Odile Marcotte

Comptes rendus

Huitième atelier de résolution de problèmes industriels de Montréal

7 au 11 août 2017

Proceedings

Eighth Montréal Industrial Problem Solving Workshop

August 7–11, 2017

CRM-3369

Préface

Le huitième atelier de résolution de problèmes industriels de Montréal eut lieu au CRM du 7 au 11 août 2017. Il fut parrainé par le CRSNG (grâce à la Plateforme d'innovation des instituts ou PII), le CRM, le Fields Institute, le PIMS et l'INCASS (Institut canadien des sciences statistiques). Parmi tous les ateliers de ce genre organisés par le CRM, ce fut celui qui a eu la plus grande envergure: neuf problèmes furent étudiés par 20 professeurs, 22 représentants industriels et 65 étudiants, stagiaires postdoctoraux ou assistants de recherche. Notons en particulier que huit étudiants français participèrent à l'atelier dans le cadre d'un échange entre les instituts canadiens de mathématiques et l'Institut national des sciences mathématiques et de leurs interactions (un institut du CNRS). Nous exprimons notre vive reconnaissance aux sept entreprises ou institutions qui ont fourni des problèmes à l'atelier: la Banque Nationale du Canada, Co-operators, FPInnovations, Optina Diagnostics, Rio Tinto, CWP Énergie et le Conseil national de recherches du Canada.

Le succès de l'atelier dépend naturellement en très grande partie de l'implication des chercheurs qui acceptent de coordonner les travaux des équipes. Nous remercions donc chaleureusement les professeurs Bruno Rémillard (HEC Montréal), Bernard Gendron (Université de Montréal), Philippe Langlais (Université de Montréal), Jean-Marc Frayret (Polytechnique Montréal), Jean-François Plante (HEC Montréal), Farida Cheriet (Polytechnique Montréal), Pierre Duchesne (Université de Montréal), Manuel Morales (Université de Montréal) et Huaxiong Huang (York University et Fields Institute). Nous remercions également les étudiants qui ont assumé la coordination de l'écriture des rapports: Anastasis Kratsios et Behnoosh Zamanlooy (pour la Banque Nationale du Canada), Steven Lamontagne (pour le premier problème de Co-operators), William Léchelle (pour le second problème de Co-operators), Jakub Witkowski (pour FPInnovations), Jeremy Budd (pour Optina Diagnostics), Diana Jovmir (pour Rio Tinto), Yiran Wang (pour CWP Énergie) et Zilong Song (pour le CNRC).

Finalement nous remercions le CRSNG d'avoir accordé la subvention appelée PII aux trois instituts canadiens de mathématiques. Cette subvention a permis, entre autres choses, d'organiser un atelier qui a connu un très grand succès.

Les membres du comité d'organisation
Thierry Duchesne (Université Laval)
Michael Lamoureux (University of Calgary et PIMS)
Odile Marcotte (UQAM et CRM)
Tom Salisbury (York University et Fields Institute)
Stéphane Rouillon (CRM)

Foreword

The Eighth Montreal Industrial Problem Solving Workshop was held at the CRM on August 7-11, 2017. It was sponsored by NSERC (through the Institutes Innovation Platform), the CRM, the Fields Institute, PIMS, and CANSSI (the Canadian Statistical Sciences Institute). The Eighth Montreal IPSW was the largest IPSW organized by the CRM: nine problems submitted by industry were tackled by 20 professors, 22 industrial representatives, and 65 students, postdoctoral fellows, and research assistants. Note in particular that eight French students took part in the workshop within the framework of an exchange between the Canadian mathematics institutes and INSMI (a CRNS institute). We wish to express our gratitude to the seven companies or institutions that submitted problems to the workshop: National Bank of Canada, The Co-operators, FPInnovations, Optina Diagnostics, Rio Tinto, CWP Energy, and the National Research Council.

The success of the workshop is due, to a large extent, to the coordination work of the professors who mentor the teams. Therefore we extend our warmest thanks to Bruno Rémillard (HEC Montréal), Bernard Gendron (Université de Montréal), Philippe Langlais (Université de Montréal), Jean-Marc Frayret (Polytechnique Montréal), Jean-François Plante (HEC Montréal), Farida Cheriet (Polytechnique Montréal), Pierre Duchesne (Université de Montréal), Manuel Morales (Université de Montréal), and Huaxiong Huang (Fields Institute and York University). We are also grateful to the students who coordinated the writing of the reports: Anastasis Kratsios and Behnoosh Zamanlooy (National Bank of Canada), Steven Lamontagne (first problem submitted by The Co-operators), William Léchelle (second problem submitted by The Co-operators), Jakub Witkowski (FPInnovations), Jeremy Budd (Optina Diagnostics), Diana Jovmir (Rio Tinto), Yiran Wang (CWP Energy), and Zilong Song (NRC).

Finally we thank NSERC for having granted the Institutes Innovation Platform (IIP) to the Canadian mathematics institutes. The IIP allowed us to organize a very successful workshop (among other things).

The Organizing Committee
Thierry Duchesne (Université Laval)
Michael Lamoureux (University of Calgary and PIMS)
Odile Marcotte (UQAM and CRM)
Tom Salisbury (York University and Fields)
Stéphane Rouillon (CRM)

Contents

1
Replication of a real-estate market index (Teranet – National Bank of Canada)

Anastasis Kratsios, Behnoosh Zamanlooy, Bruno Rémillard, Ying Li, Delphine Boursicot, Stéphane Galzin, Charles Sam, Seong-Hwan Jun, Abbas Mehrabian, Florian Lay, Vincent Croisé, Tao Lei, Jean Hounkpe, Inesh Munaweera Arachchilage, Wenyan Zhong, Xiang Gao, Negin Pasban Roozbahani, and Tae Yoon Lee

1.1 Introduction

1.1.1 *Canadian Housing Index Replication*

The typical price of a house in Canada has been steadily increasing over the last few decades. Moreover this increase has shown resilience during the market turmoil caused by the 2008 financial crisis. The red line in the following graph represents the national housing index, a global measure of the overall Canadian real-estate market. The grey portion represents the 12-month change in the index, expressed as a percentage.

House prices have been measured in many ways and of particular interest to this project is the Teranet – National Bank housing index. This monthly housing index represents the average house cost across Canada by cleverly weighting the average pricing of housing in large metropolitan clusters. As can be seen on the following figure, Toronto naturally holds most of the weight of the Canadian (Composite 11) housing index, with Vancouver and Montréal coming closely behind. Other major national cities naturally contribute less to the global picture.

Anastasis Kratsios · Behnoosh Zamanlooy · Xiang Gao
Concordia University

Bruno Rémillard · Delphine Boursicot · Stéphane Galzin
HEC Montréal

Ying Li
University of Alberta

Charles Sam · Wenyan Zhong
University of Calgary

Seong-Hwan Jun · Tae Yoon Lee
University of British Columbia

Abbas Mehrabian · Tao Lei
McGill

Florian Lay · Vincent Croisé · Jean Hounkpe
Université de Montréal

Inesh Munaweera Arachchilage · Negin Pasban Roozbahani
University of Manitoba

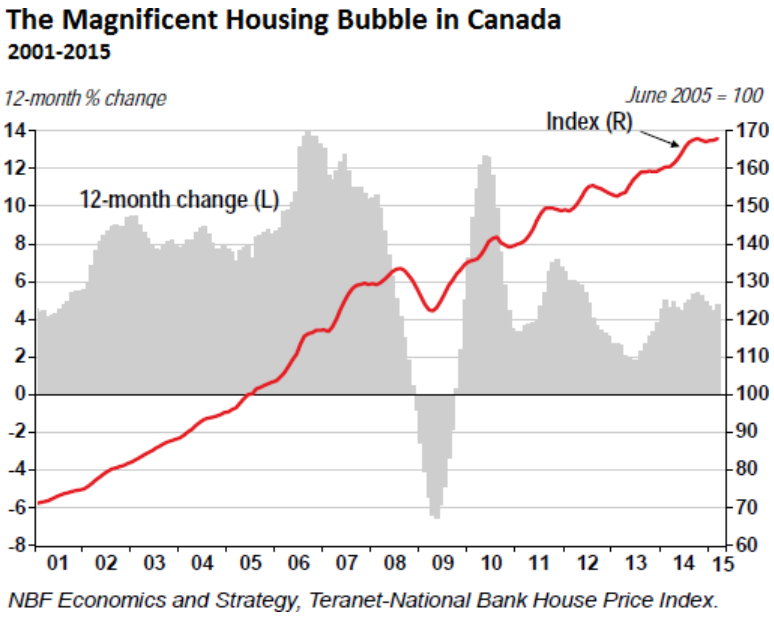


Fig. 1.1: Teranet – National Bank housing index makeup.

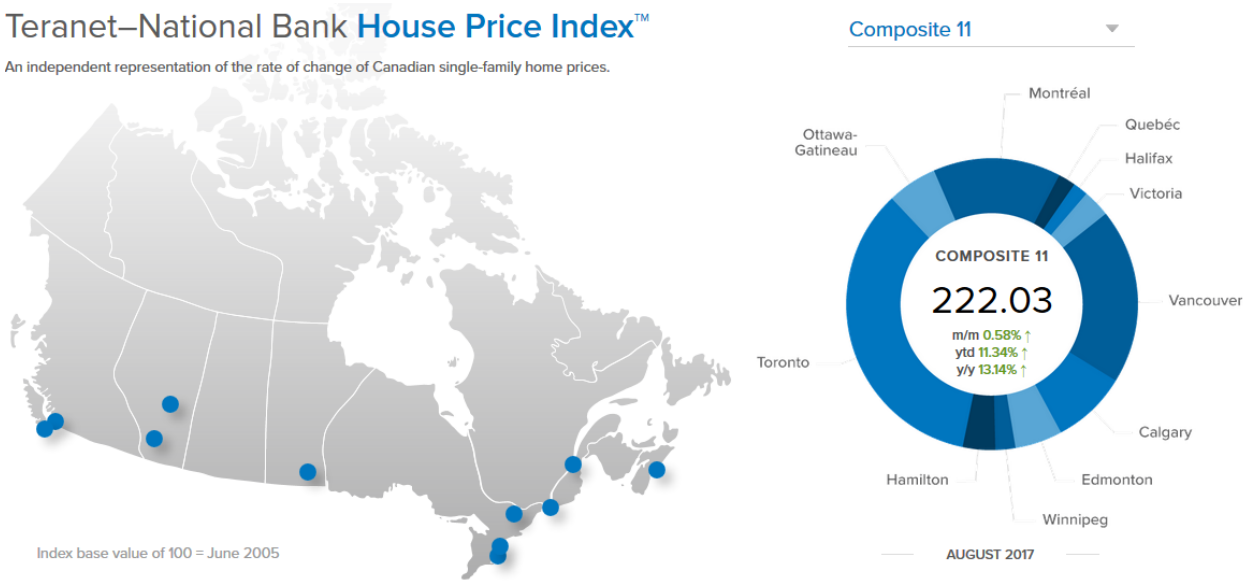


Fig. 1.2: Teranet – National Bank housing index makeup.

These two facts make investing in the Canadian real estate as a whole highly desirable. At face value, however, this investing seems impossible, since it would require buying portions of people's houses all around the world, not to mention the liquidity risk when selling people's homes. Likewise, investing in an index like the Teranet – National Bank housing index is impossible since it is not a product but only reflects the current market climate.

We propose to get around the infeasibility of investing in the Canadian real-estate market as a whole by investing directly in a replicator of the index. In the same fashion as ETFs are used to track market indices

like the S&P 500 or the VIX (where market instruments are used to recreate and track the market index), we will attempt to recreate the housing index. This problem is in fact more sophisticated and complicated than a classical market index replication problem since the market instruments are people's houses (which cannot be used to build a portfolio).

Therefore our problem is twofold. We first seek to identify a set of liquid financial instruments that are available in the market. Then we will use portfolios built from those instruments to track the housing index. We now turn to a formal statement of our problem.

1.1.2 *Mathematical Formulation*

Let $\{R_t^{f,i}\}_{i=1}^d$ denote the set of instantaneous excess returns on d financial instruments at time t, which we write in vector form as R_t^f. We wish to invest precisely w_t^i in the ith asset at time t. We denote the proportions invested in each asset at time t by the d-dimensional vector w_t. Therefore the excess returns of our portfolio at time t must be equal to

$$\sum_{i=1}^{d} w_t^i R_t^i, \tag{1.1}$$

which we may write more compactly as

$$w_t^T R_t^f, \tag{1.2}$$

where $\cdot^T$ symbolizes the transpose operation.

If we denote the instantaneous excess returns of the Teranet housing index at time t by R_t^I, our first goal is to minimize the difference between R_t^I and the excess returns of our portfolio $w_t^T R_t^f$. We measure this mis-pricing error according to the mean-squared error (MSE) loss function. Therefore we seek to minimize the quantity

$$\mathrm{MSE}(w_t) \triangleq \mathbb{E}\left[\left(R_t^I - w_t^T R_t^f\right)^2\right]. \tag{1.3}$$

Since the market is dynamic, however, it is not enough to minimize Equation (1.3) dynamically: rather we wish to predict the movement of the minimizer of Equation (1.3). To this end, we predict the movement of the distribution of the minimizer (which we denote by π_t). The prediction of the movement of the minimizer of the MSE loss function has been well studied in probability theory and electrical engineering. The problem of predicting the distribution of π_t is known as the stochastic filtering problem and goes back to the early days of NASA.

We now state the mathematical problems formally.

Model Setup Part 1. Let

$$\mathit{\Pi}_{\text{all}} \triangleq \{X_1, \ldots, X_D\}$$

be a set of D liquid financial instruments on the Canadian market. Suppose that there is a subcollection

$$\mathit{\Pi}_{\text{true}} \triangleq \{X_{i_1}, \ldots, X_{i_d}\}$$

of those assets such that for every $X_k \in \mathit{\Pi}_{\text{all}} - \mathit{\Pi}_{\text{true}}$,

$$w_t^k = 0,$$

except for a set of $(\mathbb{P} \otimes m)$-measure 0.

Problem 1. Learn the set Π_{true} (at least asymptotically).

Model Setup Part 2. Let $(\Omega, \mathfrak{F}, \mathbb{P})$ be a complete probability space.

$$\begin{aligned} dw_t &= \alpha(t, w_t)dt + \beta(t, w_t)d\varepsilon_t \\ dR_t^I &= \left[\left(w_t^T R_t^f\right) + \left(1 - \sum_{i=1}^{d} w_{t,i}\right) r_t\right] dt + Qd\epsilon_t \end{aligned} \tag{1.4}$$

In (1.4)

- R_t^I is the excess returns of the index at time t,
- R_t^f is the vector of excess returns of tradeable securities at time t,
- w_t is a vector of $w_{t,i}$, containing the weight invested in each asset i at time t,
- r_t is the risk-free rate at time t (proxied by the 1-month deposit rate),
- Q is the volatility of the replication errors, and
- ϵ_t, ε_t are independent Lévy processes defined on the same complete probability space $(\Omega, \mathfrak{F}, \mathbb{P})$. [1]

Problem 2. Estimate and predict the dynamics of the density π_t of the MSE error function optimizer. [2]

$$\pi_t \ \widetilde{D} \underset{Z \in L^2(\Omega \times [0,t), \mathfrak{F}_t, \mathbb{P} \otimes m)}{\arg\min} \mathbb{E}\left[\left(R_t^I - w_t^T R_t^f\right)^2\right]. \tag{1.5}$$

An alternative way to understand Equation (1.5) is that we seek to filter out the unobservable process π_t given the market data that secretly encodes it.

1.1.3 *Our Approach*

Since this is a two-fold problem we propose a two-phase approach. Firstly, we use two machine learning techniques in order to identify meaningful factors parsimoniously in a robust manner. More precisely, we use the method of *gradient boosting with early stopping* studied in [10] and in parallel the Adaptive Elastic Net of [12], which is an improvement upon the well-known LASSO of [9]. This is the identification step.

Once the true contributing factors have been identified, we construct a dynamic portfolio using those financial instruments to track the index. This is the replication step, carried out through stochastic filtering methods. Our primary technique is to employ a wisely chosen particle filter, which provides a numerical solution to the filtering equations that is continuously benchmarked against the more naive moving-window regression method.

Both these steps will be reviewed in detail in section 3 of our report.

1.1.4 *The Data*

We consider the following financial instruments on the Canadian market (which can be traded for liquidities).

- CN Comdty
- CD Curncy

[1] There is no loss of generality in assuming that ε_t and ϵ_t are defined on the same probability space if we appeal to the Skorokhod representation theorem.

[2] Here m is the Lebesgue measure, and $\mathfrak{F}_t$ is the augmented and right-continuous filtration generated by R_t^I and R_t^f. The notation $\widetilde{D}$ means that two random elements have the same distribution.

- CL Comdty
- GC Comdty
- HG Comdty
- ES Index
- HC Index
- PT Index
- Risk free
- CNA
- BAA

We first try to understand the behaviour of the data we are working with. A natural starting point is to examine the elementary statistics of the data: mean, standard deviation, skewness, and excess kurtosis.

Table 1.1: Statistics of the index returns (1999 – 2017)

	Mean	Std	Skewness	Excess kurtosis
Index Return	6.31	2.23	−0.23	1.42

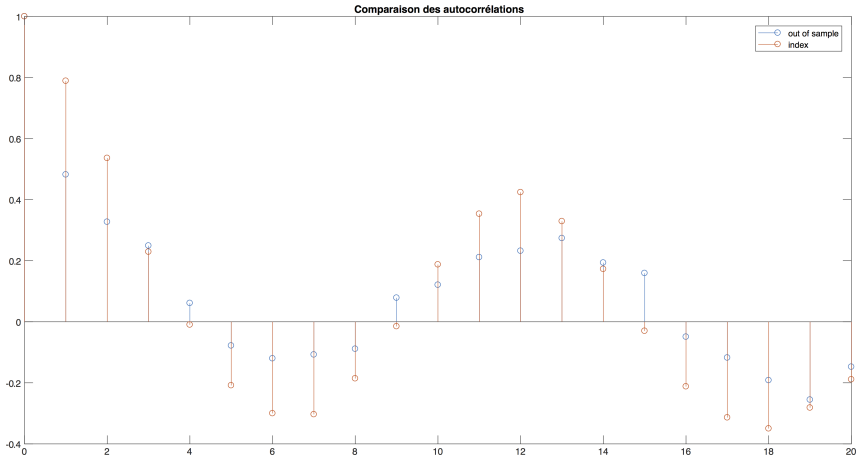
A (standard) second computation helps us visualize the correlation between any two instruments that could be included into our portfolio.

Table 1.2: The correlation matrix for the data.

	Index.Return	CN.Comdty	CD.Curncy	CL.Comdty	GC.Comdty	HG.Comdty	ES.Index	HC.Index	PT.Index	BAA
Index.Return	1	0	0.110	0.100	0	0.110	0.080	0.010	0.080	0.170
CN.Comdty	0	1	−0.250	−0.240	0.200	−0.230	−0.230	−0.140	−0.180	0.050
CD.Curncy	0.110	−0.250	1	0.430	0.380	0.520	0.550	0.450	0.500	0.010
CL.Comdty	0.100	−0.240	0.430	1	0.210	0.380	0.200	0.290	0.370	−0.110
GC.Comdty	0	0.200	0.380	0.210	1	0.290	0.020	0.180	0.200	0.020
HG.Comdty	0.110	−0.230	0.520	0.380	0.290	1	0.450	0.450	0.480	0.020
ES.Index	0.080	−0.230	0.550	0.200	0.020	0.450	1	0.480	0.770	0.180
HC.Index	0.010	−0.140	0.450	0.290	0.180	0.450	0.480	1	0.460	−0.040
PT.Index	0.080	−0.180	0.500	0.370	0.200	0.480	0.770	0.460	1	0.050
BAA	0.170	0.050	0.010	−0.110	0.020	0.020	0.180	−0.040	0.050	1

There is no significant autocorrelation of returns, with the notable exception of the last security (BAA), which exhibits very high persistence in its returns,

Then we look for factors exhibiting a seasonal trend similar to the one we observe in the housing market (where people buy more houses during the summer because moving is more difficult when there is snow on the ground). To do this we perform a statistical test for seasonality using autocorrelation metrics, in the hope of selecting financial assets in a way that replicates the seasonal trends observed in the housing index.



1.2 Solution 1: Optimal Factor Selection

1.2.1 *Factor selection using gradient boosting*

Gradient boosting with early stopping is a machine learning algorithm that selects the true subset of Π_{all} (the set of all our assets) based on their scores. This asset score is calculated as follows.

Definition 1 (Gradient Boosting). The goal of the learning algorithm in [2] is to identify the factors that significantly contribute to the model by minimizing the loss function

$$l(\Pi, \overline{w}, R^I, \lambda) = \sum_{t=1}^{d} (h_{\overline{w}}(\Pi_t) - R_t^I)^2 + \lambda |\overline{w}|, \tag{1.6}$$

where $h_{\overline{w}}(\Pi_t)$ is the value of the replicating portfolio at time t. [3] The parameter λ is called the learning rate.

The early stopping is performed by first creating pseudo-data through the introduction of various noise factors into the data. The gradient boosting algorithm is repeated until it begins to select the noise factors; then it stops. The reason for this is that the irrelevant factors, if any, must have the same predictive power as random noise and therefore should be disregarded in the same manner.

As is typical in machine learning circles, we select the learning rate λ by using leave k-out cross-validation (with $k = 5$). This algorithm is carried out by partitioning the data into three subsets: the first is used to calibrate λ, the second is used to test the accuracy of the calibrated λ, and the third is utilized as an "out-of-sample" verification of the fit.

In both numerical experiments we find that $\lambda = 0$ is optimal, which is a first indication that all the data are in fact important. Intuitively, this means that there is no *learning-rate*, and so it is not important to learn from the data. On the other hand, when running this particular machine learning algorithm, we find that only 8 financial instruments are deemed to be important.

Moreover, there is no apparent clustering, which indicates that no one instrument can be replaced by another instrument (or a group of other instruments). Instead it appears empirically that each item in the portfolio pulls its own weight, even if its impact is not large.

[3] Here $\lambda \geq 0$ is a tuning parameter that controls the aggressiveness of the factor selection algorithms by controlling the ℓ^1-penalty.

