Flexibility vs. efficiency: a study in sawmill scheduling

Csaba Kebelei
Eötvös Loránd University
Faculty of Informatics
Budapest, Hungary
kebeleicsaba@student.elte.hu

Máté Hegyháti
University of Sopron
Sopron, Hungary
University of Pannonia
Veszprém, Hungary

Abstract

Sawmills are an important stage in the primary wood industry, producing timber and other side products from harvested logs. Increasing efficiency in a sawmill is not only an economic desire, but it also propagates to the competitiveness of using more renewable resources in dependent industries such as construction, packaging, furniture. Although the road from a tree log to a shipped product involves many stages, the key processing step is the sawing performed by high-value sawing machines. As the operation of these machines plays a key role in overall efficiency, several research papers have addressed their scheduling. Naturally, allowing more freedom in the production plan may result in better financial results in exchange for increased computational needs of the optimizer. However, more complex schedules may also pose an additional burden in their real-life execution. This paper presents several variants of a model formerly proposed by the authors and investigates this trade-off relationship empirically.

Keywords

sawmill, scheduling, MILP

1 Introduction and literature

The increasing utilization of renewable resources is a major directive of many nations and companies around the world. Wood is a natural resource that not only serves as a great carbon sequestration tool, but it is a suitable raw material for many industries, including construction and interior design, paper, packaging, and the boat industry. After logging, sawmills are generally the first stage in timber processing regardless of the end-use industry. Their main task is to turn unprocessed logs from the forest into well-cut lumber. While this is a multi-stage process, the main step is the sawing itself that can be done by two of the main machine types: band saws and frame saws. The efficient operation of these machines have a significant impact on the whole plant, thus, several works have addressed this issue. Efficiency is tackled on both the operational and the planning level by designing cutting patterns and scheduling, respectively [6]. A tree log may be cut in different patterns yielding different quantities of different products, as shown by in Figure 1. Designing patterns based on long-term demands for products, statistics about defects, etc. is a well-researched topic in the literature. Moreover, scanning tools available on modern sawing machines can provide real-time information about logs to make such decisions adaptive [5]. On the planning level, selecting the cutting patterns and the corresponding log quantities for each shift while considering orders, available logs, storage capacity, etc. is a scheduling problem. The base problem was introduced by Zanjani et al. [7] and Maturana et al. [9] for the stochastic and deterministic cases, respectively. Subsequent works extended the approach in various directions, e.g., addressing uncertainties [2, 11], integrating cutting pattern selections [3, 10], or minimizing waste [4].

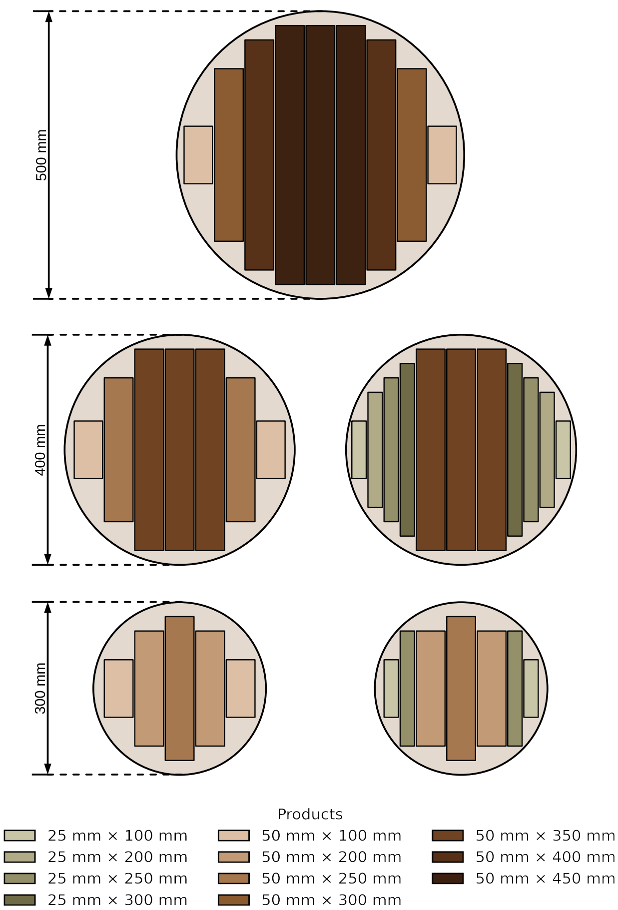


Figure 1: Cutting patterns for test cases in Section 5

In this paper, a previously proposed model by the authors [8] is further investigated, that addressed issues prevalent for small-scale sawmills: workforce availability and operation differences between sawing technologies. Compared to band saws, the changeover time for switching cutting patterns on a frame saw is non-negligible, thus the same formulation is not applicable in short-term scheduling. The proposed model introduced a significantly different formulation for the behavior of the frame saw, that only allows one change per

