 | 71 | 500 | 7 days | 5,2 |
| I-15-20 | 15 | 20 | 600 | 30 days | 3,1 |
| I-20-20 | 20 | 20 | 900 | 45 days | 3,1 |

Table 3: Characteristics of instancees used to compare MILP1 and MILP-EF.

## 4.2 Comparison of MILP Formulations

We first investigate the performance of formulations MILP1 and MILP-EF proposed in Sections 2.3 and 3, respectively. For each instance, we compare the computational time and nodes explored to reach optimality or a desired optimality gap. We used Gurobi 8.1.1 as the MILP solver. Additionally we set a time limit of 15000s (250 min) for these experiments. These experiments were performed on a 2.8 GHz Quad-Core Intel Core i7 with 16 GB RAM.

The characteristics of the test instances used in this study are presented in Table 3. For each instance size (row in Table 3) we generated five random instances with those characteristics. We solve instances I-10-67, I-10-71, and I-15-20 to the default optimality gap of Gurobi, 0.01%. For the larger instance I-20-20 we solve to optimality gap 1%.

We report the gap of LP relaxation from the optimal value (LP gap), solution time, and nodes explored in the tree to reach the specified optimality gap for the two formulations on each test instance in Table 4. We can see that on an average the number of nodes explored to reach a solution of desired optimal tolerance is fewer by a factor in the range of 1.2-3.6 when using MILP-EF as compared to MILP1. This is a consequence of the better LP relaxation of MILP-EF, which is also demonstrated by the smaller LP relaxation relaxation gap. When considering solution time, we observe that although we explore fewer nodes with MILP-EF in instances I-10-67 and I-10-71, the solution times using MILP1 are significantly smaller. On the other hand, the solution times are smaller for MILP-EF on instances I-15-20 and I-20-20. This also includes an instance I-20-20-C where MILP1 reached the time limit of 15000s while formulation MILP-EF was solved to the 1% optimality gap in 3805 seconds. The difference in solution time behavior can be explained by the size of the model, and in particular the number of periods. Instances I-10-67 and I-10-71 have 67 and 71 periods, respectively, and hence the formulation MILP-EF, which has size growing quadratically in the number of periods, gets very large. We conclude that for instances with a small number of time periods (e.g., $\leq 20$) MILP-EF formulation is preferred, but for instances with more time periods formulation MILP1 may be preferred.

Our primary interest in this work is to demonstrate the impact of the use of the MILP formulations within the rolling horizon framework for generating high-quality solutions on larger instances, which we investigate in the next two subsections. Thus, we did not computationally test the approach of using cutting planes to obtain the strength of formulation MILP-EF in the space of the formulation MILP1.

| **Inst.** | **LP gap(%)** | | **IP time (sec.)** | | **# Nodes** | |
|---|---|---|---|---|---|---|
| | MILP1 | MILP-EF | MILP1 | MILP-EF | MILP1 | MILP-EF |
| I-10-67-A | 3.72 | 2.66 | 571 | 708 | 11069 | 5112 |
| I-10-67-B | 3.98 | 2.94 | 415 | 980 | 8720 | 6039 |
| I-10-67-C | 3.35 | 2.43 | 416 | 548 | 6133 | 2295 |
| I-10-67-D | 4.60 | 3.32 | 1894 | 1704 | 39220 | 10508 |
| I-10-67-E | 3.26 | 2.21 | 157 | 393 | 2692 | 1909 |
| Avg. | 3.78 | 2.71 | 690.6 | 866.6 | 13566.8 | 5172.6 |
| I-10-71-A | 4.23 | 2.79 | 496 | 760 | 11446 | 6391 |
| I-10-71-B | 3.45 | 2.50 | 214 | 418 | 7910 | 4432 |
| I-10-71-C | 3.87 | 2.56 | 459 | 1152 | 12028 | 8497 |
| I-10-71-D | 3.37 | 2.39 | 433 | 1465 | 7706 | 9522 |
| I-10-71-E | 8.52 | 7.18 | 321 | 1234 | 10248 | 9771 |
| Avg. | 4.69 | 3.48 | 384.6 | 1005.8 | 9867.6 | 7722.6 |
| I-15-20-A | 3.61 | 2.63 | 135 | 83 | 18460 | 7244 |
| I-15-20-B | 3.16 | 1.66 | 138 | 84 | 22370 | 9774 |
| I-15-20-C | 2.86 | 1.99 | 57 | 38 | 15956 | 4795 |
| I-15-20-D | 4.38 | 3.23 | 164 | 87 | 32282 | 12616 |
| I-15-20-E | 3.72 | 2.47 | 119 | 144 | 29538 | 30647 |
| Avg. | 3.55 | 2.40 | 122.6 | 87.2 | 23721.2 | 13015.7 |
| I-20-20-A | 6.14 | 3.87 | 2294 | 742 | 276358 | 54933 |
| I-20-20-B | 5.40 | 3.12 | 326 | 101 | 39010 | 8472 |
| I-20-20-C | 6.24 | 4.16 | 15000 | 3805 | 1511503 | 346964 |
| I-20-20-D | 5.91 | 3.41 | 566 | 207 | 65531 | 20499 |
| I-20-20-E | 5.89 | 3.57 | 2395 | 1782 | 205097 | 158152 |
| Avg. | 5.92 | 3.63 | 4116.2 | 1327.4 | 419499.8 | 117804 |