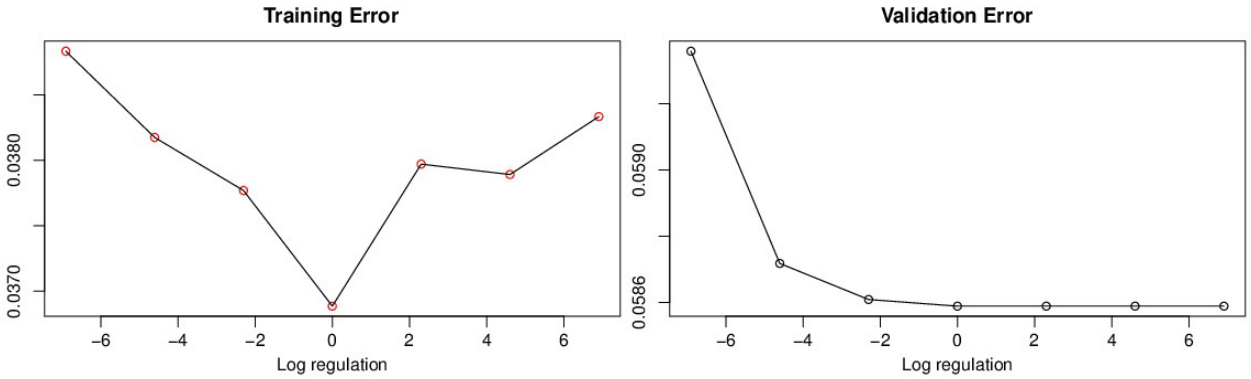
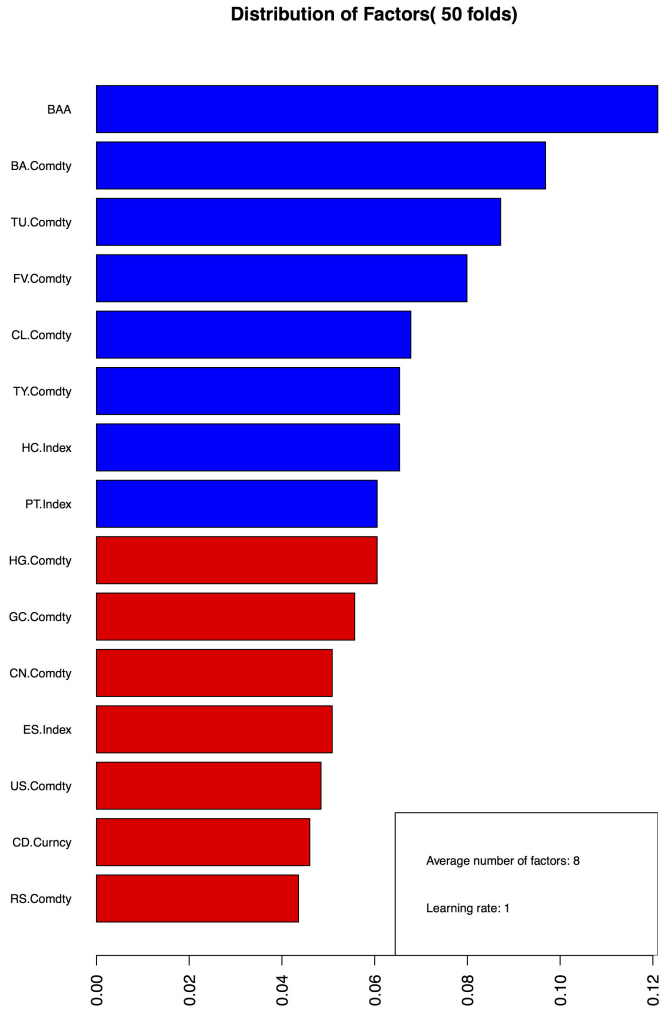
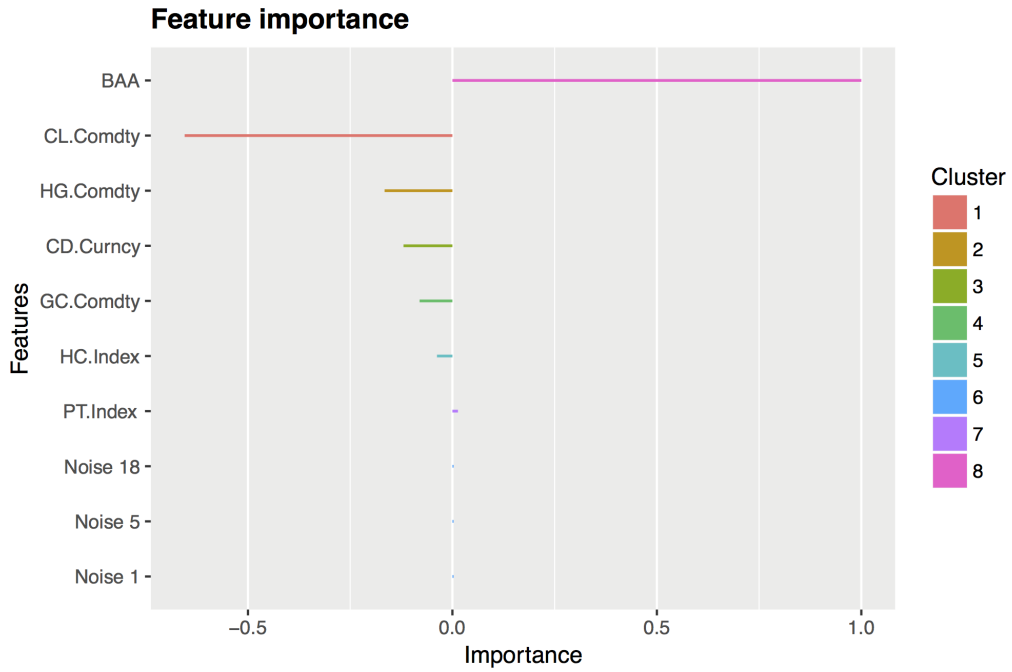


Fig. 1.3: Log segmentation





Upon examining the correlations of the selected factors, we naturally observe a similar interdependence pattern (if we use this algorithm).

Table 1.3: Correlation matrix of the selected factors.

	Index.Return	CL.Comdty	HG.Comdty	HC.Index	BAA	BA.Comdty	TU.Comdty	FV.Comdty	TY.Comdty
Index.Return	1	0.092	0.113	0.012	0.208	0.260	−0.139	−0.152	−0.148
CL.Comdty	0.092	1	0.379	0.286	−0.112	0.182	−0.121	−0.147	−0.151
HG.Comdty	0.113	0.379	1	0.455	0.014	0.097	−0.179	−0.204	−0.210
HC.Index	0.012	0.286	0.455	1	−0.034	−0.030	−0.115	−0.110	−0.103
BAA	0.208	−0.112	0.014	−0.034	1	−0.042	−0.048	−0.029	−0.052
BA.Comdty	0.260	0.182	0.097	−0.030	−0.042	1	0.017	−0.057	−0.078
TU.Comdty	−0.139	−0.121	−0.179	−0.115	−0.048	0.017	1	0.874	0.766
FV.Comdty	−0.152	−0.147	−0.204	−0.110	−0.029	−0.057	0.874	1	0.961
TY.Comdty	−0.148	−0.151	−0.210	−0.103	−0.052	−0.078	0.766	0.961	1

1.2.2 *Factor Selection Using the Adaptive Elastic Net*

We then compared this algorithm for selecting factors with a derivative of the LASSO introduced in [9], which has since become one of the most commonly used machine learning algorithms in finance, biology, and statistical learning.

For the Adaptive Elastic Net algorithm to be useful, the model must be linear and have a Gaussian distribution (amongst other assumptions). The most crucial assumption for factor selection is the linearity. Since the replication problem is a linear problem, the Adaptive Elastic Net algorithm is ideal for detecting factors.

Definition 2 (Adaptive Elastic Net). This algorithm performs variable selection for linear models with constant weights and estimates the weights $\widehat{w}$ as follows:

$$\widehat{w} \triangleq \left(1+\frac{\alpha}{d}\right)\left[\underset{w\in\mathbb{R}^d}{\arg\min}\left(\sum_{j=1}^{d}(h_{\overline{w}}(\bar{A}_t)-R_t^I)^2+\lambda\sum_{j=1}^{d}\hat{\eta}_j|\overline{w}|+\alpha(\overline{w})^2\right)\right],$$
$$\hat{\eta}_j \triangleq (|\widehat{w}_j(\text{Enet})|)^{-\gamma},$$

where $\widehat{w}_j(\text{Enet})$ is the vector of optimal weights defined in [11]. The Adaptive Elastic Net algorithm was shown in [12] to converge to the true set of explanatory variables under certain assumptions.

The parameter λ controls the model selection while the parameter α regularizes the effects of collinearity and shrinks the weights. Hence for the model selection problem the most important parameter to calibrate is λ, which is analogous to the learning rate in the "Gradient Boosting with Early Stopping" algorithm.

The Adaptive Elastic Net algorithm shines due to its Oracle Property, introduced and proven in [12, Theorem 3.2]. Roughly speaking, it states that the Adaptive Elastic Net estimator does precisely what we want since it will converge to the true subset of predictors. We now rephrase [12, Theorem 3.2] to apply it to our context. The interested reader is referred to the article for more details.

Theorem 1 (Oracle Property). *Let $w_t = (w_t^{true}, 0)$ be the partitioning of w_t into the true factor weights and irrelevant factor weights, respectively, and define*

$$\tilde{w}_t \triangleq \left(1+\frac{\alpha}{d}\right)\left[\underset{w\in\mathbb{R}^d}{\arg\min}\left(\sum_{j=1}^{d}(h_{\overline{w}}(\bar{A}_t)-R_t^I)^2+\lambda\sum_{j=1}^{d}\hat{\eta}_j|\overline{w}|+\alpha(\overline{w})^2\right)\right].$$

Then under the regularity conditions in [12], with probability tending to 1 as the sample size tends to ∞, the set

$$\Pi_{\text{AdaENET}} \triangleq \left\{w_t \in \Pi_{\text{all}} : \tilde{w}_t^i \neq 0\right\}$$

converges to the true set Π_{true}.

Recall that when we performed cross-validation on the "Gradient Boosting with Early Stopping" algorithm, in order to identify the optimal learning rate, we found it to be 0. This leads us to think that there is no learning, that is, every parameter is important. In a similar fashion, when we performed cross-validation on the factor selection parameter λ, we again found that it should be 0, which is a strong indication that every factor is meaningful.

Upon performing the estimation with the cross-validated λ set to 0, we indeed validate our empirical hypothesis that no factor effect equals 0. Even if the effect of a given factor is small, it is statistically meaningful and thus the factor cannot be replaced.

1.2.3 *Selected Factors*

Therefore we conclude our analysis by identifying all the factors in the portfolio as important. This means that the optimal portfolio Π_{true} will consist of the following assets.

- CN Comdty
- CD Curncy
- CL Comdty
- GC Comdty
- HG Comdty
- ES Index
- HC Index
- PT Index
- Risk free

- C.N.A
- BAA

We are now ready to estimate and predict the optimal weight process w_t.

1.3 Solution 2: Optimal Weighting

Our next and last goal is to solve the filtering problem. As we will see, however, the filtering problem is an infinite-dimensional problem and so it is not feasible to compute its optimal solution exactly. We can approximate it to an arbitrary precision using a three-fold method of regression, Kalman filtering, and particle filtering.

Each one of these methods helps to set up the initial parameters for the method following it in the sequence. This strategy increases the rate of convergence of the next (and more sophisticated) method; thus it greatly reduces the computation time.

1.3.1 *Rolling-Window Regression*

We segment the data into blocks of 24 months and perform a regression of the excess returns against the returns of the Teranet – National Bank index. We minimize the sum of the squared errors of R_t^I against $w_t^T R_T^f$ within each block, assuming Gaussian errors. More specifically, the regression framework minimizes the Gaussian idiosyncratic error ϵ_t defined by

$$\epsilon_t \triangleq R_t^I - \sum_{i=1}^{d} w_t^i R_t^i$$

of the following static linear problem

$$\widehat{w}_t \triangleq \underset{w\in\mathbb{R}^d}{\arg\min} \left[\left(R_t^I - \sum_{i=1}^{d} w_t^i R_t^i \right)^2 \right].$$

Here $t \in \{0 < t_1 < \cdots < t_K\}$ represents one of the K different 24-month time windows. We will use $\widehat{w}_0$ to initialize the Kalman-Bucy filter in the next step.

Remark 1. Although the rolling-window regression is fast, it performs much more poorly than other more sophisticated techniques requesting far fewer assumptions and making use of a richer structure (present in the replication problem). The rolling-window regression, however, will provide a benchmark and a fast way of initializing these techniques.

We now give a brief overview of the filtering problem.

1.3.2 *Stochastic Filtering*

We first recall the probabilistic setup of Problem 2. Let $(\Omega, \mathfrak{F}, \mathbb{P})$ be a complete probability space and let R_t^I, R_t^f, ϵ_t, and ε_t be defined on it. Moreover, let $\mathfrak{F}_t$ denote the right-continuous filtration representing the observable information in the market at time t. This filtration is defined as

$$\mathfrak{F}_t \triangleq \bigcap_{\tilde{t}>t} \left[\sigma\left(\{R_s^I, R_s^f\}_{0\leq s<\tilde{t}}\right) \vee \mathfrak{N}\right], \tag{1.7}$$

where $\mathfrak{N}$ are all the null-sets in $\mathfrak{F}$.

The minimizer of the MSE loss function is the conditional expectation

$$\tilde{w}_t \triangleq \mathbb{E}\left[w_t \mid \mathfrak{F}_t\right]. \tag{1.8}$$

Equation (1.8), however, only defines $\tilde{w}_t$ up to a set of $(\mathbb{P}\otimes m)$-measure 0. Hence it does not actually provide a description of the path that the process $\tilde{w}_t$ takes, since this would require specifying all its values on all the null-sets (left unspecified by Equation (1.7)). Thus for an uncountably infinite number of time points t in $\mathbb{R}^+$, we have no information on the behaviour of $\tilde{w}_t$. This means that the distribution of $\tilde{w}_t$ cannot be specified.

This requires us to specify a good version of $\tilde{w}_t$, preferably one with regular paths, i.e., paths that are at least right-continuous. Such a version always exists for cádlág semi-martingales and is the object of the following result.

Theorem 2 (Optional Projection). *Let X_t be an $\mathfrak{F}_t$-measurable semi-martingale with $\mathbb{P}$-a.s. cádlág paths defined on $(\Omega, \mathfrak{F}, \mathbb{P})$ such that for every $\mathfrak{F}_\star$-stopping time $\tau \geq 0$, $\mathbb{E}\left[1_{\tau<\infty}|X_\tau| \mid \mathfrak{F}_t\right]$ is finite. Then there exists a unique process X_t^{o}, up to evanescence, called the optional projection of X_t onto $\mathfrak{F}_t$, satisfying*

$$1_{\tau<\infty}X_t^{\mathrm{o}} = \mathbb{E}\left[1_{\tau<\infty}X_\tau \mid \mathfrak{F}_t\right] \tag{1.9}$$

$\mathbb{P}$-almost surely for every $\mathfrak{F}_\star$-stopping time τ.

In rigorous terms, the stochastic filtering process is concerned with finding the distribution of $\mathbb{E}[w_t|\mathfrak{F}_t]^{\mathrm{o}}$, which we denote by π_t. The dynamics followed by the measure-valued process π_t is known in general cases and given by the Kallianpur – Striebel formula.

Theorem 3 ((1.10)). *Under certain integrability conditions described in* [3, Theorem 22.1.9], *assume that w_t follows a semi-martingale of the form*

$$w_t = w_0 + \int_0^t \beta_s \,\mathrm{d}s + N_s,$$

where β_t is a predictable process with respect to the augmented-right continuous filtration generated by N_t and N_t is a square-integrable martingale independent of w_t. Then π_t solves the stochastic differential equation

$$\begin{aligned}\pi_t = \pi_0 + \int_0^t \beta_s^{\mathrm{o}}\,\mathrm{d}s + \int_0^t Q^{-1}\Bigg(&\left[\pi_s\left(\left(w_s^T R_s^f\right) + \left(1 - \sum_{i=1}^d w_{i,s}\right) r_s\right)\right]^{\mathrm{o}} \\ &- \pi_s\left(\left(w_s^T R_s^f\right) + \left(1 - \sum_{i=1}^d w_{i,s}\right) r_s\right)^{\mathrm{o}}(\omega, s, Y)\Bigg)^T \mathrm{d}V_s,\end{aligned} \tag{1.10}$$

where V_t is a martingale [4] *with respect to $\mathfrak{F}_t$ defined by*

[4] The process V_t is called the innovations process, and represents the *new* information introduced into the system at time t.

$$V_t = \int_0^t Q^{-1}\,\mathrm{d}Y_s - \int_0^t Q^{-1}\left[\left(w_s^T R_s^f\right) + \left(1 - \sum_{i=1}^{d} w_{i,s}\right) r_s\right]\mathrm{d}s.$$

Unfortunately, solving for π_t explicitly is still infeasible since the process is infinite-dimensional. If we make a few simplifying assumptions, however, we arrive at an optimal closed-form solution for the simplified dynamics of w_t. We now describe this simplified problem.

1.3.3 *Kalman – Bucy Filter*

Let us make the following assumptions on w_t.

- w_0 is Gaussian.
- The stochastic differential equation (w_t, R_t^I) has a unique $\mathfrak{F}_1$-adapted solution.
- Q is invertible.
- The map

$$t \mapsto A_t \oplus C_t \oplus \left[\left(w_t^T R_t^f\right) + \left(1 - \sum_{i=1}^{d} w_{i,t}\right) r_t\right]$$

  is continuous on $[0, \infty)$.

Under these simplifying assumptions we know that w_t is Gaussian. Therefore, at any time t, its distribution is entirely specified by its first two moments, namely its mean and covariance matrix. Thus we know where the dimension reduction takes place. We may now compute explicitly the optimal filter for the solution of this linear Gaussian problem.

Theorem 4 (Kalman – Bucy Filter). *Let $\widehat{w}_t$ and $\widehat{\Sigma}_t$ denote the conditional mean and variance of w_t, respectively. Then $\widehat{w}_t$ and $\widehat{\Sigma}_t$ solve the following stochastic differential equations:*

$$\begin{aligned} \widehat{w}_t &= \widehat{w}_0 + \int_0^t A_s \widehat{w}_s\,\mathrm{d}s + \int_0^t \widehat{\Sigma}_s \left(Q^{-1} R_t^E\right)\mathrm{d}V_s, \\ \frac{\widehat{\Sigma}_t}{\mathrm{d}t} &= A_t \widehat{\Sigma}_t \widehat{\Sigma}_t A_t - \widehat{\Sigma}_t R_T^E \left(QQ^T\right)^{-1} R_T^E \widehat{\Sigma}_t + C_t C_t^T, \end{aligned} \tag{1.11}$$

where R_t^E denotes the returns on the portfolio. Moreover V_t is a $\mathfrak{F}_t$-Brownian motion.

The assumptions for the Kalman-Bucy filter may be too stringent: it could happen that they do not reflect the "real" problem. In particular the distributions of the "noises" (ϵ and especially ε) may not be Gaussian.

Remark 2. The Kalman filter depends upon our initial estimates of w_0, and a poor estimate for $\widehat{w}_0$ may have a negative impact on the performance of the Kalman filter. This is why the Kalman – Bucy filter is preceded by a more primitive and faster technique, the rolling-window regression described above.

1.3.4 *Particle Filtering*

A more realistic and time-consuming way to compute π_t replaces the convenient closed-form solution of the Kalman – Bucy filter by a filter that is arbitrarily close to the solution. This sequential Monte Carlo scheme, called particle filtering, uses the law of large numbers to approximate the distribution of X_t sequentially. During each approximation step a number of particles are generated from the current position according to

the assumed dynamics of w_t. Then the N most relevant particles are retained and the distribution is mutated according to the new law approximated by the Monte Carlo step.

In machine learning this type of algorithm is called a genetic algorithm, since it starts from an initial guess and lets the data mutate it over time by removing the non-surviving particles. We now present the algorithm following the text of [7, Algorithm 9.6.2].

Theorem 5 (Auxiliary Sampling/Importance Resampling(ASIR) Particle Filter). *Make an initial guess for the distribution of w_t; call it η_0. Let N be a positive integer. Generate particles $\{z_0^{(k)}\}_{k=1}^N$ from the initial distribution η_0. Call this population of particles*

$$\mathcal{P}_i = \left\{ z_i^{(k)} : k \in \{1, \dots, N\} \right\},$$

which is obtained at step i from the probability distribution π_i^N (where $\pi_0^N \triangleq \eta_0$). Then repeat the following steps.

(1) *For $k \in \{1, \dots, N\}$, compute*

$$\begin{aligned} \tilde{v}_{i+1}^{(k)} &\triangleq g_{1,i+1}(k, z_i^{(k)}) \pi_i^{(k)}, \\ \tilde{\pi}_{i+1}^{(k)} &\triangleq \frac{\tilde{v}_{i+1}^{(k)}}{\sum_{j=1}^N \tilde{v}_{i+1}^{(j)}}. \end{aligned} \tag{1.12}$$

(2) *Choose N values $\tilde{z}_{i+1}^{(k)}$, $k \in \{1, \dots, N\}$, from $\mathcal{P}_i$ with the probability of choosing $\tilde{z}_i^{(k)}$ being $\tilde{\pi}_i^{(}k)$.*

(3) *Generate $\{z_{i+1}^{(k)} \widetilde{D}\, g_{2,i+1}\left(\cdot \mid \tilde{z}_{i+1}^{(k)}\right)\}_{k=1}^N$.*

(4) *Define*

$$\begin{aligned} v_{i+1}^{(k)} &\triangleq \frac{p_{Y_{i+1}|Z_{i+1}}\left(y_{i+1}|z_{i+1}^{(k)}\right) p_{Z_{i+1}|Z_i}\left(z_{i+1}^{(k)}|\tilde{z}_{i+1}^{(k)}\right)}{g_{1,i+1}\left(k, \tilde{z}_{i+1}^{(k)}\right) g_{2,i+1}\left(z_{i+1}^{(k)}|\tilde{z}_{i+1}^{(k)}\right)} \\ \pi_{i+1}^{(k)} &\triangleq \frac{v_{i+1}^{(k)}}{\sum_{j=1}^N v_{i+1}^{(j)}}. \end{aligned} \tag{1.13}$$

(5) *The empirical distribution η_{i+1} of the particles $\left\{z_{i+1}^{(k)}\right\}_{k=1}^N$ is then defined by*

$$\eta_{i+1}(\phi) \triangleq \sum_{k=1}^N \phi\left(z_{i+1}^{(k)}\right) \pi_{i+1}^{(k)},$$

for any twice-continuously differentiable function ϕ.

Then given regularity conditions (which can for example be found in [1, Section 10]), as N tends to ∞ the process π_i converges to $\pi_i(\phi)$ in distribution.

Remark 3. A crucial point in the particle filtering algorithm is that the initial distribution η_0 must be chosen by the user. Ultimately the effects of η_0 are blurred out as the algorithm converges to the true filtering solution (with precision given by N). In practice, however, the speed of convergence can be largely impacted by the choice of the initial η_0.

1.4 Implementation

1.4.1 *Rolling-Window Regression*

The in-sample fit is reasonable (as expected). When predicting out of sample, however, the rolling-window regression does quite poorly. Nevertheless it provides an excellent initial guess for the next step and a benchmark to verify the performance of our algorithm.

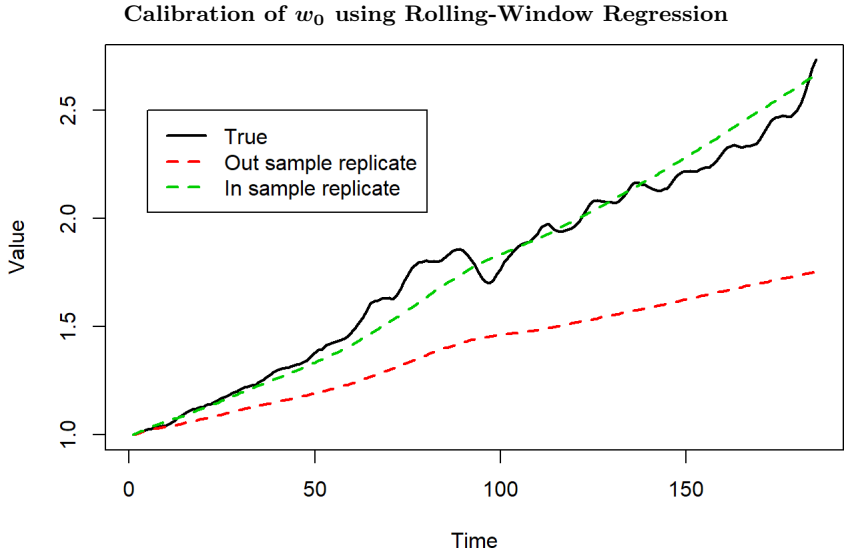


Fig. 1.4: Performance of the 24-month rolling-window regression.

1.4.2 *Kalman Filtering*

We assume that the dynamics of the signal process w_t follows a Gaussian Ornstein-Uhlenbeck process, which solves the stochastic differential equation

$$w_t = w_0 + \int_0^t \phi w_s \, \mathrm{d}s + \int_0^t Q \, \mathrm{d}W_s, \tag{1.14}$$

where w_0 is a Gaussian random variable estimated by the rolling-window regression and ϕ is a constant diagonal matrix.

Upon implementing the Kalman filter with an initial estimate of w_0 (using the rolling-window regression), we immediately observe a vastly improved performance (with respect to the preceding benchmark algorithm). The in-sample performance of the Kalman filter is visually indistinguishable from the data.

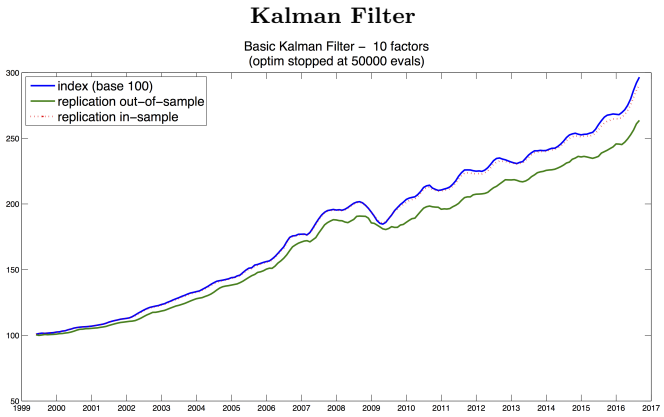


Fig. 1.5: Comparison of the in-sample and out-of-sample performances of the implemented Kalman filter.