MATCOS-25, Koper, Slovenia
2025.

shift following industry practice. In later Sections this base model will be referred to as **C−EK**, where **EK** (expert knowledge) indicates at most one change in a shift, that may be timed continuously (**C**) at any time.

The aim of this work is to investigate, whether allowing more cutting pattern changes on a frame saw would yield significantly better results due to a broader solution space, and/or does it increase the computational costs comparably. While allowing such flexibility in a mathematical model may be simple, one has to keep in mind, that the execution of overly complicated schedules in an non-automated small-scale plant is not realistic, thus a set of models will also be introduced, where cutting pattern changes may only happen at predefined discrete (**D**) time points, as detailed in Section 4. Another decision freedom, whose effect will be investigated in Section 5 is the possible reallocation of the workforce from one machine to another during a shift, as discussed in Section 3.

2 Problem definition

The objective is the same as in [8]: minimizing the under-production cost in a small-scale sawmill equipped with one frame saw and one band saw. The sawmill may operate in multiple shifts, however, from the modeling point of view, only the total number of shifts is relevant. Thus, without the loss of generality, it is assumed, that each day has a single shift (of length H), and the two terms are used interchangeably. For each day ($d \in \mathcal{D}$), the number of specialists (HR_d^{SP}) and additional workers (HR_d^{AW}) are given, along with the requirements to operate each machine ($RR^{SP,F}$, $RR^{SP,B}$,$RR^{AW,F}$, $RR^{AW,B}$). Resource related data is also given, i.e., the quantity (I_l) and volume (V_l) of logs of different sizes ($l \in \mathcal{L}$) all available at the start of the planning horizon. For each log size several cutting patterns ($\mathcal{P}_l \subseteq \mathcal{P}$) may be available with known yields ($Y_{t,p}$) for lumber products ($t \in \mathcal{T}$). For each shift the production planner may decide which machines to operate and how many logs with which cutting patterns are processed.

By allowing different flexibilities, several different problem definitions will be addressed by their corresponding models:

- **C−EK** Original problem from [8]: the frame saw may change cutting pattern once per shift, but a saw is either operating through the whole shift or not.
- **C′−EK** Same as **C−EK**, but the frame saw may also be turned on/off, and the workforce reallocated to the band saw instead of changing the pattern, once per shift.
- **D[K]** The shift is subdivided into k segments of equal length. Pattern changes and workforce reallocations are allowed only at the end of these intervals.
- **D[K]−EK** Same as **D[K]**, but the frame saw is allowed to change cutting patterns at most once a day.

If $OPT(M)$ denotes the optimal (minimal) solution for the problem/model M, the following inequalities will hold naturally:

- $OPT($**C′−EK**$) \leq OPT($**C−EK**$)$
- $OPT($**C′−EK**$) \leq OPT($**D[X]−EK**$)$
- $OPT($**D1**$) = OPT($**D1−EK**$)$
- $OPT($**D[X]**$) \leq OPT($**D[X]−EK**$)$
- $OPT($**D[Y]**$) \leq OPT($**D[X]**$)$ if $X \mid Y$
- $OPT($**D[Y]−EK**$) \leq OPT($**D[X]−EK**$)$ if $X \mid Y$

3 Model with workforce reallocation: C'−EK

The event, when worker reallocation is also possible in addition to pattern changes on the frame saw, will be referred to as the changeover. As this model is based on **C−EK**, constraints (1)–(5), (7)–(8), and (10) from [8] are copied verbatim, expressing the objective, storage balances, daily production quantities, inventory limits, frame saw production bounds before the changeover, and frame saw activation bounds.