Table 4: Comparison of formulations MILP1 and MILP-EF.

### 4.3 Baseline Scheduling Algorithm

We next turn our attention to the use of the MILP-based rolling horizon approach for generating solutions to instances that are too large to solve to optimality. To provide context for the quality of the solutions generated on these instances, we present the baseline scheduling algorithm, which to the best of our knowledge closely mimics current scheduling practice. The idea behind the baseline scheduling algorithm is to develop pads with the highest discounted revenue first, in order to obtain the revenue from the highest value pads earlier, which is beneficial due to the discounting used in the NPV calculation. Thus, we rank the pads on the basis of their discounted total revenue, which is computed in (2) as $\beta_{p,|T|-\tau_p^f}$. The operations are then scheduled by prioritizing the pads with highest discounted volume production first, while ensuring that we don't violate any precedence, capacity, operational and conflict constraints.

The details of the baseline scheduling algorithm are given in Algorithm 2. We create a drill-queue of the pads, ordered highest to lowest by discounted revenue. Pads are added to the fracture-queue (line 20) after their drilling operation is initiated. Note that the order in the two queues may be different as drilling operations may not always start in the preferred order due to possible delays due arising from conflicts. In the algorithm each pad is initialized with the 'idle' status. We update the status of each pad as it goes through different stages of the development cycle, i.e., {'idle', 'drilling', 'drilled', 'fracturing', 'fractured' }. The variables $A^d$ and $A^f$ keep track of the number of free drilling and fracturing crews available at each point in time. $A^d$ and $A^f$ are initialized with $n^d$ and $n^f$ as all crews are free at $t = 0$. Similarly, the variables $\mu^d[p]$ and $\mu^f[p]$ are used to keep track of the remaining drilling and fracturing duration of each pad $p$. These are initialized with the actual drilling and fracturing duration ($\hat{\tau}^d[p]$ and $\hat{\tau}^f[p]$). Algorithm 2 proceeds by considering each day in the planning horizon in sequence. For each day, we first assign fracturing operations (lines 6-14) and then drilling operations (lines 15-24) to start that day. Fracturing is prioritized over drilling in case there is a conflict between either starting drilling or fracturing on two neighboring pads. Fracturing is prioritized because once it is complete the pad can begin production and revenue is generated. Fracturing operations are initiated in preference order of the fracture-queue. Fracturing is initiated at the next pad in the queue if there is a free fracturing crew, drilling is complete, and initiating fracturing on the given pad doesn't violate any interference constraints (lines 11-13). If all these conditions hold true, fracturing is started and we store the fracturing start day for the pad in 'fracstart', reduce the number of fracturing crews available ($A^f$) by one, remove the pad from the fracture queue, and update the status of pad to 'fracturing' (line 12). If a pad is not assigned for fracturing, but a fracturing crew is still available, the next pad in the queue is considered, and so on until either the full queue has been checked or there are no available fracturing crews. Drilling operations are initiated in a similar way (lines 15-24). However, when initiating drilling operations we check additional conditions in order to avoid creating conflicts with upcoming fracturing operations (line 20). In particular, we check each neighboring pad $q$ to see if it is currently being fractured (status[$p$]='fracturing'), is ready to be fractured(status[$q$]='drilled'), or there is a possibility that it may begin fracturing before the drilling operation on pad $p$ under consideration would be completed (and hence starting drilling on this pad may delay fracturing pad $q$). To perform this check, we define the values 'earlieststart[$q$]' for $q \in P$ as follows:

$$\text{earlieststart}[q] = \begin{cases} 0, & \text{if status}[q] = \text{`fracturing' or status}[q] = \text{`drilled'}, \\ +\infty, & \text{if status}[p] = \text{`idle' or status}[q] = \text{`fractured'}, \\ \sum_{j=1}^{i} \mu^f[q_j]/n^f, & \text{if } q = q_i \in \text{fracture-queue} = [q_1, q_2, \ldots, q_k]. \end{cases}$$

At the end of processing each day, we determine whether any pads with status of 'drilling' or 'fracturing' will complete that process at the end of the day, and if so update their status and the number of drilling and fracturing crews available. For pads with status 'drilling' or 'fracturing' and which are not completing that day, we update the remaining time of these operations (lines 25-39).