Figure section 1.4.2 underlines the vast improvement achieved through Kalman filtering when compared with regression methods. Further gains can still be made by reducing the out-of-sample tracking error.

1.4.3 *ASIR Particle Filter*

We implement the Auxiliary Sampling/Importance Resampling Particle Filter of theorem 5. Our choice of initial sampling distribution is the Gaussian distribution obtained from the previous Kalman – Bucy filtering step, with a rolling-window regression. The result is an extremely accurate tracking of the Teranet – National Bank housing index returns, as illustrated by the following figure.

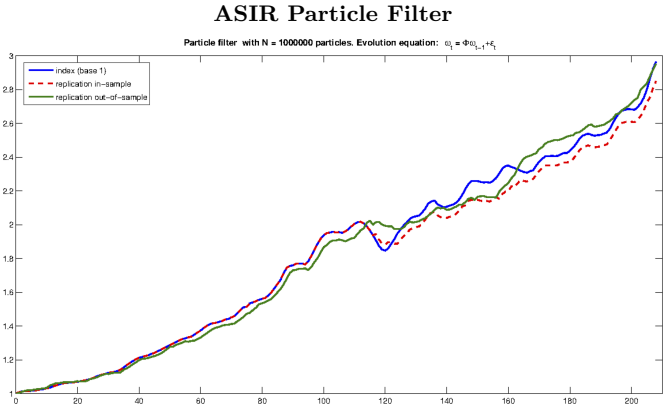
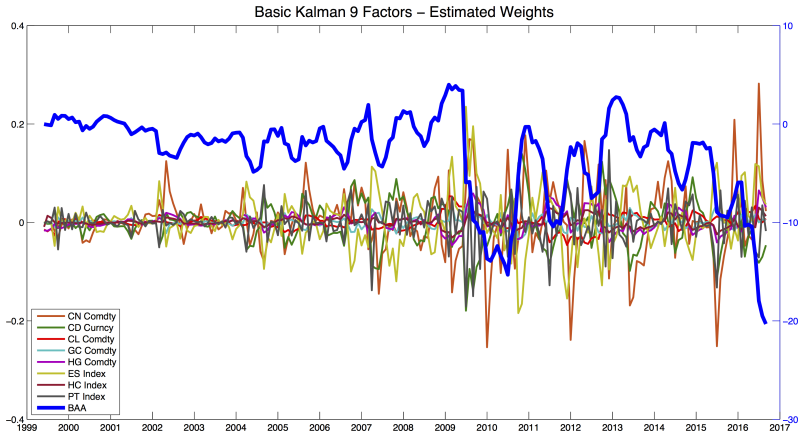


Fig. 1.6: Comparison of the in-sample and out-of-sample performances of the implemented ASIR Particle Filter.

1.4.4 *Risk-Controlled ASIR Particle Filter*

There is one risk-theoretic issue to consider when building this replicating portfolio. Although all the factors are deemed to be important, the behaviour of the replicating portfolio is heavily tied to BAA. This is illustrated in the following figure, which underlines the larger weight being placed on BAA.



Although most weights vary within small intervals, some have a negative value for ϕ, implying alternation of long/short positions. The need to rebalance the portfolio between these positions may be costly. Another risk-theoretical drawback of this replicating portfolio is that one security is leveraged up to 20 times.

To remedy these potential concerns we propose implementing the particle filtering algorithm but with a resampling step, where the particles are continuously resampled until a trajectory is taken that has low leverage risk and rebalancing cost. This may be carried out at the cost of an increase in computation time and a mild increase in tracking error: this approach, however, will decrease the risk.

The numerical performance of the particle filter is further examined in the following tables.

Table 1.4: In-sample statistics.

Portfolio	TE	Pearson Corr	Kendall Corr	Mean	Std	Skew	Excess kurt
Target	0	1	1	6.3130	2.2388	−0.2256	1.4168
Regression	6.0823	0.1331	0.0472	0.0053	0.0011	0.5543	2.0818
Kalman 1	1.3391	0.9875	0.8918	6.1752	2.0605	−0.1797	1.5414
Kalman 2	6.74e-10	1	1	6.2954	2.2430	−0.2191	1.4029
Kalman 3	7.84e-10	0.5763	0.3946	6.2738	1.7355	−0.2416	1.0534
Particle (Unc.)	4.2000	0.8450	0.8568	6.0789	2.0887	−0.1913	1.4587
Particle (Con.)	3.3496	0.9056	0.9270	5.7817	2.1457	−0.3291	1.5756

Here the in-sample and out-of-sample estimation and predictive performance of the ASIR particle filter are quantified using standard statistical metrics. More precisely, the performance of each method is described using the tracking error, the Pearson correlation, the Kendall correlation, the mean, the standard deviation, the skewness, and the excess kurtosis statistics.

Upon examining the out-of-sample performance of the risk-controlled ASIR particle filter, we see that not only is the tracking error surprisingly lower and the signal more highly correlated with the index movements, but also that it is not excessively leveraged like the weights generated through the classical unrestricted ASIR particle filter.

Table 1.5: Out-of-sample statistics.

Portfolio	TE	Pearson Corr	Kendall Corr	Mean	Std	Skew	Excess kurt
Target	0	1	1	6.3130	2.2388	−0.2256	1.4168
Regression	9.8850	0.1284	0.0132	0.0031	0.0011	0.5682	2.1050
Kalman 1	6.0297	0.6587	0.4420	5.6274	1.8757	−0.1584	1.7168
Kalman 2	6.5171	0.5763	0.3946	6.2738	2.2430	1.7355	1.0534
Kalman 3	6.5033	0.5777	0.3937	6.2792	1.7307	0.2609	1.0423
Particle (Unc.)	8.0168	0.3685	0.2484	6.2779	1.8499	0.2040	0.4957
Particle (Con.)	7.2842	0.5038	0.3306	6.6414	1.9172	0.0092	1.2384

It is worth mentioning that we also implemented two other Kalman filters, assuming different dynamics for the process w_t. More precisely, we assume the following three models for the dynamics of the signal process w_t.

(1) $w_t = w_0 + \int_0^t \phi w_s \, \mathrm{d}s + \int_0^t Q \, \mathrm{d}W_s$
(2) $w_t = w_0 + \int_0^t w_s \, \mathrm{d}s + \int_0^t Q \, \mathrm{d}W_s$
(3) $w_t = w_0 + \int_0^t Q \, \mathrm{d}W_s$

The performance of the Kalman – Bucy filter under these assumptions is summarized in the previous two tables.

1.5 Conclusion

Using a combination of machine learning methods, regression, and various stochastic filtering methods, we were able to construct a replicating portfolio for the Teranet – National Bank housing index that is not over-leveraged and predicts the index extremely closely. Moreover our machine learning algorithms for selecting the optimal factors are robust with respect to modelling errors and detect the true factors both in theory and in practice.

The combination of our machine learning and stochastic filtering methods significantly improved the performance of our index replication (when compared to the benchmark method of rolling-window regression with factor selection based on machine learning). In a similar fashion our particle-filtering-based methods performed noticeably better than the simpler Kalman method. This allowed us to build an asset that exhibits the same behaviour as the Teranet – National Bank Canadian housing index but in which any investor can invest directly. Hence the benefits of the long-term stable growth of the Canadian housing market can (for the first time ever) be captured by a portfolio of liquidly traded financial assets. Moreover this portfolio is stable and not over leveraged, making it an altogether safe and stable investment.

We thank the National Bank of Canada for providing us with an exciting problem and the corresponding data.

References

1. A. Bain and D. Crisan. *Fundamentals of Stochastic Filtering*, volume 60 of *Stochastic Modelling and Applied Probability*. Springer, New York, 2009.
2. P. Bühlmann and T. Hothorn. Boosting algorithms: regularization, prediction and model fitting. *Statist. Sci.*, 22(4):477–505, 2007.
3. S. N. Cohen and R. J. Elliott. *Stochastic Calculus and Applications*. Probab. Appl. Birkhäuser, Basel, 2nd edition, 2015.
4. R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Trans. ASME Ser. D. J. Basic Engrg.*, 83:95–108, 1961.

5. R. S. Liptser and A. N. Shiryaev. *Statistics of Random Processes. I. General Theory*, volume 5 of *Appl. Math. (N. Y.)*. Springer, Berlin, 2001.
6. M. K. Pitt. Smooth particle filters for likelihood evaluation and maximisation. Warwick Economic Research Paper 651, Department of Economics, University of Warwick, 2002.
7. B. Rémillard. *Statistical Methods for Financial Engineering*. CRC Press, Boca Raton, FL, 2013.
8. B. Rémillard, B. Nasri, and M. Ben Abdellatif. Replication methods for financial indexes. Cahier du GERAD G-2017-58, GERAD, 2017.
9. R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.
10. T. Zhang and B. Yu. Boosting with early stopping: convergence and consistency. *Ann. Statist.*, 33(4):1538–1579, 2005.
11. H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(2):301–320, 2005.
12. H. Zou and H. H. Zhang. On the adaptive elastic-net with a diverging number of parameters. *Ann. Statist.*, 37(4):1733–1751, 2009.

2
Inspection Route Optimization
Problem proposed by The Co-operators

Bernard Gendron, Thibaut Vidal, Steven Lamontagne, Belhal Karimi, David Hagenimana, Jalal Ahammad, Monuara Gagum, Dena Kazerani, Ilya Chugunov, Maikel Geagea, Mickael Albertus, Bora Yongacoglu, Mathieu Giguère, and Jean-Michel Plante

2.1 Introduction

This report is a summary of the work carried out during the Eighth Montreal Industrial Problem Solving Workshop, held at the Centre de Recherches Mathématiques. Our team was supervised by Bernard Gendron and Thibaut Vidal. The student participants for this group were Steven Lamontagne, Belhal Karimi, David Hagenimana, Jalal Ahammad, Monuara Gagum, Dena Kazerani, Ilya Chugunov, Maikel Geagea, Mickael Albertus, and Bora Yongacoglu. We were assisted by the representatives from the company, Mathieu Giguère and Jean-Michel Plante.

The Co-operators is a Canadian insurance and financial services company providing personal and casualty, life, and investment insurance to individuals as well as to companies. As part of their commercial insurance services, they send inspectors to visit buildings in order to evaluate or reevaluate the risk associated with the business. The Co- operators insure over 5000 buildings, and with comparatively few inspectors they must choose which buildings to visit during a given year. Each location is assigned a risk score between 0 and 100

Bernard Gendron
CIRRELT & Université de Montréal

Thibaut Vidal
Pontificia Universidade Catòlica do Rio de Janeiro

Steven Lamontagne
Université de Montréal

Belhal Karimi · Dena Kazerani
École Polytechnique (France)

David Hagenimana
University of Nairobi

Jalal Ahammad · Monuara Gagum
Memorial University of Newfoundland

Ilya Chugunov
UC Berkeley

Maikel Geagea
Polytechnique Montréal

Mickael Albertus
Université Paul Sabatier Toulouse

Bora Yongacoglu
Queen's University

Mathieu Giguère · Jean-Michel Plante
The Co-operators

(where 100 stands for the highest risk): these scores are used for helping choose the buildings that will be visited. Finally the following constraints must be respected when constructing the inspections schedule.

(1) Each inspector works a specified number of weeks during the year, which depends upon the inspector.
(2) Inspectors must return home for weekends.
(3) There are mandatory buildings (not necessarily high-risk ones), i.e., buildings that must be included in the inspections schedule.

In order to solve the problem, the analysts at The Co-operators use two methods. The first method consists of assigning each inspector to a territory (subdivided into provinces) based on the location of the inspector's home. Within each territory, the buildings are then chosen in descending order of risk score.

The second approach used by the company is based on a decomposition similar to that used in the first method. Instead of assigning *provinces* to specific inspectors, the second approach assigns *cities* to inspectors. The locations (i.e., buildings) are then assigned to cities by minimizing the distance between building and city. The company then solves separate optimization problems for each of the cities, with the objectives being to maximize the total risk score in the selected schedule and to minimize the total distance travelled by the inspector. The resulting minimization problem is a kind of modified traveling salesman problem, in which a single person completes a *tour* of the locations (i.e., the inspector visits each location once and only once) and the total distance travelled is minimal.

Since there are two objective functions, we can picture the set of "best" solutions as a curve of combinations of total risk score and distance travelled. Hence bounding one of the two objective functions allows us to optimize the other objective. While the results of this approach are better than those obtained with the previous approach, it has two flaws. First, the decomposition into cities is carried out arbitrarily and restricts the locations to a single region. Second, while there is a constraint on the *distance*, the real constraint should involve the *time*. The distance between the locations can be used to estimate the travel time, but it does not take into account the time it takes to inspect the locations.

The first step undertaken by the team during the workshop was to provide a mathematical formulation of the problem. The objective is simple: to maximize the total risk score of the buildings visited during a given year. Rather than having two objective functions, the objective function corresponding to the distance travelled is replaced by time constraints:

- Daily constraints: every inspector makes a tour starting and ending at a *hotel* (here, and elsewhere in the document, "hotel" refers to any place where the inspector may spend the night, be it a hotel, a motel, his home, etc.). The total duration of the tour (travel time + inspection time) may not exceed 7 hours;
- Weekly constraints: every inspector works up to 5 days (i.e., makes up to 5 tours) before going back to his home;
- Yearly constraints: every inspector works (about) 46 weeks in any given year.

The natural decomposition of constraints leads to a decomposition of the problem. First, we can construct a separate optimization problem for each week. The solutions of the weekly problems are then "merged" into a solution for the annual problem. To achieve these tasks, the team of students was split into three sub-teams, whose tasks were (respectively):

1. To extract and analyze the data provided by the company;
2. To develop and implement models for the annual problem and for the weekly problem;
3. To develop and implement heuristic methods for the weekly problem.

2.2 Data Extraction

The representatives from The Co-operators provided the team with data from the company. A data file included, for each of the 5000 prospective locations to be visited, the distance to each of the other locations as well as the distance to each of the possible hotels (this information was stored in a matrix where the

(i, j) component indicates the distance from building i to building j). The data file also included the time necessary to inspect a particular building, the information that it was mandatory to visit the building (or not), and the risk score associated with it. In order to implement the time-related constraints for the weekly problem, the first task of the data extraction team was to convert the distances stored in the data file into values for elapsed time. Because of privacy concerns, the team could not access the exact location of each building: therefore the distances were converted uniformly assuming a travelling speed of 60 km/h. The next task of the data extraction team was to reduce the size of the problem by eliminating some buildings and assigning the locations to potential cities (each "city" being a potential hotel). For instance, if the time to travel from location i to hotel j was at least 3.5 hours, then it could not possibly be advantageous to include location i within the city defined by hotel j.

2.3 Annual Problem

2.3.1 *Annual Problem: Assumptions and Data*

We use the following notation for the annual problem.

Notation	
Symbol	**Meaning**
I	Set of weekly solutions
J	Set of locations
K	Set of inspectors
M	Set of mandatory locations
δ_{ij}	equals 1 if location $j \in J$ belongs to weekly solution $i \in I$, 0 otherwise
θ_{ik}	equals 1 if inspector $k \in K$ is assigned to weekly solution $i \in I$, 0 otherwise
p_i	Profit (total risk score) of weekly solution $i \in I$
W_k	Annual number of weeks worked by inspector $k \in K$

J, K, M, and W are provided by the company, whereas I, δ, θ, and p are to be constructed as part of the solution method.

2.3.2 *Annual Problem: Model*

We define the variable x_i as follows: $x_i = 1$ if weekly plan $i \in I$ is selected and $x_i = 0$ otherwise.

$$\max \sum_{i \in I} p_i x_i \tag{2.1}$$

$$\sum_{i \in I} \delta_{ij} x_i \leq 1 \quad j \in J \tag{2.2}$$

$$\sum_{i \in I} \theta_{ik} x_i \leq W_k \quad k \in K \tag{2.3}$$

$$\sum_{i \in I} \delta_{ij} x_i \geq 1 \quad j \in M \tag{2.4}$$

$$x_i \in \{0, 1\} \quad i \in I \tag{2.5}$$

The objective (2.1) is to maximize the total risk score of buildings visited during the year. Inequalities (2.2) enforce the constraint that each building may be visited once. Constraints (2.3) enforce a limit on the number of weeks during which an inspector works (in the year). Constraints (2.4) ensure that all mandatory buildings are visited. Since the mandatory buildings do not necessarily have high risk scores, there is no guarantee that they would be visited if these constraints were not included into the model.

The objective (2.1), together with constraints (2.2) and (2.5), is a well-known NP-hard problem, referred to as the *set packing problem* (see for instance [2]). If additional constraints (such as (2.3) and (2.4)) are added, the problem is referred to as a *constrained set packing problem*.

2.3.3 *Annual Problem: Solution Methods*

There are exact methods, called *branch-and-price* methods, for solving this kind of problem: they involve a combination of *column generation* and *branch-and-bound* methods. An efficient implementation of these methods, however, requires a significant programming effort. Heuristic methods generating high-quality (though not always optimal) results can be developed fairly quickly.

The method used to obtain a solution for the annual problem is based on a simple principle: assume that we have a set with every possible *weekly schedule* (a weekly schedule being a set of 5 daily tours assigned to a single inspector). The task of finding the optimal yearly schedule is then to select the optimal subset from our collection. In practice, finding the set of all possible weekly schedules and selecting the best combination of weekly schedules is computationally intractable. During the workshop the team goal was to construct a set of weekly schedules (see Sections 4 and 5) and select an optimal subset among them (a similar approach is described in [3], p. 330).

From the team working on the weekly solutions, the team working on the annual problem would receive a file containing a set of weekly schedules. Each weekly schedule i in that file is a list that includes:

- $k \in K$ with $\theta_{ik} = 1$ (i.e., the inspector assigned to that weekly schedule);
- p_i, i.e., the total profit for that weekly schedule;
- the set of $j \in J$ with $\delta_{ij} = 1$ (i.e., the set of locations comprised in that weekly schedule).

The list of weekly schedules is denoted by I. The constrained set packing problem (2.1)–(2.5) was implemented in Python; the *branch-and-bound* method in the Gurobi solver was used to find the optimal set of weekly solutions. Over the course of the workshop, the model was tested using schedules randomly generated by a Matlab script. This was carried out in order to validate the model, as well as to observe computational times for varying numbers of weekly schedules. Because of feasibility errors with randomly generated data, a second script was developed that weakened constraints (2.2) at the cost of an increase in computing time. Thanks to the tests, it was found that the computing times for models involving more than 10,000 weekly schedules were very large (indeed some models could not be solved). The final version of the script was thus based on that number of weekly schedules, with the possibility of developing later a script that could solve instances with more than 10,000 weekly schedules by subdividing the list of schedules into subgroups of 10,000 of fewer.

2.4 Model for the Weekly Problem

2.4.1 *Formulation of the Weekly Problem*

The goal of the weekly problem is to create a set of $d \leq 5$ daily tours such that the overall profit (risk score) is maximized and the daily time limit (of 7 hours) is respected. This is a variant of the *vehicle routing problem* (VRP) known as the *team orienteering problem* (TOP), named after a game in which teams of

players must bring flags of varying worth back to the starting point within a fixed time limit (see [4], p.3). Once a TOP is solved (through mathematical programming or by using heuristic methods), the schedule data is processed by completing the week (i.e., including $5-d$ more days into it) and then assigning each weekly schedule to nearby inspectors. Since there is no requirement (at this stage) that each weekly schedule has a unique inspector (this condition is subsequently enforced by constraints (2.3) in the annual problem), weekly schedules can be generated quickly by simply changing which (feasible) inspector is assigned to it.

2.4.2 *TOP: Assumptions and Data*

We wish to construct d tours for each *city*. We use the following notation for this problem.

Notation	
Symbol	**Meaning**
$1,\ldots,n$	Locations that can be visited
$0,\ n+1$	Hotel (duplicated)
V	$\{0,1,\ldots,n,n+1\}$, the set of nodes
A	$\{(i,j)\|i\in V^{n+1}, j\in V^0\}$, the set of arcs, where $V^i = V\setminus\{i\}$
G	(V,A), the resulting directed graph
p_i	Profit at location $i\in V$ $(p_0 = p_{n+1} = 0)$
t_{ij}	Time for arc $(i,j)\in A$, including travel time from i to j and time to inspect j $(t_{0,n+1} = 0)$
T	Daily time limit (currently 7 hours, but could vary based on the day and the inspector)

2.4.3 *TOP: Model*

We introduce the following variables: y_i^l, where $y_i^l = 1$ if $i\in V$ is selected in tour l and $y_i^l = 0$ otherwise, and x_{ij}^l, where $x_{ij}^l = 1$ if $(i,j)\in A$ is selected in tour l and $x_{ij}^l = 0$ otherwise.

$$\max \sum_{i\in V}\sum_{l=1}^{d} p_i y_i^l \tag{2.6}$$

$$\sum_{j\in V^i} x_{ij}^l = y_i^l \quad i\in V^{n+1}, l = 1,\ldots,d \tag{2.7}$$

$$\sum_{j\in V^i} x_{ji}^l = y_i^l \quad i\in V^0, l = 1,\ldots,d \tag{2.8}$$

$$\sum_{l=1}^{d} y_i^l \le 1 \quad i\in V\setminus\{0,n+1\} \tag{2.9}$$

$$y_0^l = y_{n+1}^l = 1 \quad l = 1,\ldots,d \tag{2.10}$$

$$\sum_{(i,j)\in A} t_{ij}x_{ij}^l \le T \quad l = 1,\ldots,d \tag{2.11}$$

$$x_{0i}^l \le x_{0j}^{l+1} \quad i\in V^0, j\in V^0, i\le j, l = 1,\ldots,d-1 \tag{2.12}$$

$$y_i^l\in\{0,1\} \quad i\in V, l = 1,\ldots,d, \quad x_{ij}^l\in\{0,1\} \quad (i,j)\in A, l = 1,\ldots,d \tag{2.13}$$

The objective (2.6) is to maximize the total risk score visited during the week. Equations (2.7) and (2.8) refer to the condition that each location may only be visited once. In particular, equations (2.7) mean that, for every tour l, the number of arcs *leaving* node i is equal to 1 if node i is visited and 0 otherwise. Similarly,

equations (2.8) mean that, for every tour l, the number of arcs *entering* node i is equal to 1 if node i is visited and 0 otherwise. Inequalities (2.9) enforce the condition that every location (other than the hotel) may only be visited once throughout the week. Constraints (2.10) enforce the condition that the hotel must be included at the beginning and end of the tour. Constraints (2.11) mean that, for each tour l, the daily time limit must be respected.

Inequalities (2.12) are a kind of *symmetry-breaking constraints*. They are not required to model the problem but they speed up the solution method considerably. These inequalities ensure that the first building visited in the daily tour l has an index smaller than or equal to the first building in the daily tour $l+1$. Without the symmetry-breaking constraints, a given set of daily tours would be represented by different solutions, corresponding to the permutations of the daily tour indices (for example, two solutions could be generated by transposing the tour of the first day and the tour of the second day).

Model (2.6)–(2.13) is still not sufficient to obtain solutions that have the proper form. Figure 2.1 displays a solution that could be obtained with the model for $d = 1$. That solution contains two *subtours*, one that starts and ends at the hotel, as required by constraints (2.10), and another that does not contain the hotel. No constraints currently forbid such subtours. Therefore we need to introduce *subtour elimination constraints* (SEC). Classical SEC formulations involve an exponential number of constraints, a drawback that can be overcome by using cutting-plane methods. Since developing such methods requires a significant amount of time, we chose to formulate the SEC in a "compact" way by using a polynomial set of constraints. The latter constraints are *multicommodity flows* constraints (see [1], p. 410).