Binary variables w_d^F and w_d^B are split to $w_d^{F,-}$,$w_d^{B,-}$ and $w_d^{F,+}$,$w_d^{B,+}$, indicating whether the machines are active before or after the changeover. Duplicated versions of constraints (15)–(16) with these variables are also copied to express human resource needs. Superscripts + and − will indicate such division similarly in other variables, and ± will be used to indicate both cases.

The band saw time capacity constraint is reformulated as:

$$\sum_{p \in \mathcal{P}} ST_p^B \cdot q_{d,p}^B \leq \tau_d^{B,-} + \tau_d^{B,+} \qquad \forall d \in \mathcal{D} \quad (1)$$

Where variables $\tau_d^{B,\pm} \in [0, H]$ represent the available band saw cutting time, linked to changeover timing by the following constraints:

$$\tau_d^{B,\pm} \leq H \cdot w_d^{B,\pm} \qquad \forall d \in \mathcal{D},\ \pm \in \{-,+\} \quad (2)$$

$$\tau_d^{B,-} \leq t_d \qquad \forall d \in \mathcal{D} \quad (3)$$

$$\tau_d^{B,+} \leq H - t_d \qquad \forall d \in \mathcal{D} \quad (4)$$

Constraint (2) activates $\tau_d^{B,\pm}$ only when the band saw is active in the corresponding part of the shift, while constraints (3) and (4) bound them by t_d and $H - t_d$, respectively.

For the frame saw, the following constraint limits the production after the changeover:

$$\sum_{p \in \mathcal{P}} ST_p^F \cdot q_{d,p}^{F,+} \leq (H - t_d) - CT^F \cdot z_d^{F,CT} + H \cdot \left(1 - w_d^{F,+}\right) \qquad \forall d \in \mathcal{D} \quad (5)$$

Where $z_d^{F,CT} \in \{0, 1\}$ indicates whether changeover time must be deducted after the changeover. Constraint (6) sets $z_d^{F,CT}$ to one, if the frame saw is active through the whole shift and the pattern is changed.

$$z_d^{F,CT} \geq w_d^{F,-} + w_d^{F,+} + \left(s_{d,p}^{F,+} - s_{d,p}^{F,-}\right) - 2 \qquad \forall d \in \mathcal{D},\ p \in \mathcal{P} \quad (6)$$

Constraint (7) ensures that only an active frame saw can have an active cutting pattern, which is preserved by constraints (8) and (9) for the next day if the saw remains active.

$$\sum_{p \in \mathcal{P}} s_{d,p}^{F,\pm} = w_d^{F,\pm} \qquad \forall d \in \mathcal{D},\ \pm \in \{-,+\} \quad (7)$$

$$s_{d,p}^{F,+} \geq s_{d+1,p}^{F,-} - 1 \cdot \left(2 - w_d^{F,+} - w_{d+1}^{F,-}\right) \qquad \forall d \in \mathcal{D},\ p \in \mathcal{P} \quad (8)$$

$$s_{d,p}^{F,+} \leq s_{d+1,p}^{F,-} + 1 \cdot \left(2 - w_d^{F,+} - w_{d+1}^{F,-}\right) \qquad \forall d \in \mathcal{D},\ p \in \mathcal{P} \quad (9)$$

4 Discrete time-slot models: D[K], D[K]−EK

Similarly to the previous modification of the original model, some constraints from [8] remain unmodified, expressing the objective, storage balances: (1)–(3). Moreover, some variables are now defined for each segment of a shift, indexed by the set $\mathcal{K} = \{1, 2, \ldots, K\}$.

For example, the binary variables $w^F_{d,k}$, $w^B_{d,k}$ indicate machine usage, which are used in segment-wise copies of (15)–(16) of [8] to express human-resource requirements.

$w^B_{d,k}$ is also used in the band saw time capacity constraint, where $H^K = H/K$ denotes the length of a segment:

$$\sum_{p\in\mathcal{P}} ST^B_p \cdot q^B_{d,p} \le \sum_{k\in\mathcal{K}} H^K \cdot w^B_{d,k} \qquad \forall d \in \mathcal{D} \quad (10)$$