### 4.4 Parameter Study for MILP+Rolling Horizon Approach

We next study the effect of the period length ($D$) and lookahead horizon ($\zeta$) parameters in the MILP-based rolling horizon framework. Intuitively, one would expect to obtain the best solutions

**Algorithm 2** Baseline Scheduling Algorithm

```
drill-queue← Ordered list of pads based on $\beta_{p|T|}$ ; fracture-queue=[ ]
status[$p$]← ‘idle’ for all $p \in P$;
$A^d \leftarrow n^d$, $A^f \leftarrow n^f$;
$\mu^d[p] \leftarrow \hat{\tau}^d[p]$, $\mu^f[p] \leftarrow \hat{\tau}^f[p]$ for all $p \in P$;
for t=0,1,2... do
    Assign fracturing operation
    for $p$ in fracture-queue do
        if $A^f = 0$ then
            break;
        end if
        if $\nexists q \in N_p$ s.t. status[$q$]=‘drilling’ then
            fracstart[$p$] $\leftarrow t$, $A^f \leftarrow A^f - 1$, fracture-queue.remove($p$), status[$p$]← ‘fracturing’;
        end if
    end for
    Assign drilling operation
    for $p$ in drill-queue do
        if $A^d = 0$ then
            break;
        end if
        if $\nexists q \in N_p$ s.t. earliestfrac[$q$]< $\hat{\tau}^d_p$ then
            drillstart[$p$] $\leftarrow t$, $A^d \leftarrow A^d - 1$, drill-queue.remove($p$), status[$p$]←‘drilling’;
            fracture-queue.append($p$);
        end if
    end for
    Update the completed operations and remaining duration for the pads under operation
    for $p$ in $P$ s.t. status[$p$]=‘drilling’ do
        if drillstart[$p$]+$\hat{\tau}^d_p = t$ then
            $A^d \leftarrow A^d + 1$, status[$p$]←‘drilled’;
        else
            $\mu^d[p] \leftarrow \mu^d[p] - 1$;
        end if
    end for
    for $p$ in $P$ s.t. status[$p$]=‘fracturing’ do
        if fracstart[$p$]+$\hat{\tau}^f_p = t$ then
            $A^f \leftarrow A^f + 1$, status[$p$]←‘fractured’;
        else
            $\mu^f[p] \leftarrow \mu^f[p] - 1$;
        end if
    end for
end for
```

using a very large lookahead horizon $\zeta$ and small period length (daily). However, since the MILP becomes larger and more difficult to solve as the number of periods increases, it is necessary to impose a time limit when solving the MILP, and thus use the best solution found within the time limit. If the number of periods is too large, then the solution obtained within the time limit may be significantly suboptimal (or maybe even no solution could be found) leading to poor performance. Thus, we conduct a study to determine values of $D$ and $\zeta$ that lead to the best solutions.

The number of periods used in the instances in this study is relatively large (usually more than 20), and hence following the guidelines in Section 4.2 we use formulation MILP1 to solve the instances in this study. We set an optimality gap tolerance of 2% and a time limit of 1000s for each solve of the limited-horizon MILP model and use the best solution obtained by the solver when it terminates. As before, we used Gurobi 8.1.1 as the MILP solver. These experiments were obtained on a machine having two 3.2 GHz Intel Xeon CPUs with 4GB RAM. We set the number of threads in Gurobi to four.

For this study, we consider two different instances of moderate size given in Tables 5 and 6. When varying the lookahead horizon parameter $\zeta$, we consider different possibilities based on the fraction of the time it would take to develop all the pads in the instance. We refer to our estimate of this time as $\gamma$, which is calculated as:

$$\gamma = 1.3 * [\frac{\sum_{p \in P} \hat{\tau}_p^d}{n^d} + \frac{\sum_{p \in P} \hat{\tau}_p^f}{n^f}].$$

Here, the factor 1.3 is used to account for possible idling of drilling or fracturing crews due to conflict constraints. Note that, although instance I55 has only slightly more pads to be developed than I454, the estimated total planning horizon $\gamma$ is significantly longer because in I55 there are fewer drilling and fracturing crews.