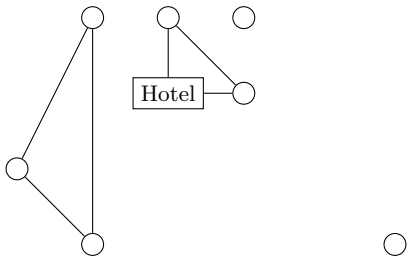


Fig. 2.1: A possible solution containing two subtours

The idea underlying multicommodity flow is to send one unit of flow of an "imaginary" product (in the sense that this product does not represent any physical object being transported) from the hotel to each node in each daily tour. This forces each daily tour to be connected. This can be implemented by adding to the model the following constraints, where w_{ij}^{kl} is the flow on $(i,j) \in A$ whose final destination is $k \in V^0$ in tour l.

$$\sum_{j \in V^i} w_{ij}^{kl} - \sum_{j \in V^i} w_{ji}^{kl} = \begin{cases} y_k^l, & i = 0 \\ -y_k^l, & i = k \\ 0, & i \neq 0, k \end{cases} \qquad i \in V, k \in V^0, l = 1, \ldots, d$$

$$0 \leq w_{ij}^{kl} \leq x_{ij}^l \quad (i,j) \in A, k \in V^0, l = 1, \ldots, d$$

Because of time constraints, no exact methods for the weekly problem were implemented. Instead the team developed heuristic methods, as described in the next section.

2.5 Heuristic Methods for the Weekly Problem

2.5.1 *TOP: Heuristic Methods*

Heuristic methods are often preferred to solve this type of problem, since they can provide useful (if not always optimal) solutions significantly faster than exact methods ([4], p. 4). In order to "feed" the annual problem, we require a large number of weekly solutions. It is thus crucial to compute such solutions quickly. Heuristic methods are also interesting for solving the TOP model presented above, because they can provide starting solutions that accelerate the running time of exact algorithms. For VRP-type problems, the *local search* heuristic methods are effective and can be implemented relatively quickly.

2.5.2 *TOP: Local Search*

Local search is a type of heuristic method by which routes (obtained via a "greedy" algorithm or simply empty routes) are improved by a series of *moves*. There are several types of moves (a more detailed description of local search moves can be found in [4], pp. 4-6).

- Location-based moves:
  - **Add (i, j):** add location i (not in tour) immediately after j (in tour);
  - **Switch (i, j):** insert location i (not in tour) in place of j (in tour).
- Routing-based moves:
  - **2-opt (i, j):** replace 2 arcs (one starting at i, the other ending at j) in a tour by 2 other arcs (intra-route);
  - **2-opt* (i, j):** replace 2 arcs (one starting at i, the other ending at j) in different tours by 2 other arcs (inter-route);
  - **Relocate (i, j):** remove a sequence of visits (starting at i and ending at j) from a tour and insert the sequence in a tour (intra- or inter-route);
  - **Swap (i, j):** swap locations i and j in a tour or in 2 tours (intra- or inter-route).

The goal of these moves is to improve the current solution by changing one of the arcs within it. This is usually carried out by trying each move on each random (i, j) couple and determining whether the solution thus obtained is better than the original. Alternate solutions may also be obtained by restarting after perturbing the solutions (*Iterated Local Search*) or restarting the entire process from scratch (*Multistart/GRASP*).

These methods can become "stuck" on solutions that are locally optimal only, and so methods such as *Simulated Annealing* (SA), *Tabu Search* (TS), or *Variable Neighborhood Search* (VNS) can be used to try to obtain solutions that are globally optimal.

In order for the heuristic methods to be efficient, complex data structures need to be implemented for structuring and keeping solutions in memory. Doubly-linked lists and arrays of pointers allow moves to be performed in $O(1)$ time, and feasibility checking for forward and backward times to be performed in $O(1)$ time (see [5]).

2.6 Conclusions

Over the course of the workshop, the team was able to propose a mathematical formulation of the problem. This involved decomposing the problem in a "natural" fashion based on the planning horizon: a year subdivided into weeks. The team built a mathematical programming model for the annual and weekly problems,

with a Python code for solving the annual problem. The team also wrote a Python code for preprocessing the data provided by the company and to solve the weekly problem using local search methods.

Because of the limited scope of the workshop, several goals were not attained. In particular, one still has to combine the solutions for the weekly schedules in order to solve the annual problem. The ultimate goal is to propose a method generating high-quality schedules for the annual problem. Exact methods to generate optimal results for both the weekly problem and the annual problem would also be desirable.

Acknowledgments

We would like to thank The Co-operators for submitting this problem, as well as Mathieu Giguère and Jean-Michel Plante for their help and support throughout the workshop.

References

1. G. Laporte. Fifty years of vehicle routing. *Transportation Science*, 43(4):408–416, 2009.
2. G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Intersci. Ser. Discrete Math. Optim. Wiley, New York, 1988.
3. J. Renaud, F. F. Boctor, and G. Laporte. An improved petal heuristic for the vehicle routeing problem. *J. Oper. Res. Soc.*, 47(2):329–336, 1996.
4. P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden. The orienteering problem: a survey. *European J. Oper. Res.*, 209(1):1–10, 2011.
5. T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. Time-window relaxations in vehicle routing heuristics. *J. Heuristics*, 21(3):329–358, 2015.

3
What Do You Do? A Cooperative Classification Problem for Insurance Purposes

David Alfonso, Graeme Baker, Guillaume Couture-Piché, Marc-André Desrosiers, Luc Gauthier, Abbas Ghadda, Fabrizio Gotti, Marion Grégoire-Duclos, Samuel Laferrière, Philippe Langlais, William Léchelle, and Zakaria Soliman

3.1 Introduction

The Co-operators is a Canadian insurance cooperative, with approximately 5,000 employees, selling a wide range of insurance policies, from life insurance and businesses to vehicles and farms.

Early in the relationship with a client, a financial *advisor* meets the client and inquires into what his particular needs are, from an insurance point of view, given his line of business. This is done in order to prepare his specific insurance policy and give him a quote for his insurance contract. It might include covering the handling of dangerous materials, the loss of expensive equipment, risks associated with medical care, office space and vehicles insurance, among many other hazards and professional liabilities such as someone tripping and falling in their office. This is the problem we call *asking "What do you do?"*.

It is important to note that the client has little knowledge about the insurance business and cannot be expected to know what his insurance policy should cover. The advisors know more but are not themselves experts in all lines of business and rely on an in-house knowledge base; ultimately they discuss the issue with the *underwriters* who built the resource and know most about hazards and liabilities.

Currently, it is the advisor who asks "What do you do?", and the process sometimes involves a back-and-forth discussion between the client and the advisor and underwriters, which is undesirable (as it takes time and delays the quote).

The ideal situation for The Co-operators would be as follows: the client is able to determine himself what would likely be his insurance policy and gets a quote, solely by answering some questions about his activities on their website, in a self-serve fashion. [1] The underlying system would have to classify the client into one of the various business types that The Co-operators know about, of which there are approximately a thousand. Determining the insurance policy from the business type is a process already in place.

An intermediary step towards this ideal would consist in the advisor answering questions about the client posed by the automated system. The questions could then be slightly harder to understand and answer, because the advisor knows more than the client about insurance. Currently the advisors have to pick the correct activity code (i.e., business type) by hand, out of a thousand, which sometimes calls for difficult judgment calls or leads to small errors, such as picking the incorrect code from among similar codes. In some cases the advisor's choice must be validated by an underwriter.

David Alfonso · Abbas Ghadda · Fabrizio Gotti · Samuel Laferrière · Philippe Langlais · William Léchelle · Zakaria Soliman
Université de Montréal

Graeme Baker
Queen's University

Guillaume Couture-Piché · Marc-André Desrosiers · Luc Gauthier · Marion Grégoire-Duclos
The Co-operators

[1] similarly to the service offered at https://www.coveryourass.ca/

Also, it is well known to The Co-operators that the current taxonomy of business types is not perfect. In particular, new categories are only added on a yearly basis and after enough people have determined that it was missing. In the mean time some advisors have had to (mis-)classify their clients in other categories. It would be helpful if a system could unambiguously find that a new client doesn't fall in the available types and that a new category should be created (or that two categories should be the same).

3.2 Problem Definition and Available Data

The task is to classify new clients into activity codes, based on *some* human interaction with and about them (their activities, possessions, employees and workplace, etc.). A potential solution is allowed to query the client in any way, given that it is user-friendly: only questions should be asked that the client can readily answer (and above all, understand).

Then we have to build a classification procedure to match a client to his most suitable activity type (IBC code). This being a key step for the insurance business, mistakes can be very costly and the system should answer "I don't know" rather than make an error (in other words, minimize the number of errors).

The Co-operators made available to us two main sources of data:

- The "master document" is the taxonomy of IBC codes, a sample of which is pictured in Table 3.1. This is the list of possible classes for our classification problem, along with a little information about the class. [2]

- The "AM-Best" corpus: a collection of approximately 500 documents describing activity sectors in great detail. An available many-to-one mapping gives the correspondance between IBC codes and each AM-Best document. The documents are rich in specific knowledge but are written in natural language, targeted to human readers. Each document follows a predefined structure (sections include *Risk Description*, *Materials and Equipment*, *General Liability: Premises and Operations*, *Professional Liability*, etc.), but is not organized for machine-reading beyond this structure. A short sample about Nail salons (from the document about *Beauty Salons and Barber Shops*) is presented below. A typical document consists of 100 such paragraphs.

  Nail salon

  Manicurists or nail specialists pamper the nails on people's hands and feet. They clean, shape, strengthen, and extend nails; soothe and protect them with special treatments, paint them, and possibly adorn nails with artwork or stones. In 2008, there were 76,000 manicurists and pedicurists in the United States, according to the government's Bureau of Labor Statistics. The nail care industry reports that the number of people who seek nail salon services is increasing each year, and Americans spend billions on nail care products and services.

Our team came up with two major approaches to tackle the problem: a method based on search engines and a method based on decision trees. We will present them in turn.

3.3 Search Engine

In this approach, we ask the client a sentence-long description of his activities. This could be the "about us" section of his website, for instance. We treat this description as a query and look for the best-matching activity types in our database, with a search engine. In the context of a search engine, a *query* is answered with matching *documents*: here the documents to be returned are the IBC codes.

One team * (F. Gotti, Z. Soliman) implemented a prototype of this approach.

[2] A very desirable resource would be the current population of the classes: examples of clients for each class. This would allow us to train systems on real data and make better predictions.

Table 3.1: Sample from the taxonomy of IBC codes: these are the types of businesses that a new client has to be classified into. The leftmost "IBCCode" is the activity type to be determined. The next two columns (IBCCodeGroup and SubIBCGroup) show a part of the existing taxonomy structure. The IBCCodeDescription in the center is used by the Search Engine approach presented in Section 3.3. The TRUE/FALSE flag columns are used by the Decision Tree approach in Section 3.4. There were 1,311 total IBC codes in the file version we used. Several columns have been left out for brevity.

IBCCode	IBCCodeGroup	SubIBCGroup_CG	IBCCodeDescription	CookingFrying_	HighPiledStock_C	Welding_	Woodworking_	PlasticManufacturing_
0730	Agriculture	FarmsOther	Agriculture, horticultural services, N.O.C. (incl. Tree and Sod farms)	FALSE	FALSE	FALSE	FALSE	FALSE
0190	Agriculture	OtherAnimals	Alpacas / Llamas	FALSE	FALSE	FALSE	FALSE	FALSE
0141	Agriculture	Horses	Horse farms - equestrian, riding schools, rental, etc. No breeding. No boarding.	FALSE	FALSE	FALSE	FALSE	FALSE
0734	Agriculture	FarmsOther	Nurserymen, florist (not store or greenhouse)	FALSE	FALSE	FALSE	FALSE	FALSE
0150	Agriculture	Poultry	Poultry - table egg production	FALSE	FALSE	FALSE	FALSE	FALSE
7408	B&P Services	ServicesCorporate	Authors	FALSE	FALSE	FALSE	FALSE	FALSE
7297	B&P Services	ServicesIndividuals	Beauty parlours - no tanning beds- no laser therapy	FALSE	FALSE	FALSE	FALSE	FALSE
7291	B&P Services	ServicesIndividuals	Funeral home, mortician, undertakers	FALSE	FALSE	FALSE	FALSE	FALSE
7408	B&P Services	ServicesCorporate	Genealogists	FALSE	FALSE	FALSE	FALSE	FALSE
8291	B&P Services	BPServicesOthers	Libraries	FALSE	FALSE	FALSE	FALSE	FALSE
1717	Contractors	HVAC	Air conditioning equipment installation - incl. heat pumps	FALSE	FALSE	FALSE	FALSE	FALSE
1515	Contractors	ContractorsOthers	Blasting contractors	FALSE	FALSE	FALSE	FALSE	FALSE
1535	Education	Schools	Public schools - secondary	TRUE	FALSE	FALSE	FALSE	FALSE
5523	Garage	MecanicsandGarages	Auto specialty shop (Mufflers, transmissions etc.) Excl audio	FALSE	FALSE	TRUE	FALSE	FALSE
5526	Garage	MecanicsandGarages	Car wash - self serve	FALSE	FALSE	FALSE	FALSE	FALSE
8950	Government	Government	Jails / correctional centres / prisons, penal institutions	TRUE	FALSE	FALSE	FALSE	FALSE
8949	Government	Government	Municipal - administration or office building; adoption agencies	FALSE	FALSE	FALSE	FALSE	FALSE
8026	Health Serv.	HealthServicesShortTerm	Dermatologists offices	FALSE	FALSE	FALSE	FALSE	FALSE
8062	Health Serv.	HealthServicesShortTerm	Surgical centers (outpatient) including Laser surgery	FALSE	FALSE	FALSE	FALSE	FALSE
7022	Hospitality	Hotels	Hotels - seasonal - not licensed	TRUE	FALSE	FALSE	FALSE	FALSE
7061	Hospitality	Motels	Motels / tourist courts - seasonal, licensed with food	TRUE	FALSE	FALSE	FALSE	FALSE
2011	Mfg./Proc.	Food	Cheese manufacturing	FALSE	FALSE	FALSE	FALSE	FALSE
2052	Mfg./Proc.	Food	Chocolate and cocoa mfg.	FALSE	FALSE	FALSE	FALSE	FALSE
3570	Mfg./Proc.	ManufacturingOthers	Lamp manufacturing - electric - including light bulb and tubes	FALSE	FALSE	FALSE	FALSE	FALSE
3590	Mfg./Proc.	NonMetallicMinerals	Nail manufacturing	FALSE	FALSE	FALSE	FALSE	FALSE
3595	Mfg./Proc.	NonMetallicMinerals	Plumbing supplies mfg.	FALSE	FALSE	FALSE	FALSE	FALSE
2820	Mfg./Proc.	Chemicals	Tank mfg. - plastic, synthetic resins & fibres , not pressurized	FALSE	FALSE	FALSE	FALSE	TRUE
2081	Mfg./Proc.	Food	Vinegar distilling	FALSE	FALSE	FALSE	FALSE	FALSE
1394	Oil&Gas	OilGas	Pipe Insulation	FALSE	FALSE	FALSE	FALSE	FALSE
5302	Realty	CommercialandIndustrialRealty	Public markets (open air)	TRUE	FALSE	FALSE	FALSE	FALSE
7840	Recreation	Theatres	Drive-in movie theatres	TRUE	FALSE	FALSE	FALSE	FALSE
7983	Recreation	RecreationOthers	Skateboard parks	FALSE	FALSE	FALSE	FALSE	FALSE
5811	Restaurant	Restaurants	Pizza Restaurants - not licensed, no deep frying	TRUE	FALSE	FALSE	FALSE	FALSE
5994	Retail	RetailOthers	Pet shops	FALSE	FALSE	FALSE	FALSE	FALSE

3.3.1 *Lucene "off the shelf"*

Lucene [3] is a classic search engine software library, developed by Apache. First, and offline, documents are preprocessed (tokenized and lowercased) and indexed: a document is converted into a vector of its tokens. To answer a query, the query is preprocessed in the same fashion and its words vector is matched to document vectors using TF-IDF. Lucene precomputes intermediary results for fast processing.

In our case, we want the search engine to return IBC codes, so the natural documents to be matched are the corresponding descriptions (the *IBCCodeDescription* field values in the master document, Table 3.1). It becomes obvious that the documents are much too short for the matching function to be effective (typical information retrieval documents are 1,000 tokens long, whereas our descriptions usually contain fewer than 10 words). We therefore expand the documents' contents using available resources: expert documents and the Web.

3.3.2 *Web Scraping*

IBC codes' descriptions (which we will call *labels*) are enhanced using content from 3 sources:

- the AM-Best documents corresponding to each IBC code;
- relevant web pages found by querying the Bing search engine with the label;
- relevant Wikipedia articles as returned by Bing.

The web search returns five to twenty results, depending on many variables such as sponsored links, related queries, etc. Those web pages are downloaded (this step took 3 hours) and converted to text. Wikipedia articles require some special preprocessing to take markup and templates into account. Those results are added to the label for each IBC code, providing us with an enriched corpus of documents.

3.3.3 *Evaluation*

With help from an expert underwriter, we conducted a small-scale evaluation of the search engine. We were provided with 50 examples of possible user queries, along with the correct IBC code for those queries. Most queries were short (about 7 words long on average) but altogether diverse. Here are some examples from the evaluation set.

- *I have a nail salon* — 5224 – Barber shop & beauty salon supplies;
- *Welcome Centre Immigrant Services is a one-stop service designed to guide and support immigrants in Durham and York Regions* — 8644 – Civic, community, special interest (non-medical) associations or groups;
- *We make artisanal beer* — 2070 – Distilleries (alcohol mfg.);
- *We are health care professionals trained to evaluate hearing loss and related disorders, including balance (vestibular) disorders and tinnitus (ringing in the ears) and to rehabilitate individuals with hearing loss and related disorders* — 8027 – Audiologists;
- *I drive a limo for special occasions (weddings, bachelor parties, etc.)* — 4115 – Limousine services;
- *I breed horses* — 0140 – Horse farms – breeders.

Performance was then measured as the percentage of queries for which the correct document was returned in the top k search results. The results of our method are displayed in Figure 3.1, with k varying from 1 to 10, with and without the corpus expansion by web scraping. Web scraping yields a large improvement, of

[3] https://lucene.apache.org/

about 20 percentage points across all values of k. For nearly 60% of queries, the top matching document is the correct one. The correct IBC code is found in a shortlist of 10 results for 80% of queries.

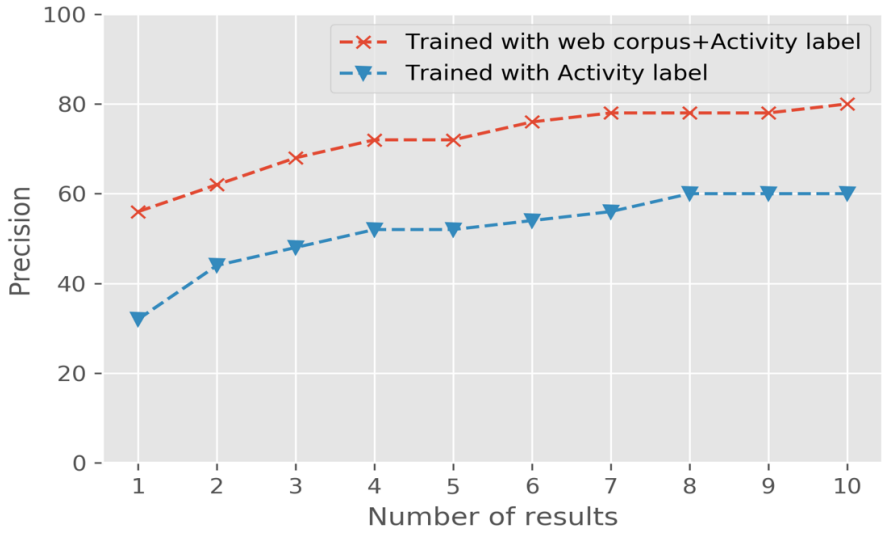


Fig. 3.1: Small-scale evaluation of the search engine approach on 50 examples. Fraction of queries for which the correct activity code is found in the top k results, depending on k. Web scraping yields an improvement of 20 percentage points. With web scraping the correct result can be found in the top 10 search results for 80% of queries.

3.3.4 *Future work*

While promising the implemented system could be improved in many ways, described below.

- IBC code descriptions were augmented automatically, but more precise labels could be produced by humans. The advisors know best what their (current) clients' businesses entail, and what keywords best match the IBC codes. Furthermore, additional document data could be gathered from the existing clients, either in a crowd-source manual fashion or automatically, e.g., through scraping the clients' websites or textual interactions (such as email exchanges).
- The web scraping step could be improved by refining the processing of the current labels (e.g., keeping only the key words, or splitting the descriptions into several queries), or improving the current labels. The most relevant phrases from the AM-Best documents could be used to retrieve more relevant documents. The resulting web pages could also be processed in a better fashion, e.g., by focusing on their contents (removing templates, headers, etc.).
- The negative parts of the current descriptions (e.g., in *smoldering no abrasives*) should be translated as negative queries (using the NOT search operator) rather than being ignored.
- The search engine could be improved with standard Information Retrieval techniques such as query expansion (adding relevant words to the client's description of himself to match the corpus more closely (see Section 3.3.5).
- The input field to the search engine should be spell-checked.

- The returned results could be reranked based on complementary sources of information, for example drawing from the current client base. For instance, TF-IDF scores for each class could be weighted by current client frequency, so that categories used more often bubble up in the result list.
- With training data, a confidence estimator could be built: the system would yield results when it is confident enough or else say "I don't know." A well-tuned confidence score would also allow the system to return a variable-length list of results.
- Many potential features depend upon the integration with the graphical user interface. For instance, plausible phrases could be offered for dynamic auto-completion while the client is entering his description.
- The evaluation of the search engine should be based on real client queries and manual classification decisions made by the advisors. The test set we employed was created artificially by domain experts and may not be representative of what the actual users would enter as input.

3.3.5 *Keywords Discovery*

A subgroup of the team (D. Alfonso, S. Laferrière) pursued a lead related to keywords discovery. Using word2vec and WordNet, [4] the subgroup used the words of the IBC label to retrieve synonyms and related words. The results of this work could be used for the query expansion improvement of the search engine mentioned above.

The labels were tokenized and stopwords removed. Then, for each word, the most similar words according to word embeddings, and from WordNet the holonyms, hyperonyms, hyponyms, synonyms, and words with a small enough distance, are returned. [5]

For the activity label "Cheese manufacturing," this yields results such as (for WordNet) *brie, camembert, cheddar, cottage cheese, pot cheese, farm cheese, farmer's cheese*, *yoghurt, whey, curd, mozzarella, dream up, fabrication, manufacture, manufacturing, formation, shaping, create*, out of which *dream up*, *formation* and *shaping* seem incorrect. For word2vec, this yields *cheese industrial, butter, cheddar, milk, industries, cheeses, factories, automotive, manufacturing, manufacture*, *manufacturers, cottage cheese, bacon, mozzarella, manufactures, products, machinery*, out of which *automotive* and *bacon* look incorrect.

We used the gensim [6] Python implementation of word2vec, with pretrained embeddings. The AM-Best corpus could be used to train more domain-adapted word embeddings. In WordNet, the first synset (the most frequent) was selected for each word. In the context of a client query, words could be disambiguated and the correct synset chosen accordingly. The list of stopwords was taken from NLTK.

3.4 Decision Tree

In this approach, we ask the client a series of questions about his activities, such as "*Does your job involve welding?*". Each consecutive answer makes some activities more probable (and the others less so) until one class emerges as most certain. In the "hard" decision tree setup, some classes could be filtered out at each step, but it is better to allow for mistakes, so a *probabilistic* decision tree (also called "soft" decision tree) was chosen. Another subgroup of the team (G. Baker, W. Léchelle) implemented a prototype of this approach.

In this approach it is difficult to find the appropriate *questions*. A question should be easy for the client to answer, and the practitioners' answer (for any business type) has to be known (that is, there is an expected answer for each activity code to any particular question). To simplify the problem, we focused almost exclusively on questions with a *yes or no* answer (multiple-choice questions could also be considered).

[4] https://wordnet.princeton.edu/

[5] or antonyms if the word was preceded by "no" in the label

[6] https://radimrehurek.com/gensim/

Furthermore, it is a hard task to generate automatically natural language questions to obtain various types of information, while for small numbers of questions it is enough to write them by hand. We didn't address the "question generation" part of the problem and always asked about items in a generic form: "*Does your job involve X?*".

All questions being about a particular item, the answers being binary, and the answer from most businesses to most questions being negative, we refer to the questions as *flags*: most flags are false for most clients (e.g., most businesses don't involve *nails*,say, or *cheese*), and true for a handful of them (*nail salons*, *nail manufacturers*, *cheese manufacturers*).