Another such variable is $q^F_{d,k,p} \in \mathbb{Z}_{\ge 0}$ indicating the frame saw production, used in constraints (11) and (12) to calculate the daily production quantity and limit the inventory:

$$y_{d,t} = \sum_{p\in\mathcal{P}} Y_{t,p} \cdot V_{l_p} \cdot \left(q^B_{d,p} + \sum_{k\in\mathcal{K}} q^F_{d,k,p} \right) \qquad \forall d \in \mathcal{D},\ t \in \mathcal{T} \quad (11)$$

$$\sum_{d\in\mathcal{D}} \sum_{p\in\mathcal{P}_l} \left(q^B_{d,p} + \sum_{k\in\mathcal{K}} q^F_{d,k,p} \right) \le I_l \qquad \forall l \in \mathcal{L} \quad (12)$$

Active consecutive segments on the frame saw should be merged together if the pattern is unchanged. These conditions are expressed by binary variable $s^F_{d,k,p}$ which is 1 iff pattern p is active in segment d, k and the one preceding it. Another binary indicator variable, $s^{c,F}_{d,k,p}$ is 1 iff p is active in segment k but not in the preceding one. This logic is enforced by the following constraints:

$$\sum_{p\in\mathcal{P}} \left(s^{c,F}_{d,k,p} + s^F_{d,k,p} \right) = w^F_{d,k} \qquad \forall d \in \mathcal{D},\ k \in \mathcal{K} \quad (13)$$

$$s^F_{d,k,p} \le s^F_{d,k-1,p} + s^{c,F}_{d,k-1,p} + \left(1 - w^F_{d,k-1}\right) \qquad \forall d \in \mathcal{D},\ k \in \mathcal{K} \setminus \{1\},\ p \in \mathcal{P} \quad (14)$$

$$s^F_{d,1,p} \le s^F_{d-1,K,p} + s^{c,F}_{d-1,K,p} + \left(1 - w^F_{d-1,K}\right) \qquad \forall d \in \mathcal{D} \setminus \{1\},\ p \in \mathcal{P} \quad (15)$$

Constraint (13) ensures that when the frame saw is operated in segment k, exactly one pattern is selected (either a changeover to p or continuation with p), and when it is idle, neither indicator is active. Constraint (14) allows pattern continuation only if p was active in the previous segment, and the frame saw is active. The same condition is set across days is by Constraint (15).

The merging of active segments with the same pattern is modeled by constraints (16), (17), and (18), introducing a spare time variable $\sigma_{d,k,p} \in [0, H^K]$ that records unused time in segment k for pattern p and can be carried to the next segment.

$$\sigma_{d,k,p} \le ST^F_p \cdot s^F_{d,k+1,p} \qquad \forall d \in \mathcal{D},\ k \in \mathcal{K} \setminus \{K\},\ p \in \mathcal{P} \quad (16)$$

$$ST^F_p \cdot q^F_{d,1,p} \le \left(H^K - CT^F\right) \cdot s^{c,F}_{d,1,p} + H^K \cdot s^F_{d,1,p} - \sigma_{d,1,p} \qquad \forall d \in \mathcal{D},\ p \in \mathcal{P} \quad (17)$$

$$ST^F_p \cdot q^F_{d,k,p} \le \left(H^K - CT^F\right) \cdot s^{c,F}_{d,k,p} + H^K \cdot s^F_{d,k,p} + \sigma_{d,k-1,p} - \sigma_{d,k,p} \qquad \forall d \in \mathcal{D},\ k \in \mathcal{K} \setminus \{1\},\ p \in \mathcal{P} \quad (18)$$

Constraint (16) limits the carryover from segment k to one log's sawing time ST^F_p and activates it only if the next segment continues with the same pattern, while the production time bounds follow from Eqs. (17) and (18), accounting for changeover time CT^F and the inflow/outflow of spare time.

To obtain D[K]−EK from D[K], Constraint (19) needs to be added, which enforces at most one cutting pattern change by the frame saw per day:

$$\sum_{k\in\mathcal{K}} \sum_{p\in\mathcal{P}} s^{c,F}_{d,k,p} \le 1 \qquad \forall d \in \mathcal{D} \quad (19)$$

5 Empirical results

To evaluate the new approaches, the randomly generated test set from our previous study [8] is used, where cutting pattern yields were precomputed with Pitago Optimizers [1] and technical parameters reflect industry practice and recommendations by experts. Figure 1 shows the five patterns defined for three log diameter classes in all of the test cases.

For practical reasons in execution, the values 1, 2, 4, and 8 were considered for K, and the **EK** variant for $K = 1$ is omitted, as it is equivalent to the non-**EK** case.