The results of these experiments are reported in Tables 7 and 8. In our case study we use $D = 15, 30,$ and 45 days for period length and use lookahead amounts of 20%, 40%, 60%, and 80% of the entire planning horizon ($\gamma$). For each combination of period length ($D$) and lookahead horizon ($\zeta$), we report the percent improvement in NPV of the rolling horizon solution over the solution obtained by the baseline scheduling algorithm (negative percentage indicates the solution was worse than that obtained from the baseline scheduling algorithm). We also report the fraction of the limited-horizon MILP instances for which the time limit was reached before the desired optimality gap of 2% was reached (Time-lim. frac.) and the average ending optimality gap of the limited-horizon MILP instances (Mean opt. gap).

| **Parameter** | **Value** |
|---|---|
| Pads | 45 |
| Crews(Drill, Fracture) | 6,2 |
| Length of Horizon ($\gamma$) | 1350 days |

Table 5: Instance I45.

| **Parameter** | **Value** |
|---|---|
| Pads | 55 |
| Crews(Drill, Fracture) | 3,1 |
| Length of Horizon ($\gamma$) | 2700 days |

Table 6: Instance I55.

| **Period len. ($D$)** | **Lookahead horizon($\zeta$)** | $\|T\|$ | **Imprv.(%)** | **Time-lim. frac.** | **Mean opt. gap(%)** |
|---:|---:|---:|---:|---:|---:|
| 15 | 0.2 $\gamma$ | 18 | 3.35 | 0.00 | 1.17 |
| 15 | 0.4 $\gamma$ | 36 | 4.04 | 0.34 | 1.80 |
| 15 | 0.6 $\gamma$ | 54 | 3.59 | 0.36 | 2.63 |
| 15 | 0.8 $\gamma$ | 72 | 3.73 | 0.39 | 2.25 |
| 30 | 0.2 $\gamma$ | 9 | 3.13 | 0.00 | 1.05 |
| 30 | 0.4 $\gamma$ | 18 | 3.41 | 0.00 | 1.59 |
| 30 | 0.6 $\gamma$ | 27 | 3.77 | 0.28 | 1.81 |
| 30 | 0.8 $\gamma$ | 36 | 4.04 | 0.27 | 1.85 |
| 45 | 0.2 $\gamma$ | 6 | -1.80 | 0.00 | 1.18 |
| 45 | 0.4 $\gamma$ | 12 | 1.14 | 0.00 | 1.65 |
| 45 | 0.6 $\gamma$ | 18 | 1.16 | 0.38 | 2.14 |
| 45 | 0.8 $\gamma$ | 24 | 1.80 | 0.42 | 2.28 |

Table 7: Results of MILP-based rolling horizon algorithm on instance I45. NPV of baseline scheduling algorithm solution=$1.4293 * 10^{10}$.

| **Period len. ($D$)** | **Lookahead horizon ($\zeta$)** | $\lvert T\rvert$ | **Imprv.(%)** | **Time-lim. frac.** | **Mean opt. gap(%)** |
|---|---|---|---|---|---|
| 15 | 0.2 $\gamma$ | 36 | 5.21 | 0.00 | 1.48 |
| 15 | 0.4 $\gamma$ | 72 | 5.30 | 0.16 | 1.79 |
| 15 | 0.6 $\gamma$ | 108 | 5.24 | 0.45 | 2.29 |
| 15 | 0.8 $\gamma$ | 144 | 5.19 | 0.44 | 18.13 |
| 30 | 0.2 $\gamma$ | 18 | 2.78 | 0.00 | 1.47 |
| 30 | 0.4 $\gamma$ | 36 | 4.97 | 0.03 | 1.68 |
| 30 | 0.6 $\gamma$ | 54 | 5.10 | 0.30 | 1.81 |
| 30 | 0.8 $\gamma$ | 72 | 5.22 | 0.29 | 1.95 |
| 45 | 0.2 $\gamma$ | 12 | 2.61 | 0.00 | 0.90 |
| 45 | 0.4 $\gamma$ | 24 | 3.56 | 0.00 | 1.63 |
| 45 | 0.6 $\gamma$ | 36 | 3.15 | 0.34 | 1.89 |
| 45 | 0.8 $\gamma$ | 48 | 3.54 | 0.37 | 2.20 |

Table 8: Results of MILP-based rolling horizon algorithm on instance I55. NPV of baseline scheduling algorithm solution=$\$1.7721 * 10^{10}$.