3.4.1 *Algorithm*

The algorithm works as follows. The system maintains a probability distribution over possible classes. The initial probability distribution is based on the current set of clients: the categories that appear most frequently have a higher a priori probability. At each step the system asks the highest entropy question (i.e., the question it is "most uncertain" about: that to which an answer will yield the most information and will affect the probability distribution the most). Then the system updates the probability of each class based on the answer. It multiplies the score of each matching class by a parameter α and normalizes the scores to obtain probabilities. A class matches an answer if its members would return this answer: for example, to the question "*Does your job involve cheese?*", nail manufacturers match "no" and cheese manufacturers match "yes."

We used a value of 10 for the parameter α after briefly experimenting with a value of 4, which seemed less effective. Higher values should be used when users are very confident about their answers and lower values can be used to assign a smaller weight to certain questions (and a larger weight to other priors). Users can answer "maybe" to any question, meaning that they don't understand the question or don't know the answer. Then the system skips the question without updating the probability distribution. This corresponds to a value of 1 for α.

3.4.2 *Flags*

We used flags coming from 3 sources:

- The master document has default values for all IBC codes for 22 known hazards, some of which are shown in Table 3.1 (*Cooking Frying*, *High Piled Stock*, *Welding*, *Woodworking*, etc.).
- Preliminary work carried out by The Co-operators on hazards extraction from AM-Best documents (using word embeddings and community clustering) gave us approximately 1500 truncated single words as flags. [7] Examples of these flags can be seen in the sample run in Section 3.4.3: *custom*, *farm*, and *horse*.
- A team member (A. Ghaddar) looked into hazard extraction from the AM-Best corpus, which yielded about 700 flags. Each hazard being extracted from a specific AM-Best document, it is associated only to the few IBC codes described in the document and we assume the flag to be negative for all other activities. The flags produced by this method are noun phrases, such as "*an elevator*," "*stairs on site*," "*a night/weekend security service*," "*a number of vehicles that are stored on site, repairing or rebuilding the facility*," or "*identification numbers etched on all essential tools and equipment*."

For hazard extraction from the AM-Best documents, we preprocessed and dependency-parsed the documents with Stanford CoreNLP, [8] from which we extracted the noun phrases (NPs). We looked specifically for two patterns:

[7] As these hazard words were stemmed during the extraction process, what the flags entailed was not always clear.

[8] https://stanfordnlp.github.io/CoreNLP/

- "... *the insured has* ⟨NP⟩ ..."; and
- "... ⟨NP⟩ *should be* ...".

This program returned a list of about 1200 results, noun phrases that were good candidate flags. A manual inspection of these noun phrases by a team member allowed him to merge some very close phrases (mostly identical, with slight modifications such as additional adverbs or adjectives), bringing the list down to 700 candidates in one hour. A better way would be for a domain expert to filter out the noise and pick those flags that are most relevant (a few hundreds of them).

3.4.3 *Results*

A sample run of the system is pictured below. Each page displays two iterations of the algorithm. Each iteration is represented by 4 quadrants: the decision tree being dynamically constructed in the top left quadrant, the ongoing interaction with the user in the top right one, the current probability distribution in the lower left one, and the current top beliefs in the lower right quadrant.

The first question asked is "*What is your field*?". This is one multiple-choice question we considered, the possible answers being the highest level of categorization of the IBC codes (the *IBCCodeGroup* column in Table 3.1 [9]). The example concerns a horse breeder: thus the answer is *Agriculture*.

The prior distribution over all classes is displayed in the first picture; it is restricted to the classes of the *Agriculture* category afterwards. The step before the first binary questions is not showed: at this step, all classes in the *Agriculture* category have the same prior belief, because of missing data. The question "*Does it involve a farm?*" makes about half the classes gain some probability weight and the other classes lose some weight.

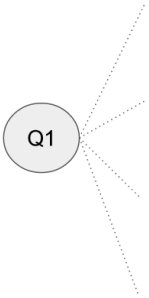
After 4 questions, the correct answer (*Horse farms – breeders*) is in the top 3 most probable answers. At this point the systems asks questions to discriminate between the top results (making the highest probability change place, picking the highest entropy question), such as "*Does it involve breeding?*".

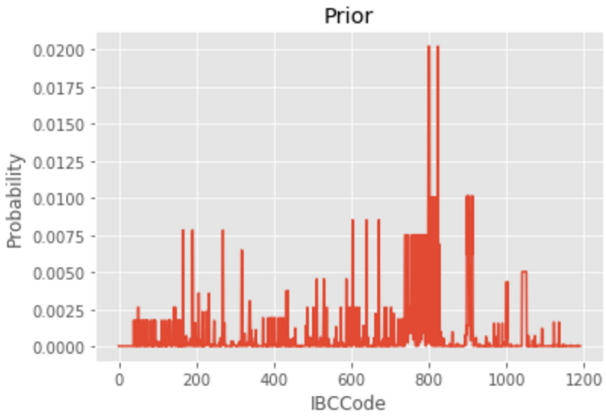
We can automatically measure how many questions it takes for the correct class to bubble up to the top belief, in the case that all expected answers are provided. This depends on the α parameter (if it is lower more questions are necessary) and on the set of flags used (using more flags means that better questions are available at each step, so fewer questions are necessary). On average, it takes 5 to 7 steps to identify the correct activity code. Worst case scenarios require around 12 questions. These results are encouraging since normal client interactions can involve answering upwards of 20 questions (more if the questions are easy) before becoming tedious by insurance standards.

We also estimated that it takes about two correct answers to make up for one mistake (a mistake meaning an answer that is not the expected answer for the user's target class).

[9] There is only a dozen possible groups, so a client usually knows which group he falls into, and advisors certainly do.

Question	Answer
What is your field?	

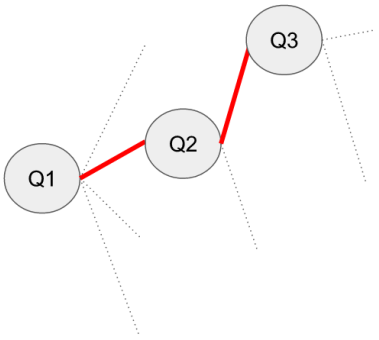


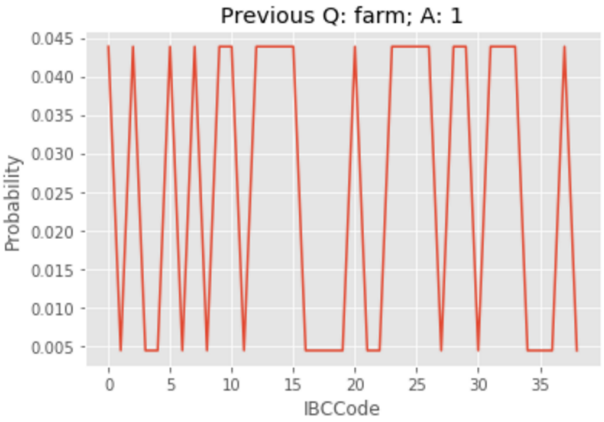


My top guesses are:

2.02% Building, over 6 units/suites
2.02% Building, 1 or 2 units/suites
2.02% Building, 3 to 6 units/suites
2.02% Residential condominium
1.01% Fast food restaurant - licensed
1.01% Family restaurant - licensed
1.01% Pizzeria - licensed, incl deep frying
1.01% Pizzeria, no deep frying - licensed
1.01% Fine cuisine restaurant - licensed
1.01% Condominiums - No mercantile
...

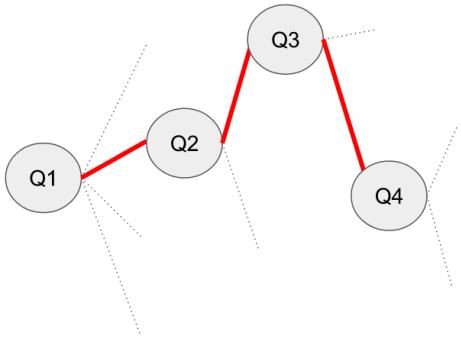
Question	Answer
What is your field?	Agriculture
Does it involve farm?	Yes
Does it involve custom?	



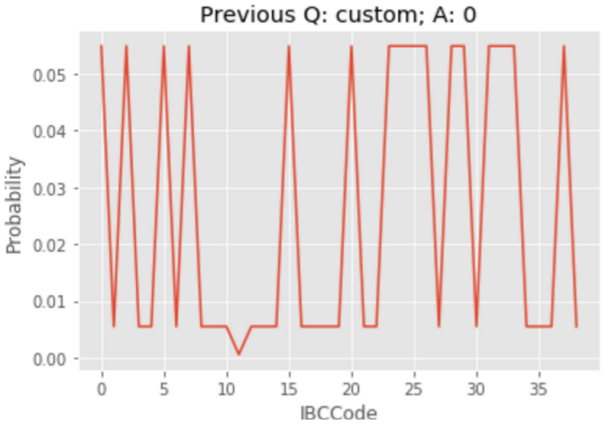


My top guesses are:

4.39% Agriculture, N.O.C.
4.39% Custom Harrowing, Rock Picking
4.39% Horse farms - breeders
4.39% Hog farms
4.39% Mixed farms (small exposures)
4.39% Fur farms
4.39% Mushroom farms
4.39% Other than animal farms - NOC
4.39% Peat plants (farming)
4.39% Dairy farming
...

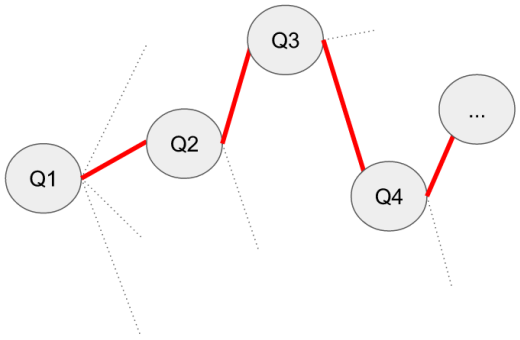


Question	Answer
What is your field?	Agriculture
Does it involve farm?	Yes
Does it involve custom?	No
Does it involve horse?	

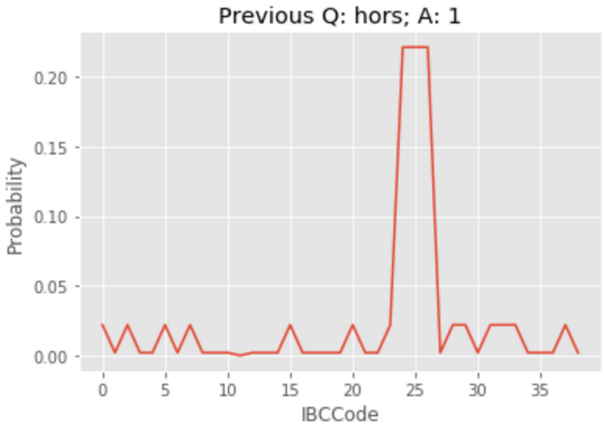


My top guesses are:

5.49% Agriculture, N.O.C.
5.49% Peat plants (farming)
5.49% Fur farms
5.49% Horse farms - breeders
5.49% Horse farms - No breeding/board.
5.49% Horse farms w board - No breeding.
5.49% Dairy farming
5.49% Mixed farms (small exposures)
5.49% Mushroom farms
5.49% Other than animal farms - NOC
...



Question	Answer
What is your field?	Agriculture
Does it involve farm?	Yes
Does it involve custom?	No
Does it involve horse?	Yes
...	



My top guesses are:

22.12% Horse farms - breeders
22.12% Horse farms - No breeding/board
22.12% Horse farms w board - No breeding
2.21% Agriculture, N.O.C.
2.21% Peat plants (farming)
2.21% Fur farms
2.21% Dairy farming
2.21% Mixed farms (small exposures)
2.21% Mushroom farms
2.21% Other than animal farms - NOC
...

3.4.4 *Future work*

In the implementation of this approach, the biggest issue was the lack of precision of the flags' expected answers (i.e., the user make mistakes or are not sure about the answer). This problem arose because the flags had been obtained automatically in very coarse ways. For instance, for the 22 flags from the master document, it is the **default** values for the activity code that are listed. This means, for instance, that most librarians' businesses don't involve *High Piled Stock* (because books are heavy), even though some do. Those who do would make a mistake when answering the question and would need to provide additional correct answers for a correct prediction. Similar problems arise with automatically extracted flags (for instance "*an elevator*").

Constructing the questions by hand would yield the most adequate interaction for the user: the questions could be well formulated and easy to understand. The entropy-maximizing question-picking algorithm ensures that the questions asked are relevant, so the human writers would only need to write as many as possible.

It would be possible to learn the correct expected answers for each question in every class by surveying the current clients (or advisors) to know their answers to a sample of questions. Then the most discriminating questions could be easily determined and questions to which users answer "I don't know" could be dropped.

Up to now we have assumed that a client in any given class would always answer in one way to any given question (defaulting to "no"), but it is straightforward to extend the method to classes giving mixed answers: it suffices to reduce the impact of the probability update on those classes (assign a lower α value to this particular class). Tuning α and assigning larger weights to positive answers than negative ones could yield improvements to the method.

Integration with the user interface can have an impact on how to run the algorithm: for instance, users could be presented with several confidence options on the yes – no spectrum, e.g., *No – I don't think so – Maybe – Probably – Definitely.*

3.5 Conclusion

Our task was to classify incoming clients into insurance-based activity types, using input to be obtained in user-friendly ways. We proposed two complementary solutions, one based on a search engine and the other on a decision tree. The search engine returns the best matching IBC codes and uses a free-text query produced by the client (a short description of his activity), matched against the IBC labels and enhanced by expert documents and web scraping. The decision tree method poses yes-or-no questions to the client regarding his business, updating the probability of each class on the fly until one answer emerges as most probable.

Both solutions were implemented as proof-of-concepts. In a small scale manual evaluation of the search engine system, the correct class was returned in the top 10 results for 80% of queries. With perfect answers, the decision tree finds the correct class typically after 5 to 7 questions. Both implementations could be further improved in many ways, above all based on real client data, as detailed in their respective sections.

4
Parameters Affecting the Operational Control of Log Turners

Jakub Witkowski, Jean-François Plante, Frédéric Godin, Yvon Hubert, and Serge Constantineau

Abstract In a sawmill, the log turner is a machine centre that plays a crucial role in the log breakdown. It rotates a log before it reaches the first cutting machine-centre on the production sawline, in such a way that the position of the log is optimal with respect to the cutting tools. Any deviation between the targeted rotation angle and the actual rotation results in a decrease of the log value. The goal of this project is to analyze the operational data pertaining to a specific sawmill and to determine whether (or not) one or several parameters have a significant impact on the performance of the log turner. If such an impact exists, the information gained will be used within the framework of another project in order to improve the operational control of the log turner.

4.1 Introduction

The forest industry is one of the largest industrial branches in Canada. In 2013, forest industry activities were worth 1.25% of the country's GDP, making it a crucial sector for Canada's economy. FPInnovations is "a not-for-profit world leader that specializes in the creation of innovative scientific solutions in support of the Canadian forest sector's global competitiveness."[1] The BID group is an integrated company that delivers "a complete range of innovative world-class equipment, services and turnkey installations for the forestry industry."[2] One of its constituents, COMACT, "is a North American leader in the design, manufacturing, and optimization of wood processing equipment as well as PLC/Controls for sawmills and planer mills."[3] To improve lumber output volumes, the log processing in sawmills is highly optimized. This report is a summary of the data analysis that took place in the context of the Eighth Montréal Industrial Problem Solving Workshop held at the Centre de recherches mathématiques from August 7 to August 11, 2017. In

Jakub Witkowski
SGH Warsaw School of Economics

Jean-François Plante
HEC Montréal

Frédéric Godin
Concordia University

Yvon Hubert
BID Group

Serge Constantineau
FPInnovations

[1] https://fpinnovations.ca/about-us/pages/our-work.aspx
[2] http://www.bidgroup.ca/about/mission/
[3] http://www.bidgroup.ca/products/comact/

the context of the workshop, we focused on the log turners, which are machines rotating logs in an optimal position before they get sawn. Errors in the rotation angles obtained by the log-turners are suspected to cause a significant loss of value by reducing lumber output. The purpose of this project is to understand how this process could be improved.

As logs enter the cutting process, a 3D vision system scans them, yielding a 3D representation of the log that is used to derive its physical characteristics. An engine then recommends an optimal rotation angle for maximizing the output value. The recommended angle is then passed on to a Programmable Logic Controller (PLC), which activates the log turner equipment to achieve the rotation. After the log is rotated, it enters the sawline and gets cut into pieces. This process is illustrated in Figure 4.1.

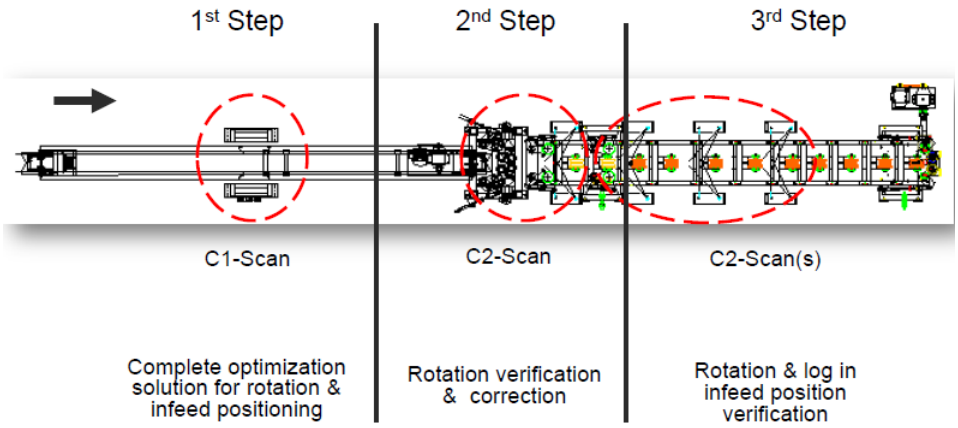


Fig. 4.1: Illustration of the steps of log processing for the log turner in a sawmill.

Logs are not always entering the sawline at the angle proposed by the optimization algorithm. For instance, the machine might be unable to achieve the recommended rotation due to sliding. A deviation from the targeted rotation angle may result in a decrease of the value of the log output. Experiments performed by FPInnovations show that the difference between the optimal and the realized angle of rotation can be as big as 20 to 25 degrees, possibly leading to a waste of lumber and monetary losses. For the workshop, we obtained one year of data from the optimizer module of a sawmill to investigate relationships between turning errors and potential explanatory variables. Factors that are potential causes of errors can be grouped into three main categories: log characteristics (e.g. geometrical characteristics such as length or diameter, or log attributes such as wood species), operational parameters (e.g. line speed or distance of rotation), and machine conditions. A better understanding of error-causing factors should help improve the production process.

After presenting the data and a descriptive analysis thereof in Section 4.2, further analyses based on models are presented in Section 4.3. Recommendations and conclusions are formulated in Section 4.4.

4.2 The Data

A dataset was provided by COMACT (The BID Group) through FPInnovations. The full data set includes records for 2,651,295 logs that were processed between the 11th of June 2016 and the 29th of June 2017. As it is being processed in the production line, each log gets analyzed by different scanners. The database made available to us contains data generated by three of them: Scanners 0, 1 and 5. Scanner 0 is used to obtain the general specifications of the log, used to determine the optimal turning angle. Scanner 1 measures

the rotation angle after the log has been turned and in some sawmills, a further rotation is possible; this second stage of turning, however, was disabled at the sawmill that provided the data. Scanner 5 collects the estimated final monetary value of the log after cutting has started. In our database, each scanner generates one line of data for each log.

There are two main response variables of interest. The first is `delta_angle`, which is the difference between the optimal angle provided by the optimization algorithm and the realized turning angle obtained and measured by Scanner 1. The second variable of interest is the value of the log denoted by `price` and updated at each scanner based on the operations carried out thus far. All the characteristics collected by scanners are potential explanatory variables. Another data source that we used to supplement the original database was the production calendar, which provides information about the species of wood cut for each day of production. Table 4.1 presents the list of variables that were available along with basic descriptive statistics. The variable names are self-explanatory. Although each scanner generates one line of data, only the information from Scanner 1 contains `delta_angle`. Summary statistics in Table 4.1 are only for the values saved by Scanner 1. The last column of the table indicates the type of the variable: a characteristic of the log, an operational parameter of the sawmill, a property derived from the optimizer, a response variable.

The database that we were provided with corresponds to a very small fraction of the data that is generated in the sawing process. The PLC, for instance, holds much richer information about the mechanical commands sent to the mill and the positions of the different devices. Similarly, the raw scanner data is used to create a 3D representation of every log, but we only got a few features derived from it during the sawmill operations. To investigate specific factors that are believed to have an impact, additional measures such as log moisture or temperature could also be considered if they were available.

4.2.1 *Data Quality and Cleaning*

Data from Scanner 1 were of primary interest and that subset of the data was first cleaned.

Removing Unusable Data

Some observations from Scanner 1 were removed:

- all logs processed on weekends and on days where fewer than 500 logs were processed (the latter being referred to as machine-testing days);
- logs for which `delta_angle` was not measured;
- logs processed during days where multiple wood species were processed; there are few such days and it is difficult to determine the exact type of wood of each log during such days.

This step of the cleaning procedure resulted in the removal of about 300,000 logs from the data set.

Distribution of `delta_angle`

The empirical distribution of the response variable `delta_angle` shows apparent problems that raise doubts about its reliability. Figure 4.2 shows a histogram having a shape similar to the normal distribution, except for the very high proportion of logs with a precise angle error of 0, 5, or -5. It is highly improbable that this distribution comes from a "natural" process. It could be due to some kind of measurement error by scanners, to the use of a systematic fixed value under certain operational circumstances, or to other technical glitches. In any case, we looked for links between the overabundance of those values (-5, 0, and 5) and any other variable, or any cycle, including time variables, but could not find any link. It is therefore difficult to determine which observations are impacted by such errors, making the removal of the corresponding logs

Table 4.1: Summary statistics of the variables available at Scanner 1.

Variable	min	mean	max	sd	Type
Continuous variables					
angle	−180.00	−6.08	359.00	102.18	Operational
angle_solution_rotation	−180.00	−6.22	180.00	108.07	Optimizer
big_end_diameter	0.00	8.29	20.68	2.17	Characteristics
chip_volume	−6332.81	3308.23	32 754.80	1391.69	Optimizer
curve	0.00	0.51	2.38	0.34	Characteristics
delta_angle	−179.00	−0.01	179.00	20.85	Response
diameter	0.00	7.11	21.13	2.15	Characteristics
diameter_jas	0.00	6.90	19.69	2.12	Characteristics
length	0.00	143.94	219.00	29.22	Characteristics
line_gap	0	157	7 006 640	4670	Condition
line_speed	0.00	420.44	550.00	124.64	Condition
max_diameter	0.00	9.10	23.90	2.42	Characteristics
min_diameter	0.00	6.91	19.20	2.07	Characteristics
nominal_value	0.00	15.04	246.57	16.90	Optimizer
nominal_volume	0.00	3935.29	32 820.00	2343.80	Characteristics
plc2	-1.17E+09	6.24E+03	1.95E+09	2.08E+06	Condition
position_left_canter	0.00	2.86	9.02	1.04	Operational
position_left_front_quad	−32.07	2.95	11.00	1.06	Operational
position_left_rear_quad	−32.07	10.89	11.00	0.94	Operational
position_right_canter	0.00	2.86	9.03	1.04	Operational
position_right_front_quad	−32.07	3.02	11.00	1.09	Operational
position_right_rear_quad	−32.07	10.89	11.00	0.94	Operational
price	0.00	15.04	246.57	16.90	Response
product_mixt_loss	0.00	0.03	91.54	0.24	Optimizer
real_volume	0.00	3324.04	30 819.90	2324.63	Characteristics
reduction_factor	0.00E+00	3.64E+23	9.49E+29	5.88E+26	Operational
sawdust_volume	0.00	408.44	4999.49	394.83	Characteristics
shift	4.00	259.97	510.00	140.47	Operational
small_end_diameter	0.00	7.30	19.80	2.11	Characteristics
sweep	0.00	1.27	19.20	0.79	Characteristics
taper	0.00	1.00	17.92	0.54	Characteristics
total_value	−1.00	15.04	246.57	16.91	Optimizer
turning_diameter	0.00	1.58	20.44	3.11	Operational
turning_distance	−28.30	−0.46	30.05	7.19	Operational
volume	0.00	7047.90	51 580.20	3837.13	Characteristics
volume_international	−1.99	18.00	222.01	13.90	Characteristics
width	0.00	4.60	8.38	1.26	Characteristics
x_axis_offset	−17.96	0.15	15.71	3.43	Operational
x_axis_skew	−17.17	0.02	13.75	1.02	Operational
y_axis_offset	−18.66	−0.17	15.94	3.34	Operational
y_axis_skew	−12.82	0.02	35.85	1.18	Operational
Discrete variables					
block_sawn	0.00	0.47	1.00		Operational
butt_first	0.00	0.49	1.00		Operational
gap_too_long	0.00	0.05	1.00		Operational
gap_too_short	0.00	0.02	1.00		Operational
optimization	0.00	2.98	3.00		Condition

infeasible. This issue might influence the results of our analyses since none of the models we used accounts for an overabundance of those three values.