All problems in the dataset were solved with all of the models by Gurobi Optimizer 12.0.1 on a computer with an Apple M2 CPU and 16 GB of RAM available, with a time limit of 1500 seconds.

Table 1 shows the aggregated results from the 50 cases with 4-week planning horizon.

The left block shows objective value distributions (m^3) as box plots. The median relative deviation from the **C−EK** reference is also reported in the column labeled $\tilde{\Delta}_{\text{C−EK}}$ (%). On the right, CPU times (s) are shown as box plots on a base-10 logarithmic scale.

The results match the theory:

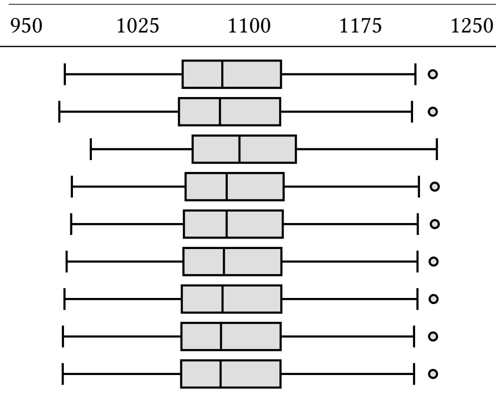
- Adding on/off reallocation in **C′−EK** increases quality
- Discretizing the changeover timing degrades it when only one cutting pattern change is allowed per shift (**EK** models), while smaller slot lengths mitigate the loss.
- Imposing the "at most one change" rule on the frame saw (**−EK**) has a negative impact on solution quality.

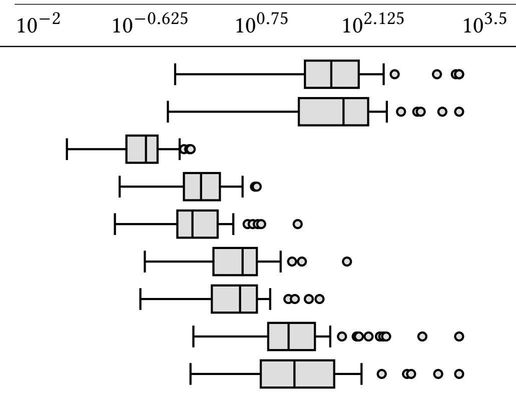
It can be observed, however, that the differences in objective values across models are small: $\tilde{\Delta}_{\text{C−EK}}$ ranges from −0.1072% for **C′−EK** (best) to +0.8536% for **D1** (worst), and difference between D[K] and D[K]−EK is less than 0.05 percentage points.

A key takeaway is that even **D8** does not outperform **C−EK**. For these test cases, continuously timing a single change is more valuable than permitting multiple changes at fixed time-slot boundaries. Even so, the loss from discretization is small: as k increases, the median gap shrinks from +0.85% (**D1**) to about +0.0058% for **D8−EK** or +0.0024% for **D8**.

Regarding runtimes, increasing K for the discrete models increases CPU time, as expected, yet the $K = 8$ variants still run faster

Table 1: Results on randomly generated instances with a 4-week planning horizon across model variants.





Model	$\tilde{\Delta}_{\texttt{C-EK}}$ (%)	Avg.
C−EK	+0.0000	151.36
C′−EK	-0.1072	125.98
D1	+0.8536	0.24
D2−EK	+0.2908	1.37
D2	+0.2440	1.52
D4−EK	+0.1065	5.00
D4	+0.0760	4.21
D8−EK	+0.0058	67.76
D8	+0.0024	82.82

on average than the continuous baselines. Notably, **D8** is within ≈ 0.11 percentage points of **C′−EK** in objective value yet is faster on average. The **−EK** restriction leaves quality essentially unchanged and tends to reduce runtime.

Time-limit hits were infrequent (Table 2).

Table 2: Time-limit hits (counts).

Model	3-week	4-week	5-week	6-week
C−EK	3	2	5	4
C′−EK	0	1	2	3
D8−EK	0	1	1	0
D8	0	1	1	1

These counts suggest that the discrete-slot models remain computationally stable across longer planning horizons.

Overall, the discrete time-slot-based models offer a tunable trade-off: coarse slotting is extremely fast but less accurate, whereas $K = 8$ achieves near-continuous quality at a substantially lower average runtime than the continuous baselines.