The results in Tables 7 and 8 confirm that using a longer lookahead horizon usually yields better solutions, and that this holds for each period length $D$. On the other hand, while there is generally a significant improvement when lookahead is increased from $0.2\gamma$ to $0.4\gamma$, there is little to no improvement obtained increasing beyond that. In both instances, for the shortest period length, $D = 15$, we observed that the quality of the rolling horizon solution was actually smaller for the larger lookahead horizons. This can be explained by looking at the fraction of instances that were terminated due to the time limit and the mean optimality gap of the limited-horizon MILP problems in these instances. In particular, a significant fraction of these problems were terminated due to the time limit, and their average optimality gap was relatively large, indicating that the solutions obtained may have been significantly suboptimal.

Next we discuss the effect of period length on the quality of the solutions obtained. We observe from Tables 7 and 8 that when the period length $D$ is 45 days, the quality of the solutions obtained is significantly lower than when using $D = 15$ or $D = 30$. Using $D = 15$ tends to provide the best solutions, although the difference in the solutions obtained using $D = 15$ and $D = 30$ is small provided the lookahead horizon is $0.4\gamma$ or larger.

Thus, in these experiments we found that a lookahead window of $0.4\gamma$ and period length of either $D = 15$ or $D = 30$ yields the best results overall. We emphasize, however, that because the solution of the MILP problems was stopped after a time limit, the values of these parameters that yield the best solutions are dependent on the time limit used, as well as the MILP solver and computational environment used. In addition, when using the MILP-based rolling horizon algorithm in practice, it is only necessary to solve a single limited-horizon MILP instance whenever a drilling or fracturing crew completes an operation and becomes available. Given that in real time there may be several weeks between such events, a significantly longer time limit could potentially be used for solving these instances, thus enabling use of a longer lookahead window and smaller period length.

### 4.5 Large-Scale Instances

We finally test our rolling horizon framework on larger-scale instances using the parameters $D = 15$ and $\zeta = 0.4\gamma$, as suggested in Section 4.4. The instances we use for this study are described in Table 9. We report the baseline scheduling algorithm NPV and the improvement obtained by the MILP-based rolling horizon algorithm in Table 10. As in Tables 7 and 8, we also report the fraction of MILP instances for which the time limit was reached and the mean

| Instance | Pads | $n^d, n^f$ | $\gamma$ (days) |
|---|---|---|---|
| I-80-160 | 80 | 6,2 | 2400 |
| I-90-100 | 90 | 9,3 | 1500 |
| I-100-170 | 100 | 6,2 | 2550 |
| I-110-180 | 110 | 6,2 | 2700 |

Table 9: Large-scale instance data.

| Instance | Baseline NPV (\$) | Roll. Imprv.(%) | Time-lim. frac. | Mean opt. gap(%) |
|---|---|---|---|---|
| I-80-160 | $2.7339 * 10^{10}$ | 5.13 | 0.47 | 2.45 |
| I-90-100 | $3.0461 * 10^{10}$ | 5.01 | 0.19 | 1.79 |
| I-100-170 | $3.3928 * 10^{10}$ | 4.47 | 0.59 | 2.70 |
| I-110-180 | $3.7399 * 10^{10}$ | 6.54 | 0.59 | 8.38 |

Table 10: Improvement using MILP-based rolling horizon framework.

optimality gaps of the MILP instances. The results in Table 10 demonstrate that the solutions obtained using our MILP-based rolling horizon algorithm consistently have significantly higher NPV than those obtained with the baseline scheduling algorithm.

## 5 Conclusion and Future Work

In this paper we presented a novel MILP-based rolling horizon algorithm to schedule drilling and fracturing operations in an unconventional oil field development while considering the interaction effects between various pads. We provided two MILP formulations for the limited horizon MILP solved as part of this approach. The second formulation provides better LP relaxation bounds, which translates to shorter solution times for instances with a small number of time periods, but for larger instances the first formulation was found to be more effective. A key feature of the rolling horizon approach we propose is that it yields a solution at the daily time-scale, while solving a sequence of coarser time-scale MILP problems. An empirical study demonstrated that the approach can be used to plan development of fields with more than 100 pads, and the solutions obtained have 4-6% higher NPV than solutions obtained with a baseline scheduling algorithm that mimics current practice.