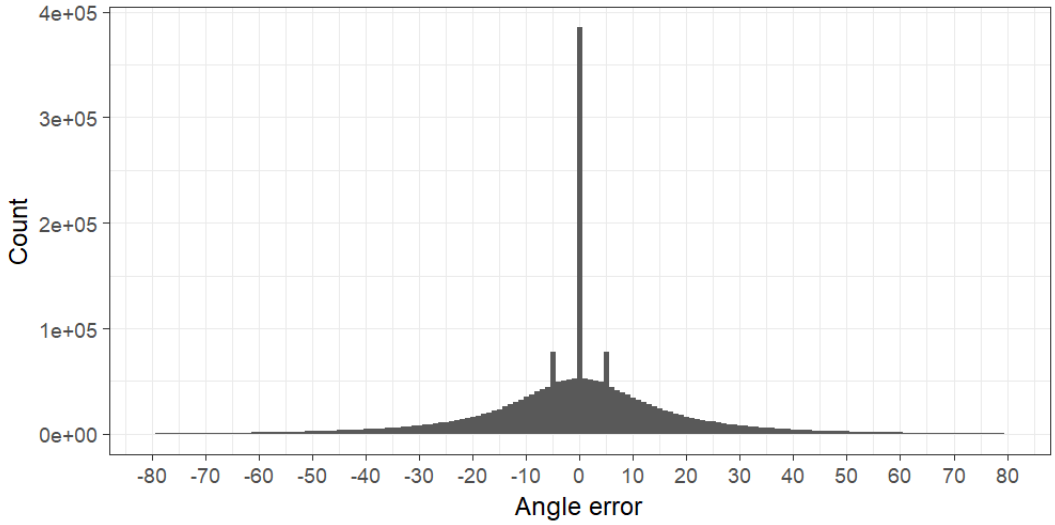


Fig. 4.2: Empirical distribution of `delta_angle`, the difference between the requested and realized turning angles as recorded by Scanner 1.

Lack of a Unique ID

In the dataset, each scanner records one line for each log, but the ID assigned by scanners to the same log are different. Although the system tracks the log and generates a unique ID for the log during the production, that unique ID is not saved in the database that was made available to our team. While `delta_angle` is calculated during the production and is already available in the database, any other measurement that relies on multiple scanners, such as the difference in the projected value of the log (based on `price`), requires that we match the data from the three scanners. Looking at the characteristics of the logs, we found that `sweep`, a measure of geometry, is identical for a given log in the three scanners, most likely because it was calculated from the more thorough image of Scanner 0 but saved in the database for the three scanners. Records with equal sweep within 15 minutes were merged as a unique log. Based on this procedure, 1,373,272 logs were matched with reasonable confidence, and analyses of changes in log values were performed on this smaller data set.

4.2.2 *Descriptive Analysis of Angle Rotation Error*

The variable `delta_angle` was identified as the main variable of interest. As mentioned above, measurement errors probably impact the quality of data points for this variable and the analyses we conducted might be affected by such errors. Keeping this in mind we analyzed the delta rotation angle from two different perspectives: firstly we observed its evolution through time (a time series perspective), and secondly we investigated its relationship with potential explanatory variables.

The first part of the analysis focuses on identifying time-related patterns in the variability of the `delta_angle`. Figure 4.3 shows the evolution of the daily mean of the absolute value of `delta_angle` (i.e.,

an error of +10 or −10 degrees is coded as 10) in order to observe patterns over the year, including potential seasonal cycles. The plot indicates the presence of high autocorrelation in daily averages of absolute errors, meaning that the magnitude of the errors on a given day seems to depend on the magnitude of the errors observed in the previous days. Discussions with BID Group and FPInnovations team members led to the identification of potential causes for the observed autocorrelation; ad hoc maintenance operations on the cutting machinery are sometimes performed throughout the year. Sudden decreases in errors could therefore be explained by the fact that broken machine components were fixed by maintenance. When such external factors can be measured, accounting for them in a model allows to see better the other sources of error. Unfortunately we could not get access to the maintenance data during the workshop. Log conditions (temperature, moisture content), which are currently not incorporated in the dataset, could also play a role in the time-dynamics of errors.

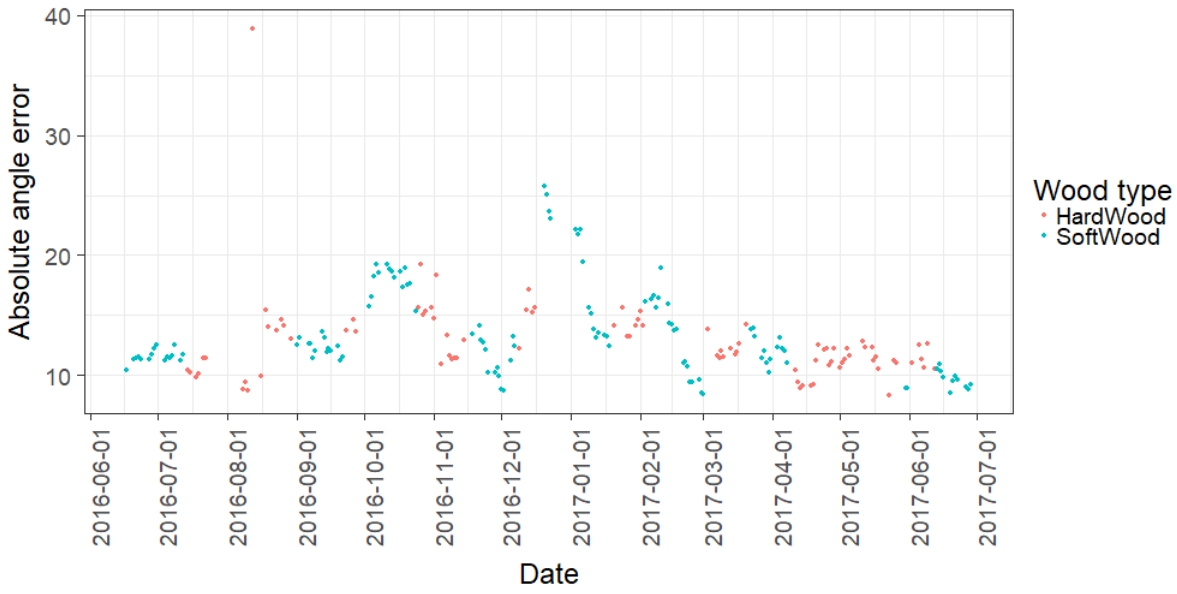


Fig. 4.3: Daily mean absolute value of `delta_angle` through time.

Figure 4.4 shows the evolution of the mean absolute value of `delta_angle` across months and days of the week to explore the potential presence of seasonal patterns or weekly cycles. During the early months of the year the mean error is following a downward trend that reaches its minimum in May. From May on, the trend changes and the mean absolute error increases. It is interesting to observe the big jump in the value of the error between September and October, followed by a drop in November. It is possible that the log turner configuration was changed during these months, or that the system went through some technical problems. Since only one year of data is available, it is impossible to determine whether a yearly cycle repeats itself or not, and therefore it is hard to draw any conclusion about the seasonality of errors.

From a day of the week perspective there is no clear pattern; from month to month, the influence of the day of the week changes.

The second part of the `delta_angle` analysis consists of identifying variables that may explain its variations. It is reasonable to believe that both machine-related parameters and log characteristics can have an influence on the rotation error. Pearson correlation is used to determine which variables have the highest potential to predict `delta_angle`, both in nominal terms (shown on Figure 4.5) or in absolute terms (as seen on Figure 4.6). The majority of correlation coefficients for both the nominal and the absolute errors are very small, which means that there is at most a fairly weak linear relationship between these variables and the response variable in the dataset. There are only a few variables whose absolute value of

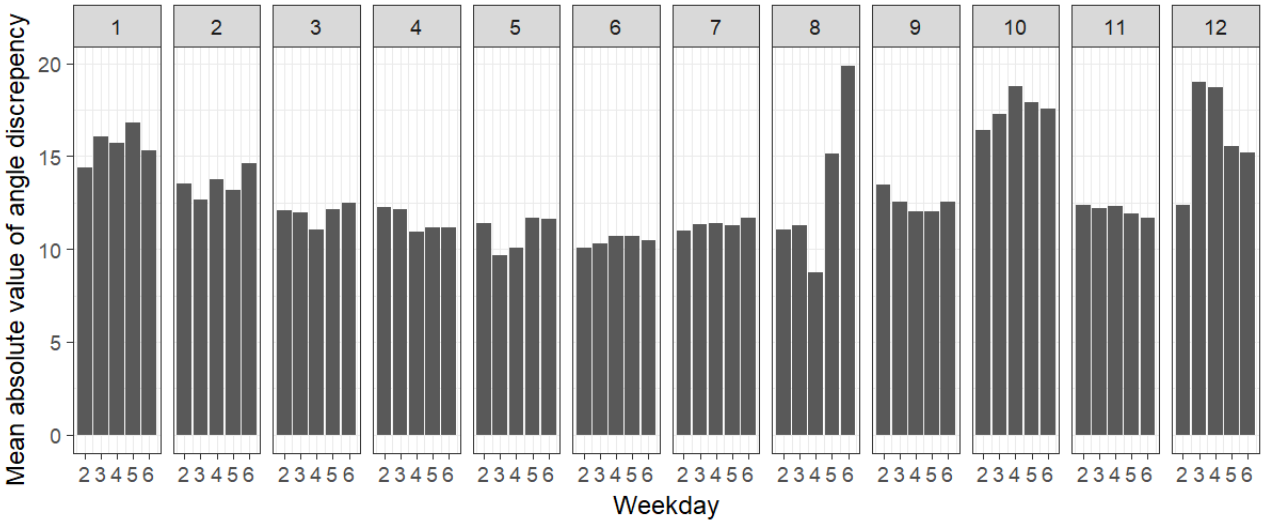


Fig. 4.4: Daily mean absolute value of `delta_angle` by month (1 = January) and day of the week (2 = Monday).

the correlation coefficient is larger than 0.1, which is still a weak relation. For nominal errors, these variables are: `angle_solution_rotation` (the optimal turning angle reported by the optimization algorithm), `turning_distance` (the translation of an angular measure to a length of movement for plates turning the logs), and `x_axis_offset` (offset of the log from the x-axis). Variables that are the most correlated to absolute errors are `turning_diameter` (diameter of turning) and `y_axis_offset` (offset of log from the y-axis). The low number of efficient predictors and their rather weak correlation with response variables might indicate that models built to explain variability of turning errors will have a low explanatory power.

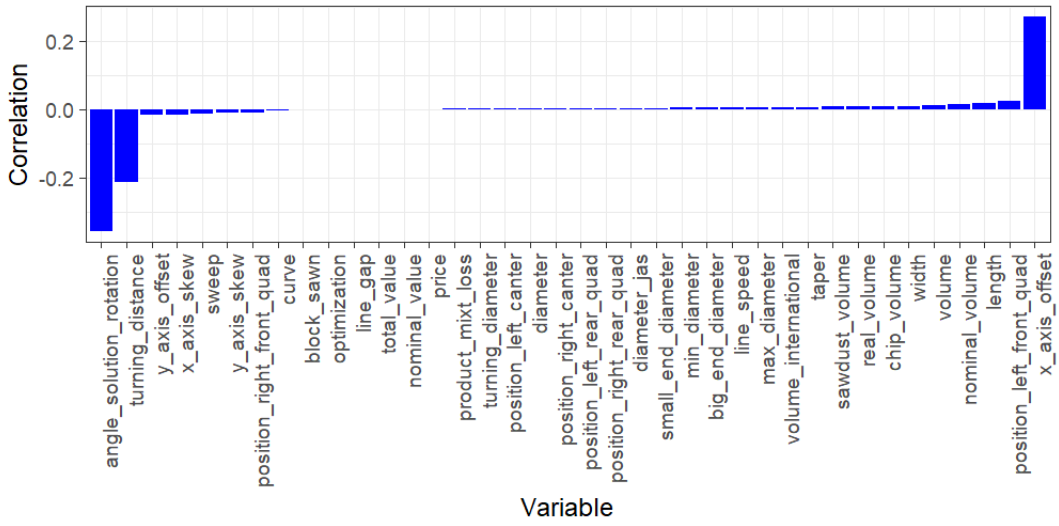


Fig. 4.5: Correlation between `delta_angle` and other variables for Scanner 1 data.

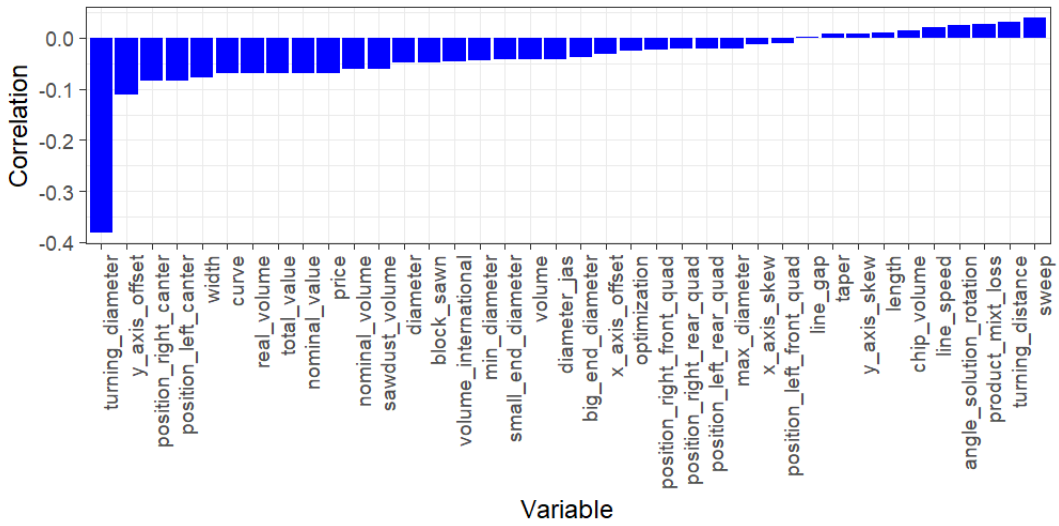


Fig. 4.6: Correlation between absoulte `delta_angle` and other variables for Scanner 1 data.

We can already note here that the negative correlation between `turning_diameter` and the absolute error of rotation in Figure 4.6 indicates that logs with a smaller diameter are more prone to large rotation errors.

The analysis of the correlations shows that `delta_angle` is linked to the `turning_distance` and `angle_solution_rotation` variables. Regression models show that there are days on which this relationship is quite strong. Figure 4.7 shows the scatterplot of these two investigated variables for the 29th of June 2016. This scatterplot looks very similar for other dates. Points on the plot are colored with the `delta_angle` magnitude. For any fixed value of the `angle_solution_rotation`, the `delta_angle` is increasing as the absolute value of `turning_distance` increases. In other words, the conditional correlation between `delta_angle` and `turning_distance` given `angle_solution_rotation` is positive. This means that large errors in `delta_angle` are more likely to materialize if the `turning_distance` recommended for a given log highly departs from the mean value proposed for other logs with a similar `angle_solution_rotation`.

4.2.3 *Price of Logs and Loss of Value*

The second important response variable is `price`, the expected value of the products that the optimal solution plans to extract from the log. More exactly, the change in `price` between scanners is very relevant. This variable can be used as a monetary measure of the loss resulting from cutting logs with the wrong angle. It could thus give an estimate of the potential gain arising from an improvement in the turning process. As mentioned above, there is no unique ID for each log across the scanners. To track the variation in estimated value, the data from the three scanners were matched based on their attributes. This procedure allowed to match 1,373,272 logs. The rest of the data were dismissed. Some summary statistics obtained from this matching procedure are presented in Table 4.2.

Table 4.2: Summary of matched logs

	Number of logs	Mean value of the logs	Total value of the logs
Scanner 0	1,373,272	14.97	20,569,645
Scanner 1	1,373,272	14.74	20,250,476
Scanner 5	1,373,272	9.57	13,154,643

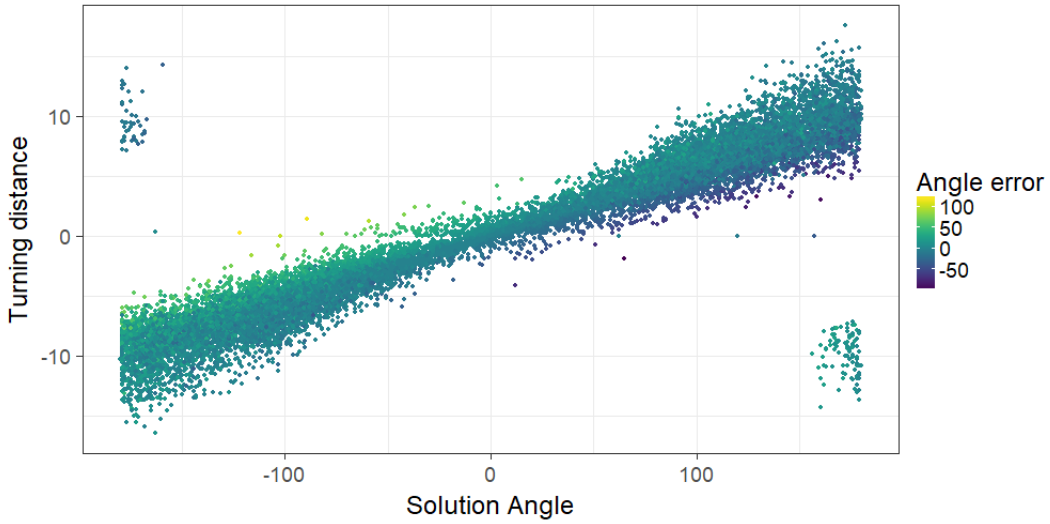


Fig. 4.7: Solution angle and rotation distance.

As seen in Table 4.2, the expected value of the logs decreases as they are evaluated by the different scanners, with the biggest drop occurring between scanner one and five. The average value of logs at Scanner 5 does not seem realistic and should not be considered reliable. The value of `price` at Scanner 5 was never used in any type of reporting, and its measurements were not verified. Therefore analyses of the value loss estimates were only conducted using data from Scanners 0 and 1. The difference between the total value of logs at the two scanners is about 320,000 CAD for 1.3 million of logs: thus for the full sample the total value of the loss should be estimated at approximately 600,000 CAD.

Let us define a new variable, `value_loss`, which corresponds to the difference in `price` between Scanner 0 and Scanner 1. Figure 4.8 shows the distribution of `value_loss` for the 1,373,272 logs. Although errors in rotation would typically be expected to translate into a loss of value, some logs see their value increase after an unsuccessful rotation. This surprising situation occurs because once the final position of the log is known, the optimizer uses it to determine the final cutting pattern. By the nature of optimization techniques and their requirements for computing resources, it is impossible to consider up front all possible rotations of the log, and sometimes the actual angle (different from the precomputed angle) may yield a slightly better solution than all those previously considered.

Figure 4.9 shows the daily average of `value_loss` across the year to detect possible cycles or seasonal effects. The general shape and trends in the plot are quite similar to Figure 4.3: thus it is likely that variables `value_loss` and `delta_angle` are linked.

Figure 4.10 displays average `value_loss` as a function of the days of the week and months, in order to see whether any weekly cycle can be observed. There does not seem to be a weekly trend that repeats month after month.

From this perspective, a very important question arises: how much of the loss in value can be attributed to errors in log rotations? The models presented in the next section will help answer this question.

4.3 Business Questions and Analyses

In this section, we focus on a few questions that were raised in the project description, or that emerged from the explanatory data analysis. The ultimate goal for each of these questions is to inform decisions on future developments and tuning of sawmills, through understanding the possible causes for errors and their

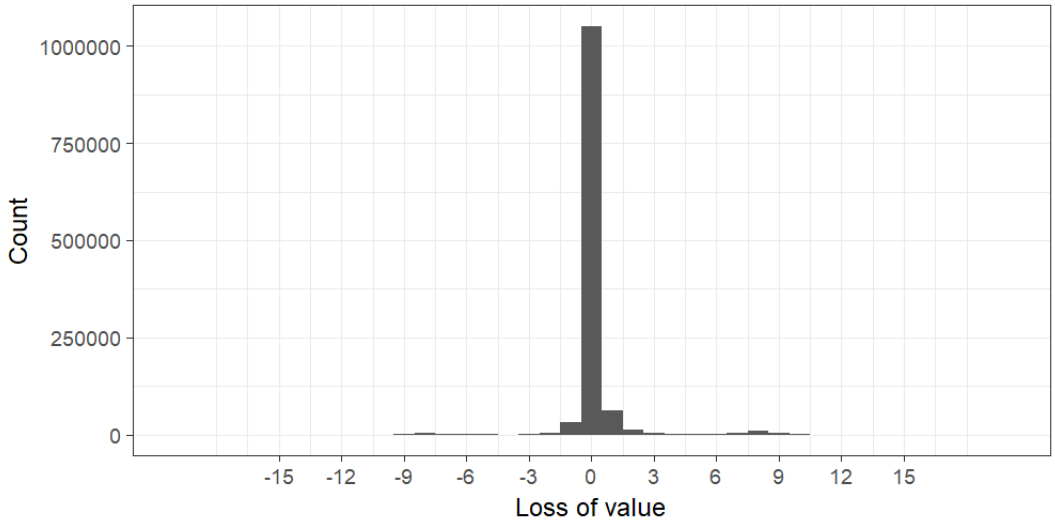


Fig. 4.8: Histogram of `value_loss`, the price change between Scanner 0 (before rotation) to Scanner 1 (after rotation).

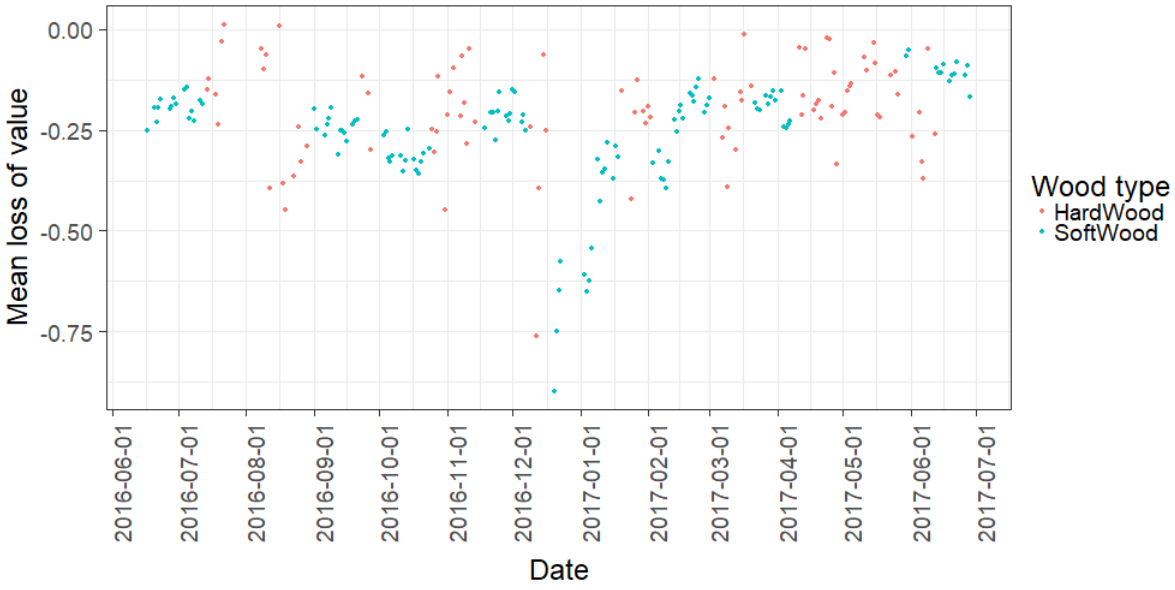


Fig. 4.9: Daily mean `value_loss` through time.

consequences on the value of the output. The scope of the results is constrained by the quality of the data, which exhibits some limitations. Namely, we ask the following questions:

- What portion of the loss in value of a log can be attributed to a rotation error?
- How well can the collected variables explain the angle rotation error?
- Can `delta_angle` or `value_loss` be predicted with data mining tools?