6 Concluding remarks

We revisited a MILP scheduling model for small-scale sawmills and examined three design choices: (i) allowing workforce reallocation, (ii) restricting changeovers to discrete within-shift time-slot boundaries, and (iii) allowing multiple changes per shift. Computational tests on a large synthetic instance set show that objective differences across variants are modest. Allowing on/off operation with reallocation provides a consistent, albeit small, improvement in solution quality, whereas discretizing the changeover into time slots introduces a controllable loss that diminishes as the slot granularity is refined. Overall, the discrete formulations offer a tunable trade-off between accuracy and speed. In our setting, even coarse slots perform well, while a moderate refinement (e.g., hourly segments) achieves near-continuous quality at substantially lower average runtimes. In practice, multiple within-shift changes bring marginal gains and complicate execution, whereas **−EK** yields simpler plans with similar quality and often lower runtime. For the investigated test cases, even completely forbidding changes within shifts resulted in less than 1 percent quality reduction. These findings support the discrete-slot approach as a practical planning tool for small, low-automation sawmills. Future work will expand the set of test instances, include an industrial case study to validate and calibrate the models on operational data, and, where appropriate, incorporate uncertainty in key inputs.

References

[1] *Pitago Optimizers.* Retrieved August 24, 2025 from https://pitago.eu/

[2] Pamela P Alvarez and Jorge R Vera. 2014. Application of robust optimization to the sawmill planning problem. *Annals of Operations Research* 219 (2014), 457–475. doi:10.1007/s10479-011-1002-4

[3] Diego Broz, Nicolás Vanzetti, Gabriela Corsano, and Jorge M. Montagna. 2019. Goal programming application for the decision support in the daily production planning of sawmills. *Forest Policy and Economics* 102 (2019), 29–40. doi:10.1016/j.forpol.2019.02.004

[4] Giacomo Da Col, Philipp Fleiss, Alice Tarzariol, Erich C. Teppan, and Elena Wiegelmann. 2025. A Declarative Approach to Tackle Sawmill Production Scheduling with Answer Set Programming. In *Computational Science and Computational Intelligence*, Hamid R. Arabnia, Leonidas Deligiannidis, Farid Ghareh Mohammadi, Soheyla Amirian, and Farzan Shenavarmasouleh (Eds.). Springer Nature Switzerland, Cham, 313–323. doi:10.1007/978-3-031-90341-0_23

[5] Kamran Forghani, Mats Carlsson, Pierre Flener, Magnus Fredriksson, Justin Pearson, and Di Yuan. 2024. Maximizing value yield in wood industry through flexible sawing and product grading based on wane and log shape. *Computers and Electronics in Agriculture* 216 (2024), 108513. doi:10.1016/j.compag.2023.108513

[6] Seyed Mohsen Hosseini and Angelika Peer. 2022. Wood Products Manufacturing Optimization: A Survey. *IEEE Access* 10 (2022), 121653–121683. doi:10.1109/ACCESS.2022.3223053

[7] Masoumeh Kazemi Zanjani, Daoud Ait-Kadi, and Mustapha Nourelfath. 2010. Robust production planning in a manufacturing environment with random yield: A case in sawmill production planning. *European Journal of Operational Research* 201, 3 (2010), 882–891. doi:10.1016/j.ejor.2009.03.041

[8] Csaba Kebelei and Mate Hegyhati. 2024. Sawmill scheduling: an application-oriented model. *Acta Technica Jaurinensis* 17, 3 (2024), 104–110. doi:10.14513/actatechjaur.00743

[9] Sergio Maturana, Enzo Pizani, and Jorge Vera. 2010. Scheduling production for a sawmill: A comparison of a mathematical model versus a heuristic. *Computers & Industrial Engineering* 59, 4 (2010), 667–674. doi:10.1016/j.cie.2010.07.016

[10] Nicolás Vanzetti, Diego Broz, Gabriela Corsano, and Jorge M. Montagna. 2018. An optimization approach for multiperiod production planning in a sawmill. *Forest Policy and Economics* 97 (2018), 1–8. doi:10.1016/j.forpol.2018.09.001

[11] Mauricio Varas, Sergio Maturana, Rodrigo Pascual, Ignacio Vargas, and Jorge Vera. 2014. Scheduling production for a sawmill: A robust optimization approach. *International Journal of Production Economics* 150 (2014), 37–51. doi:10.1016/j.ijpe.2013.11.028