Our work assumed all data, including the drilling and fracturing durations, are deterministic. In reality, these are estimated via forecasts that may have significant errors. The rolling horizon framework we propose can naturally be applied in this setting, by using the updated state of the system, and updated estimates of the durations, whenever a new limited-horizon MILP is solved. However, an interesting direction for future work is to investigate the use of a stochastic or robust optimization formulation of the scheduling problem within the rolling horizon framework to see if this may yield improved solutions.

## References

Arps JJ (1945) Analysis of decline curves. Transactions of the AIME 160(1)

Arredondo-Ramirez K, Ponce-Ortega J, El-Halwagi M (2016) Optimal planning and infrastructure development for shale gas production. Energy Conversion and Management 119:91–100

B Tarhan IG, Goel V (2009) Stochastic programming approach for the planning of offshore oil or gas field infrastructure under decision-dependent uncertainty. Industrial and Engineering Chemistry Research 48(6):3078–3097

Cafaro C, Grossmann I (2014) Strategic planning, design, and development of the shale gassupply chain network. AIChE Journal 60:2122–2142

Cafaro C, Drouven M, E GI (2016) Optimization models for planning shale gas well refracture treatments. AIChE Journal 62:4297–4307

Cafaro DC, Drouven M, Grossmann I (2018) Continuous-time formulations for the optimal planning of multiple refracture treatments in a shale gas well. AIChE Journal 64:1511–1517

Carvalho MCA, Pinto J (2006) A bilevel decomposition technique for the optimal planning of offshore platforms. Brazilian Journal of Chemical Engineeringl 23(1):67–82

Drouven M, Grossmann I (2017) Stochastic programming models for optimal shale well development and refracturing planning under uncertainty. AIChE Journal 63(11):4799–4813

Drouven MG, Grossmann I (2016) Multi-period planning, design, and strategic models for long-term, quality-sensitive shale gas development. Process Systems Engineeringl 62:2296–2323

Etherington J, McDonald IR (2004) Is bitumen a petroleum reserve? SPE annual technical conference and exhibition Society of Petroleum Engineers

Garey M (1979) Computers and intractability: A guide to the theory of NP-completeness. Freeman, NewYork

Goel V, Grossmann I (2004) A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. Computers and chemical engineering 28(8):1409–1429

Gupta J (1988) Two-stage, hybrid flowshop scheduling problem. Journal of Operational Research Society 39(4):359–364

Iyer RR, Grossmann IE, Vasantharajan S, Cullick AS (1998) Optimal planning and scheduling of offshore oil field infrastructure investment and operations. Industrial and Engineering Chemistry Researchl 37:1380–1397

Knudsen B, Foss B (2013) Shut-in based production optimization of shale-gas systems. Computers and Chemical Engineering 58:54–67

Lee T, Loong Y (2019) A review of scheduling problem and resolution methods in flexible flow shop. International Journal of Industrial Engineering Computations 10:67–88

Lin X, Christodoulos A (2003) A novel continuous-time modeling and optimization framework for well platform planning problems. Optimization and Engineering 4:65–95

Marquant JF, Evins R, Carmeliet J (2015) Reducing computation time with a rolling horizon approach applied to a milp formulation of multiple urban energy hub system. ICCS 2015 International Conference On Computational Science pp 2137–2146

Office of Fossil Energy (2013) Natural gas from shale. US Department of Energy

Ondeck A, Drouven M, Blandino N, Grossmann I (2019) Multi-operational planning of shale gas pad development. Computers and Chemical Engineering 126:83–101

Rahmanifard H, Plaksina T (2018) Application of fast analytical approach and ai optimization techniques to hydraulic fracture stage placement in shale gas reservoirs. Journal of Natural Gas Science and Engineering 52:367–378

Sam M, DAriano A, Pacciarelli D (2013) Rolling horizon approach for aircraft scheduling in the terminal control area of busy airports. Procedia-Social and Behavioral Sciences 80:531–552

Silvente J, Kopanos GM, Pistikopoulos EN, Espuna A (2015) A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids. Applied Energy 155:485–501

Spratt B, Kozan E (2018) An integrated rolling horizon approach to increase operating theatre efficiency. https://arxiv.org/abs/1808.10139

US Energy Information Administration EIA (2019) Tight oil development will continue to drive future U.S. crude oil production. Today in Energy

Xie J, Wang X (2005) Complexity and algorithms for two-stage flexible flowshop scheduling with availability constraints. Computers and Mathematics with Applications 50(10-12):1629–1638