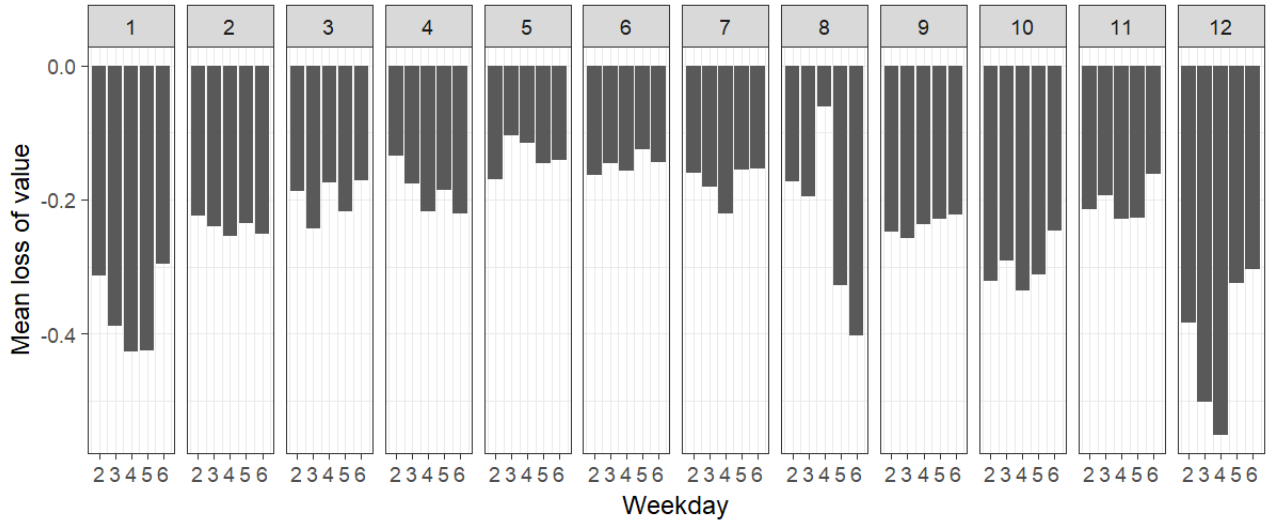


Fig. 4.10: Daily mean absolute value of `value_loss` by month (1 = January) and day of the week (2 = Monday).

4.3.1 *Explaining the Loss of Value*

This project is based on the assumption that improperly rotating a log reduces its value. Linear regression is used to measure how much of the variability in `value_loss` is explained by the rotation error (`delta_angle`). The absolute value of `delta_angle` is used since it should be the magnitude of the error rather than its sign that drives the loss. Surprisingly the regression has an R^2 of 0.01, which means that angular error has no ability to explain `value_loss`. Figure 4.11 shows a scatter plot of `value_loss` as a function of `delta_angle`. The absence of a link between the two variables is obvious, and larger values of `delta_angle` even tend to be associated with smaller losses. Adding other variables to the regression did not significantly change the results; the best estimated model (including more variables) has an R^2 of 0.05, which is still very small. In other words, we are unable to explain the loss of value.

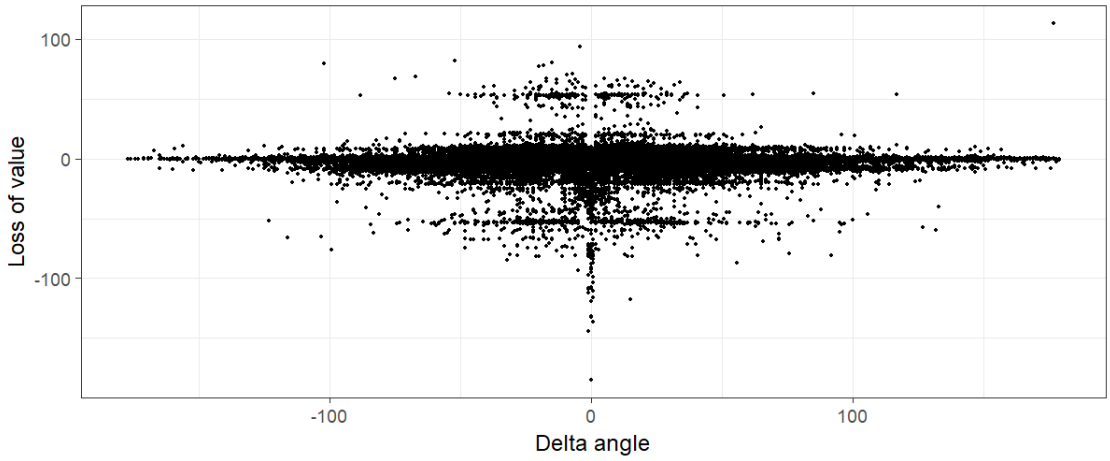


Fig. 4.11: Scatter plot of `value_loss` as a function of `delta_angle`.

We noted previously that:

- the total value of the logs at Scanner 1 is inferior to the total value at Scanner 0 and the only difference between these two steps is the rotation;
- Figure 4.3 and Figure 4.9 show similar patterns during the year for the daily averages of `delta_angle` and `value_loss` (respectively), meaning that these two variables should be correlated.

Both these remarks indicate that the total and mean value of the logs, whether on a daily or annual basis, seem to behave as expected. The individual price difference, however, behaves in unexpected ways and is difficult to explain. The simplest explanation is that the `price` values stored in the database are scrambled. For instance, if Scanner 1 saved the `price` of the previous log rather than that of the current one, we would still have very accurate averages for periods such as days or years, but any relationship at the level of individual logs would be lost and seem random, as it does now.

The process for determining the price of a log should be investigated carefully as it is very likely that there is an issue with the prices stored in the database. If there is truly no relation between `value_loss` and rotation error, then improving the rotation process will not yield monetary gains. It is clear, however, that the variables for determining `value_loss` are problematic: thus we keep working with the assumption that an incorrect rotation reduces the value of the logs, but any further modelling of `value_loss` seems futile until the problem is fixed.

4.3.2 *Explaining Rotation Error*

Understanding the link between `delta_angle` and the explanatory variables could provide insight and help find strategies to reduce the rotation errors. Linear regression is a simple method that allows the discovery of relationships in a data set. It may be used as a tool for inference to determine, for each variable, whether its link with `delta_angle` is real, or whether it could be due to chance, all while taking into account the contribution of the other variables.

Let us first consider a regression model where all the appropriate variables in Table 4.1 are used to explain `delta_angle`. The unusually large sample size is likely to yield many significant tests, but it is also relevant to look at how well `delta_angle` is explained. The R^2 statistic represents the proportion of the variability in `delta_angle` that is explained by the other variables. We obtain a value of $R^2 = 0.2$. Intuitively, an error of $+10$ or -10 degrees should have the same influence on value loss. Using the absolute value of `delta_angle` instead yields $R^2 = 0.195$. For the rest of this section, the nominal value of `delta_angle` will be used rather than its absolute value, but all models considered were also tried with the absolute value of `delta_angle` and lead to equivalent conclusions from a qualitative point of view.

Instead of keeping all the variables in the model at all time, we used model selection techniques to select a subset of variables that seemed to be the most important. We also included the square of all variables and their interaction terms to accommodate relationships that would not be linear. On the whole data set, this model gives $R^2 = 0.18$. Table 4.3 shows its estimated parameters and their significance.

The evolution of the R^2 is displayed in Figure 4.12. To draw that figure, the same regression model was fitted to the data of each production day. It seems that the relationship is very dynamic: there are periods where almost 60% of the variation in angle errors can be explained by the variables considered, but there are also periods where this relationship hardly exists. This might be the effect of measurement errors since the correlation between the daily R^2 of the models and the daily proportion of "fives" (logs with an angle error value of 5 or -5) is -0.78, indicating a strong negative correlation.

In our context, the residuals of the regression represent for each log the error that the model failed to explain. In other words, it represents the error that remains even after using the model. Figure 4.13 displays the average absolute residuals through time. Comparing it with Figure 4.3, we realize that the regression model did not manage to capture the variations that were described previously. Similarly, Figure 4.14 shows the average absolute residuals by month and day of the week.

Table 4.3: Results of the regression for angle error.

	Estimate	Std. Error	t value	p-value
Intercept	2.635e-02	2.472e-02	1.066	0.286
`angle_solution_rotation`	-1.442e-01	2.543e-04	-566.866	2e-16
`turning_distance`	1.725e+00	4.216e-03	409.208	2e-16
`x_axis_offset`	1.009e+00	6.478e-03	155.817	2e-16
`angle_solution_rotation` squared	1.656e-05	2.309e-06	7.171	7.46e-13
`turning_distance` squared	-4.601e-03	5.042e-04	-9.124	2e-16
`x_axis_offset` squared	2.201e-02	2.655e-03	8.289	2e-16
`angle_solution_rotation:turning_distance`	1.190e-05	3.325e-05	0.358	0.720
`angle_solution_rotation:x_axis_offset`	1.613e-03	1.491e-04	10.818	2e-16
`turning_distance:x_axis_offset`	2.850e-03	2.527e-03	1.127	0.260

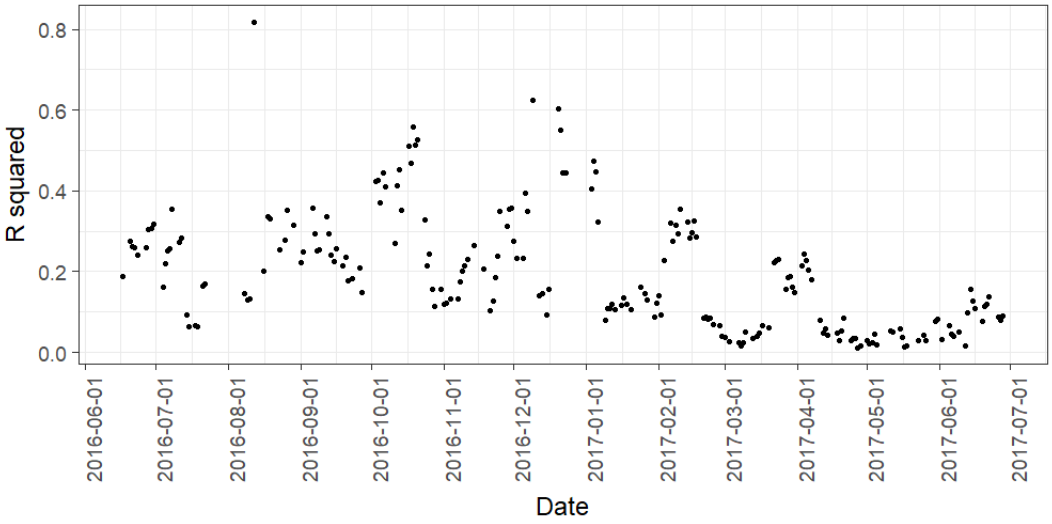


Fig. 4.12: Daily R^2 time series from the regression for angle error.

On the bright side, the variables that are used in the model are significantly linked to `delta_angle`. That link, however, is not as strong as expected. The angles stored in the database are estimated with scanners who provide noisy images of fast moving logs in a mill. There is certainly some degree of imprecision in those measurements and we have no information allowing us to estimate the magnitude of those errors. Suppose for instance that the scanners have a precision of ± 10 degrees, and other characteristics of the log are also derived from the scanner images. Even if the true values of the explanatory variables could predict the rotation error perfecly, the link between noisy measurements of these variables and the rotation error will be much weaker. Also recall that the magnitude of R^2 for daily regressions is negatively correlated with a known issue in the data (overabundance of ± 5). The most probable conjecture is that the variables at hand are able to explain rotation errors but the measurements at our disposal are too noisy to let us draw clear conclusions (or to make predictions that are good enough to be used in production).

4.3.3 *Predicting Error or Loss with Data Mining*

We initially thought that a data mining approach could be useful: if the rotation error can be predicted before the log is turned, then a predictive model could be added to the sawmill operation to account for the predicted error in advance. Sophisticated models can capture relationships between the variables beyond the

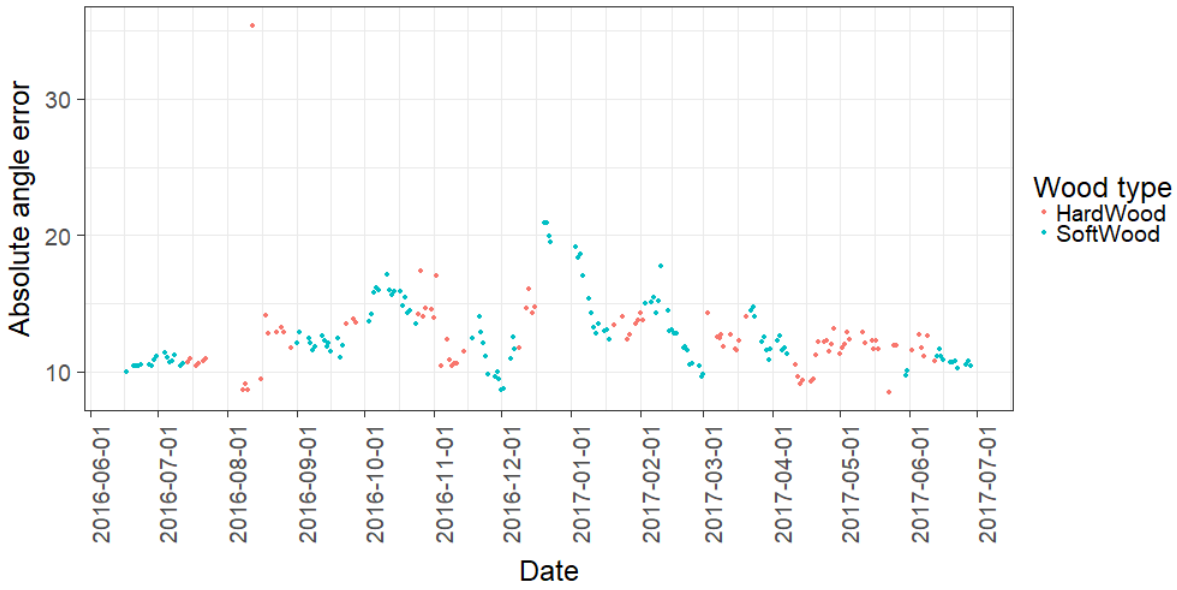


Fig. 4.13: Daily mean absolute value of regression residuals through time.

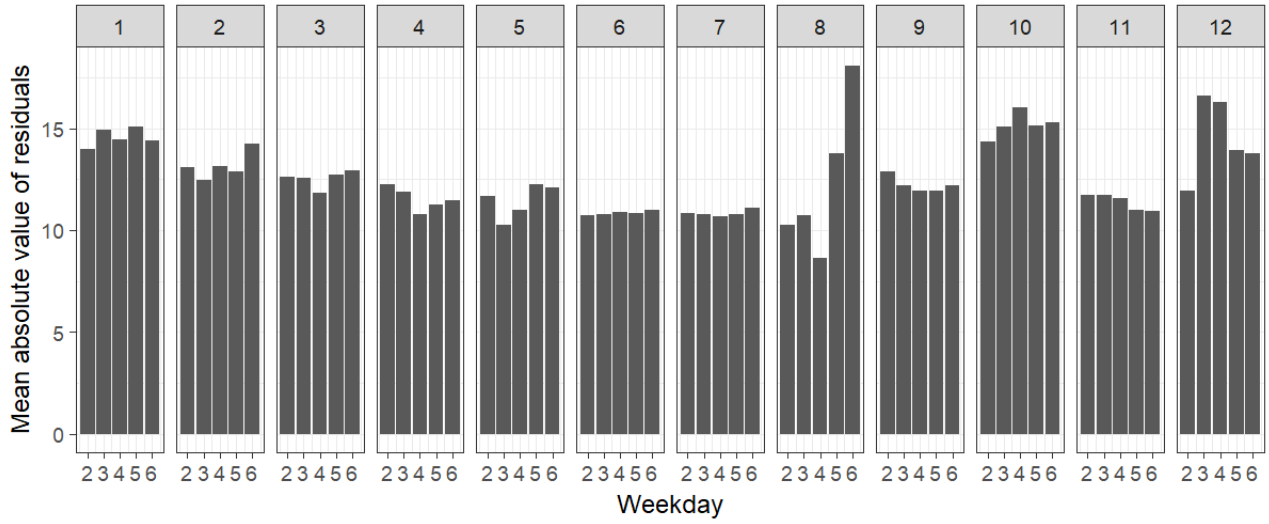


Fig. 4.14: Daily mean absolute value of regression residuals by month (1 = January) and day of the week (2 = Monday).

linear paradigm, but the low predictive abilities of linear regression (described in the previous sections) cast doubt on this strategy. Nonetheless, we did try different versions of decision trees and random forests, but these models did not add much to our conclusions – for both `delta_angle` and `price`. We believe that we are very far from a model that would be reliable enough to be embedded into the mill and to yield improved performance.

These findings stand in line with our exploratory analysis – if there is no clear and strong relationship between variables, then using sophisticated methods will typically be fruitless. This does not mean that the measured effects are unrelated. The presence of measurement errors, for instance, could hide existing

relationships. Fitting models with measurement errors is very similar to driving with a fogged windshield: even if there are things out there, you cannot quite figure them out, they are just too blurred.

4.4 Recommendations and Conclusions

To recap, we have studied how operational variables and characteristics of logs can affect the performance of a log turner and the resulting loss of value.

For the value, we observed that the average loss on many logs was as expected, but when we looked at the value of individual logs, all relationships between rotation error and value loss disappeared. We suspect that the values stored in the database may be scrambled.

For the turning error, we found significant links with a number of variables, but the strength of the links was weaker than expected. Estimating the angle of a log from pictures thereof is not easy, especially if it is symmetric. We suspect that measurement errors play an important role in diluting the strength of the relationships between variables.

Between Scanner 0 and Scanner 1, the log is rotated and no other operations are performed on it. For the mill we studied, the loss in value is approximately $600,000 over a 12.5 month period. It is unlikely that this full amount could one day be recovered, but the potential for increased profitability and reduced waste seems substantial.

We now formulate some recommendations:

- store a unique log ID in the database;
- evaluate the precision of the scanners at measuring angles. There may be a need for experiments with marked logs that are assessed while being processed at full speed;
- investigate the computation of the variable `price`;
- consider using additional information such as
  - log conditions such as temperature and humidity content (may require additional sensors);
  - maintenance logs;
  - additional information on the logs (location where they came from, time at which they were cut);
  - data from a longer period of time to detect potential annual or seasonal cycles;
- consider exploring other sources of data such as:
  - PCL records;
  - records from the 3D scanners (they could be reprocessed).

To sum up, getting data of higher quality is the key to increasing the odds of success in this project.

5
Registration of Hyperspectral Images of the Retina

Athmane Bakhta, Jeremy Budd, Farida Cheriet, Karl Deutscher, Faten M'hiri, Michael Lamoureux, and Hayley Wragg

Executive Summary The study group participants had the task of investigating retina imaging and trying to develop an effective way of registering the images and quantifying the quality of the registration. The investigation was carried out through computational modelling and mathematical analysis of the transformations.

The participants developed several approaches to the problem of image registration, including feature extraction, optimization, and optimal transport. The feature extraction approach used the company's code for extracting the features then tried to optimize the transformation taking one image to another. In the optimization approach the density differences between corresponding cells were optimized but extreme values were assigned to upper and lower bounds to prevent the average from being skewed. The optimal transport approach was investigated and reviewed and appears to suit the problem: there was not enough time, however, for a full investigation of this approach during the workshop. An additional suggestion for the problem was to change the hardware so that multiple images can be taken. This does not involve image registration but if implemented correctly would save the company the effort of registering the images.

The approach for evaluating the quality of registration was to analyze the smoothness of the spectral signature. This takes into account the requirement that the quantity evaluating the registration be independent from the optimizations used in the registration itself. Although there was not time to investigate this fully during the workshop, the results are promising.

The report concludes with recommendations to the company based on the results.

Athmane Bakhta
CERMICS

Jeremy Budd
University of Nottingham

Farida Cheriet
Polytechnique Montréal

Karl Deutscher
University of Alberta

Faten M'hiri
Optina Diagnostics

Michael Lamoureux
University of Calgary

Hayley Wragg
University of Bath

5.1 Introduction

5.1.1 *Company Background*

Optina Diagnostics is a company using retina imaging to help with early Alzheimer's diagnosis. The result of the early detection of Alzheimer's disease is that the patient can be treated before irreversible damage occurs. The impressive technology developed by Optina Diagnostics allows doctors to pick up early symptoms of Alzheimer's from the retina images.

The hyperspectral camera used observes the retina for 91 uniformly spaced wavelengths in the range of 900nm to 450nm. The data is recorded as a cube containing a frame projecting the retina for each wavelength. The problem with the result is that there is movement in the eye within the one to three seconds it takes to record the data. The movements include but are not limited to eyeball rotation and cardiac blood vessel expansion.

5.1.2 *The Experiment*

To take the images of the retina white light is shone on the eye. A rotating filter disk is placed inbetween the light source and the eye as shown in Figure 5.1. The filter disk controls which wavelength of light the eye receives and then a hyperspectral camera takes the images of the retina for each wavelength.

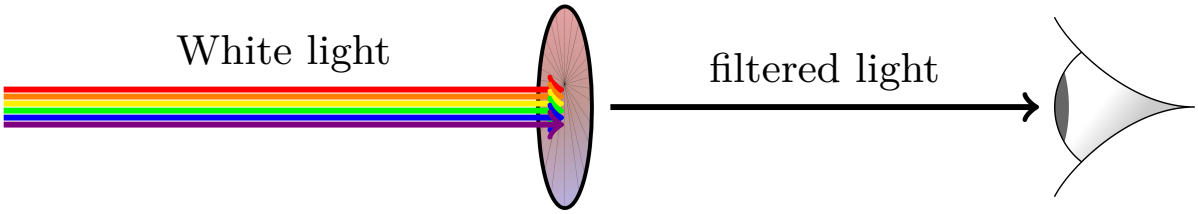


Fig. 5.1: Experiment Diagram

5.1.3 *Description of the Problems*

The aim of the project was twofold. The initial aim was to develop a method that could register these images with subpixel precision. One difficulty in the registration process is that blood absorbs light differently for the different wavelengths. As a result the intensity varies substantially from one image (corresponding to a wavelength) to another. The algorithm Optina Diagnotics had developed to tackle the problem was slow (it took 45 minutes to run in MatLab).

The second aim was to improve the assessment of the registration quality in order to compare different registration methods.

5.2 Literature Review

5.2.1 *Multispectral Imaging*

Multispectral imaging is an imaging technique in which an object is exposed to a wide range of electromagnetic frequencies. This provides detailed information about the way different tissues absorb, reflect, and fluoresce the various frequencies. This information can be used to infer biological facts such as the concentration of haemoglobin in a tissue [8], which is a key diagnostic tool for detecting cancer. Due to the delicate nature of the eye, diagnosis of retinal abnormalities relies heavily on optical techniques. For this reason multispectral imaging has recently been explored as an effective technique for determining oxygen saturation in the eye and detecting early signs of blindness [8].

5.2.2 *Image Registration*

Image registration is the problem of finding the transformation by which one image, the *source image*, must be deformed to yield a second image, the *target image*. There are two basic techniques for image registration: intensity-based and feature-based [12]. Intensity-based approaches look at minimizing the difference in intensity (e.g., ℓ^2 distance) or in the intensity gradient (or some other similarity measure) between the target image and the deformed source image [12]. Feature-based techniques, by contrast, discard most of the intensity information and instead identify (via some detection algorithm) sets of features in the source and target that are then mapped to one another. In the case of retinal applications, the vascular structure provides the predominant features [12].

5.2.2.1 Intensity-Based Similarity Methods

In an intensity-based method the source image and target image are compared globally. Generally speaking, the idea is to find an alignment of the source to the target such that a probability P is maximized, where P is the probability of corresponding pixels in the source having their intensities, given the intensities in the target (see [10]). This likelihood function is often simplified by assuming that these pixel-wise probabilities are independent. This yields a similarity measure that takes no account of spatial dependence.

In medical imaging, however, there are often spatially-varying intensity distortions [10], so this assumption needs to be refined. One common solution is to consider "local" similarity measures, on the assumption that the distortions should be approximately uniform over a small neighbourhood of a pixel. For example localized versions of the correlation coefficient (CC) and mutual information (MI) measures have been recently employed [10]. Other methods, such as that proposed by [10], attempt to register the images and simultaneously correct for the intensity distortion, assuming independence of the pixels once the intensity has been corrected. This gives rise to a similarity measure called Residual Complexity (RC), which is defined as

$$\sum_{n=1}^{N} \log(a_n^2/\alpha + 1),$$

where the (a_n) are the coefficients of the difference of the target image and transformed source image in some basis ([10] chose the discrete cosine transform basis) and α is a trade-off parameter.

A weakness of these methods is that there are a great number of local minima that must be avoided to achieve the correct registration (see [12, 10]). This necessitates the use of global optimization techniques such as Simulated Annealing and Genetic Algorithms ([9]), but these methods may involve large numbers of function evaluations ([12]).

5.2.2.2 Dual-bootstrap Iterative Closest Point

Dual-bootstrap Iterative Closest Point (hereafter denoted by DB-ICP) is a feature-based registration technique. The core idea is to start by registering a small region of the images, the “bootstrap" region, and then gradually expand this region, iterating the process until the entire image is registered. The phrase “dual-bootstrap” refers to the manner in which the transformation model (affine, quadratic etc.) at each step is chosen, based on the previous step. The algorithm can be described as follows [12].

(1) **Extract features** from the source and target images and identify particularly distinctive “landmark” features.
(2) Discover **initial correspondences** between landmark features. Choose one to initialize the algorithm.
(3) **Initialize the transformation** model to the simplest possible, and compute an initial transformation estimate. Establish the initial bootstrap region.
(4) (Dual-bootstrap iteration) While the estimate has not converged do the following.
   (a) **Estimate the parameter** of the transformation by minimizing an objective function depending on the bootstrap region, transformation model, and previous transformation using the ICP algorithm. **Calculate the covariance matrix** of this estimate. For details on the objective function, see [12]. The covariance matrix is found by taking the inverse Hessian of the objective function.
   (b) **Bootstrap the model**: Update the transformation model via a statistical model selection method. If the model changes recompute the estimate in the previous step.
   (c) **Bootstrap the region**: Use the covariance matrix and the new model to expand the bootstrap region.
      - The magnitude of the increase in bootstrap region size depends on the uncertainty in the transformation estimate. This is quantified by the covariance matrix.
      - In particular, the covariance matrix Σ of the transformation is used to compute the covariance of a transformed point p_i' by approximating it via the Jacobian of the transformation evaluated at p_i. This allows the computation of the “transfer error” at a transformed point. The bootstrap region is then expanded in such a way as to control this error within the transformed region.
   (d) **Check for convergence**, i.e., whether the bootstrap region is the whole image.
(5) If convergence has been achieved, the transformation estimate is sufficiently accurate, **terminate**.
(6) If the algorithm **fails to terminate** for each initial correspondence, then the images have **failed to be registered**.

5.2.2.3 Comparison of Techniques

In [7] various examples of these techniques were compared using data drawn from a synthetic eye (the synthetic test) and using deformed versions of real medical data (the semisynthetic test) drawn from [2]. The authors found that among the intensity-based approaches the similarity measures CC, MI, and RC all performed relatively well, with MI being sensitive to the degree of deformation and RC having low median error but high error variance.

Other measures such as ℓ^1 and ℓ^2 had a much worse performance. DB-ICP performed extremely well, achieving close to sub-pixel accuracy and very low error variance. A diffusion-based Maxwell demon method [13] had a very high variance in error and produced globally unrealistic transformations, but could register locally very accurately. The article concluded that DB-ICP was the best method among those considered.

5.3 Problem 1: Registration

5.3.1 *Considering the Intensity and the Transformation*

5.3.1.1 Difficulties with Intensity

To extract the noise from the images the differences between the images must be observed. To observe the changes between sequential images, the differences between the intensities of corresponding pixels are measured. The problem with directly comparing the images is that the different wavelengths yield different light absorption in the blood and differences may be detected where no movement actually occurs. To overcome this problem the mean of the images was used to rescale the images for comparison.

The resulting image was the cube defined by the formula below (where i represents a frame).

$$\text{For each frame:} \qquad f_2^i = |(|f^i - O * f^i_{mean}| - |f^{i-1} - O * f^{i-1}_{mean}|)| \quad \forall i = 2, ..., 91.$$

In this formula f is the frame from the original image, f_2 is the comparison frame, and O is the matrix consisting of ones.

Although the algorithm was carried out on a synthetic eye that did not move, noise was observed. This implies that the equipment is producing the noise on the image: we must take this into account. The noise observed appears to move through the frames according to an unclear pattern. Figure 5.2 shows the noise (which appears to be rotating).

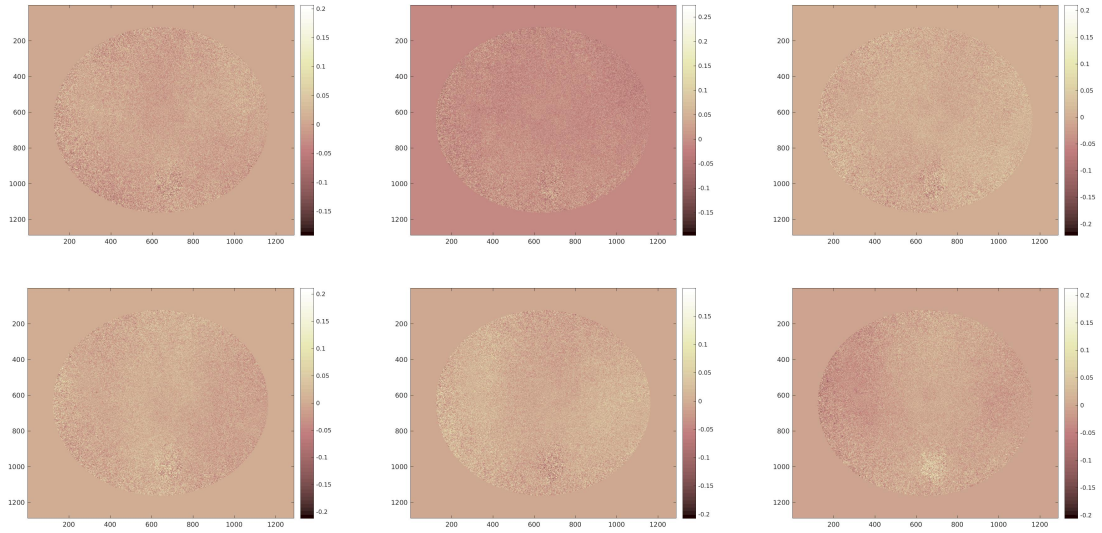


Fig. 5.2: The synthetic eye for consecutive wavelengths

5.3.1.2 Transformation Choice

Image registration includes the problem of finding an appropriate transformation from a source image to a target image. In order to reduce this optimization problem to a finite-dimensional one, we consider a parametrised family of transformations. Our choice of transformation was guided by a consideration of the geometry of the eye: the eye is not flat, so it makes little sense to look for Euclidean rotations and translations.

The registration code currently employed by Optina, and the DB-ICP method described above, choose the family of *quadratic transformations*. This choice is justified mathematically in [4], which approximates the surface of the retina through a quadratic surface and argues that the motions of the retina are predominantly rigid motions such as rotations. Suppose we have two images of such a surface, expressed (respectively) in the coordinates (X, Y, Z) and (X', Y', Z'). The quadratic assumption is that the formula

$$Z = aX^2 + bXY + cY^2 + dX + eY + f$$

holds and the assumption of rigid transformation is expressed as

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix},$$

where $R = (r_{ij})_{1 \le i,j \le 3}$ is an orthogonal rotation matrix. The authors of [4] show how to derive from these assumptions the existence of a quadratic map from the 2D coordinates (x, y) associated with the (X, Y, Z) coordinates (used in the first image) to the corresponding (x', y') 2D coordinates (for the second image).

The approximately spherical geometry of the eye, however, suggested that the appropriate transformations were likely to be *Möbius transformations*, in particular, the following transformations.

$$z \to z + \frac{az + b}{cz + 1}, \quad z \in \mathbb{C}$$

This transformation was chosen because the Möbius transform $(az + b)/(cz + d)$ corresponds to a rigid transformation of a sphere conjugated with the stereographic projection between the sphere and the complex plane. Hence these transformations are appropriate because they respect the roughly spherical shape of an eyeball, assuming that the deformations primarily result from rigid spherical motions. The $d = 0$ case creates a singularity, hence rescaling can be used to ensure that $d = 1$ holds. This transformation also has the desirable property of requiring fewer estimations of parameters than in the quadratic case.

5.3.2 *Registration Technique* 1: *Feature Extraction Approach*

The variation in intensities between the images, combined with the moving noise, makes it very difficult to use a simple residual approach to solve this problem.

The location of the blood vessels is the key indicator of the rotation in the eye since their positions are fixed relative to the eye. Using feature extraction to detect the blood vessels simplifies the transformation detection. The transformation mapping the sequential feature image should in theory correct any rotational movement in the eye. *Note that this does not correct for the expansion and contraction of the vessels due to the cardiovascular cycle.*

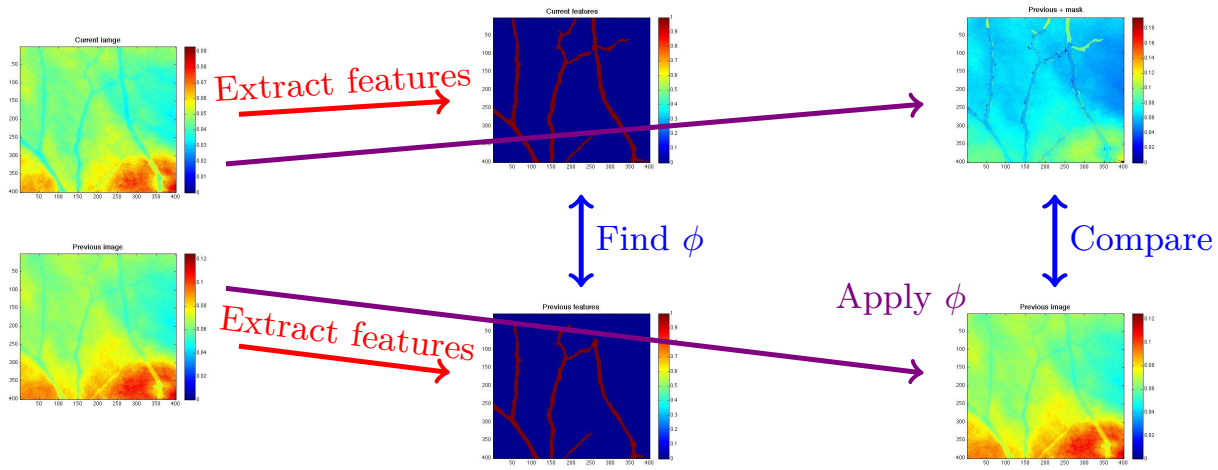


Fig. 5.3: Feature Tracking Diagram

Here are the steps of the feature tracking algorithm.

(1) **Extract the features** of the images. The feature extraction used in Figure 5.3 was carried out with a program written by Optina Diagnostics. Further research into this topic could improve the speed of the algorithm.
(2) **Find the transformation** from one feature image to the other using a simple optimization approach on the residual. A Nelder–Mead optimization was used in Figure 5.3 but other methods could be used.
(3) Apply this **transformation to the original** detailed image.

The results appear promising but there was not enough time within the workshop to investigate this approach further.

5.3.3 *Registration Technique* 2: *Optimization Approach*

5.3.3.1 The Key Idea

In the optimization approach we use an optimization method to minimize the ℓ^2 norm on the residual of sequential images. We can then determine the parameters of the **Möbius transformation**. The transformation is the following.

$$z \to z + \frac{az + b}{cz + 1}$$

5.3.3.2 Test Image

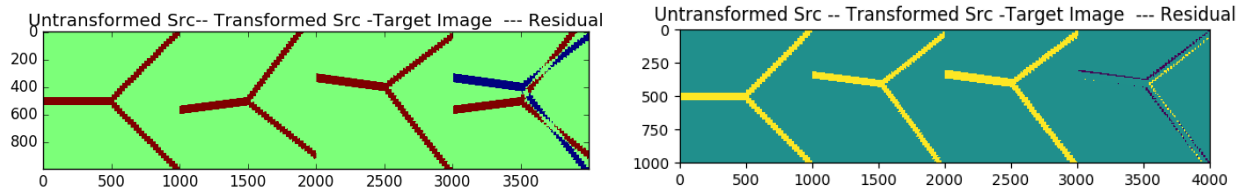
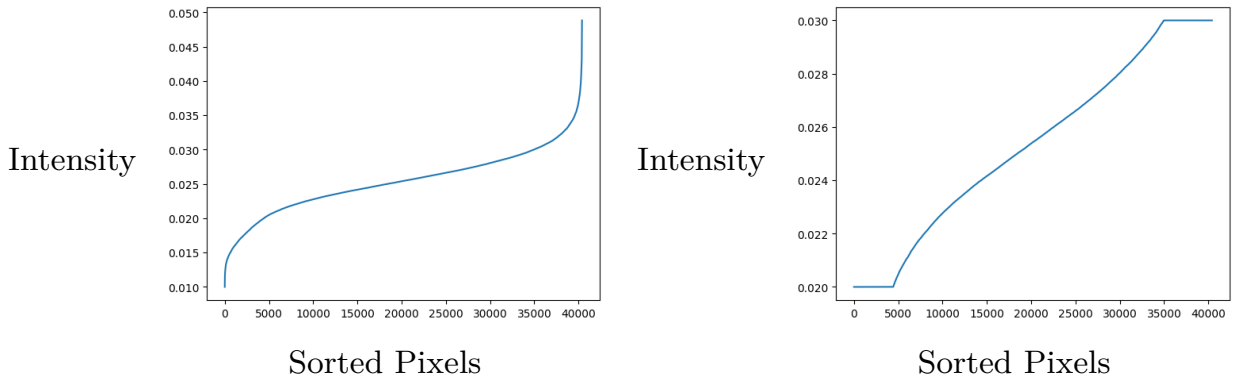


Fig. 5.4: Attempt to locate the transformation by eye and by optimization

In Figure 5.4 a Nelder–Mead optimization is used to minimize the ℓ^2 norm and to try and locate the transformation between two sample features. This is compared to an attempt to find the transformation by eye. The results clearly show that the optimization method is much better at locating the transformation.

The wide range of intensities in the real images causes a problem when using this optimization method. By smoothing out the extreme values, this range is made smaller but the key change at the location of the features is preserved.

Figure 5.5(a) shows there are a few pixels (approximately 5% of pixels) that are very bright; this shows up as a cluster of really large values. Similarly, there are approximately 5% that are very dim, also showing up as a cluster of really small values. Smoothing prevents the optimization method from spending time trying to match the mean and the extreme values. To correct for this, the 95th percentile and the 5th percentile are identified and the intensity values bounded within these regions. This is shown in Figure 5.5(b). The mean is then subtracted to account for the intensity variations caused by the wavelength changes.



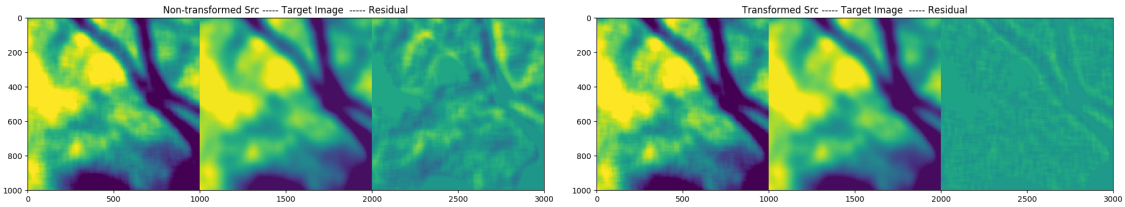
(a) Sorted intensity of pixels.

(b) Sorted intensity of pixels after smoothing

Fig. 5.5: Pixel intensities before and after smoothing

Here are the steps of the variational approach algorithm.

(1) Divide the image into **subregions**.
(2) Smooth the image to **eliminate extreme values**.
(3) Use an optimization algorithm to **locate the parameters** of a Möbius transformation.



(a) Source and Target images after smoothing

(b) Optimization of the transformation on the smoothed images

Fig. 5.6: Optimization of transformation using smoothing

Figure 5.6 displays the smoothing applied to a subregion of a retina image and then the optimization performed on the smoothed outcome. By eye the residual seems to show that the features have been matched well by this method.

5.3.4 *Runtimes*

Optina Diagnostics feature extraction is written in MatLab. When finding the transformation between the feature images, the code takes less than five minutes when the image is partitioned in a mesh and corresponding pieces are compared. Reapplying this to the original image takes a similar amount of time. The overall process takes 45 minutes to run. In theory this could be improved by parallelizing the code and using a more efficient language such as Julia or C.

The optimization approach was written into Julia from the start. The smoothing takes approximately five minutes to perform; the optimization, however, takes approximately 14 minutes. It is apparent that the time spent optimizing on the intensity images is the biggest contribution to computational time. The total run time for smoothing and optimization on one section of the image is 1168.576433125 seconds. The section takes up 1/49th of the total image but theoretically all sections could be processed in parallel and then a block matrix could be found for the transformation. The total time for optimizing on all 91 images is therefore approximately 19 minutes.

The feature extraction is slower than the optimization approach but it is written in MatLab. In future work one should consider rewriting the feature extraction code in a more efficient programming language.

5.4 Problem 2: Quantifying Accuracy

The above methods give algorithms for registering a pair of retinal images. Beyond this, the company required a quantification of the accuracy of the computed registration. A natural measure of registration quality is the minimal value of the objective function. Optina, however, were concerned about the possibility of a low objective function value not necessarily corresponding to an accurate registration.

Therefore we seek an independent measure of registration quality. Ideally this ought to be “invisible” to the optimization approach, so that an inaccurate registration is unlikely to perform well on the measure. A measure that seems to fit these criteria is the smoothness of the spectral signature.

5.4.1 *The Spectral Signature*

The spectral signature (at a pixel) is the behaviour of the intensity at that pixel as a function of wavelength. Figure 5.7 displays an example of spectral signature taken from data provided by Optina, registered using their current algorithm.

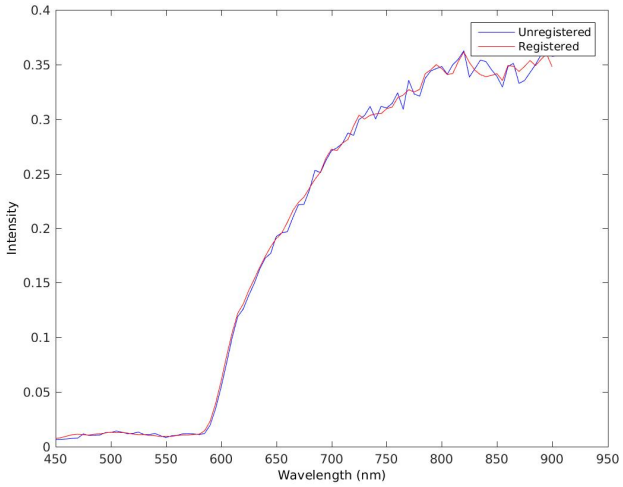


Fig. 5.7: Plot of registered and unregistered spectral signatures, from data provided by Optina

The spectral signature is an important diagnostic tool in multispectral imaging, as its shape can for example indicate the quantities of Oxygen in the blood at that blood vessel [6, 8]. Since the signature is produced by the spectra of the compounds in the blood, simulations of which can be seen in Figure (1) of [2], one might expect it to vary smoothly. This expectation is further justified by the spectral signatures exhibited in Figure (4) of [6], and by the observation that there are fewer sudden jumps in the registered spectral signature than the unregistered one. Hence if non-smooth behaviour in the signatures is an artefact of movements of the eye, then a good registration will remove these artefacts and increase smoothness. There is also some genuine non-smoothness at around 580nm caused by the light beginning to penetrate beneath the retina [5], so we exclude frequencies smaller than 600nm in the subsequent discussion.

5.4.2 *Quantifying Smoothness*

If we are to use smoothness as a measure, we will need to devise a good way of quantifying the increase in smoothness induced by our registrations. Furthermore, we will need to make sure that this measure does actually demonstrate improvement when registration is applied to our data.

5.4.2.1 Magnitude of the Gradient and Auto-Correlation

The simplest measure of smoothness would be to consider how much the signature changes with small changes in wavelength, that is, to record the signature change between each pair of consecutive wavelengths. As this is an approximation to the gradient of the signature, this could give rise to a measure of the smoothness

such as the *total variation*. One could also take a less global norm: e.g., consider something closer to the ℓ^∞ norm of the gradient instead of the ℓ^1 norm.

A more sophisticated application of this idea would be to quantify smoothness in terms of the *auto-correlation* of the signature. This measures the similarity of a series with a delayed version of itself, as a function of the delay, by taking the ℓ^2 inner product of the series with the series shifted by the delay. This amounts to computing

$$R(n\delta\lambda) = \sum_{i=0}^{90-n} f(\lambda_i) f(\lambda_{i+n}),$$

where f is our signature, $\delta\lambda$ is the step size between wavelengths, and $\{\lambda_i\}_{i=0}^{90}$ are the sampled wavelengths. Consideration of this value for numerous small delays could provide a better measure than just looking at a single delay as one would when considering the gradient.

We experimented with this measure by computing the differences over a single time-step for the data in Figure 5.7. Figure 5.8 below shows the differences for the registered and unregistered data, as well as for unregistered data artificially smoothed with a moving average. We clearly see a decrease in magnitude after applying registration, and a large decrease for the very smooth data indicating that it is indeed smoothness we are measuring.

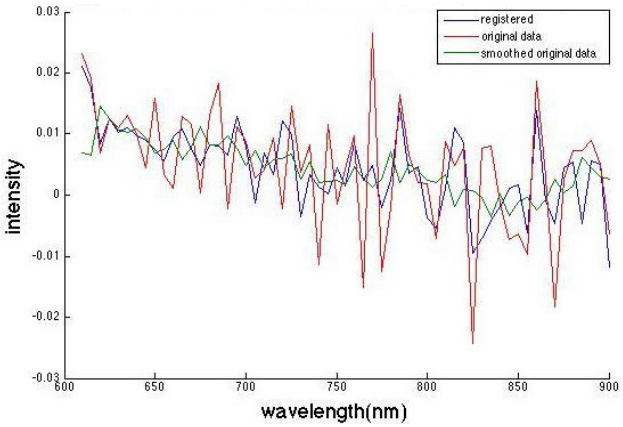
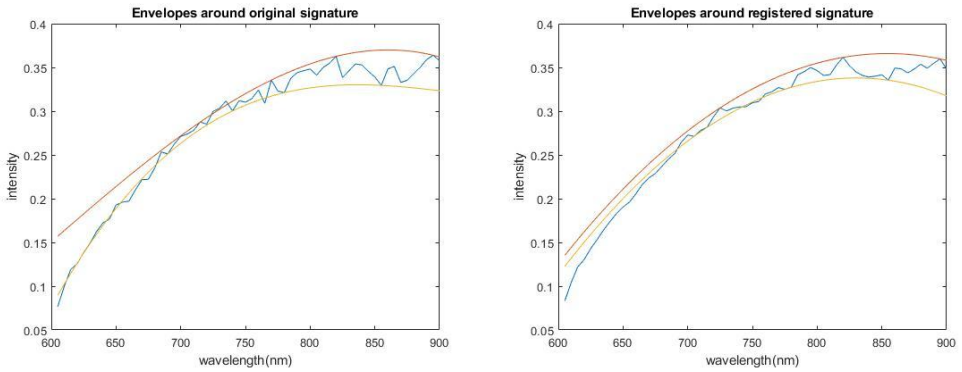


Fig. 5.8: Plot of differences over a single step for registered, unregistered, and smoothed data

5.4.2.2 Envelope Width

A second measure we considered is obtained by putting a smooth envelope around the signature. The size of this envelope (e.g., its widest diameter) decreases as the function gets smoother, and thus provides a smoothness measure.

We experimented with this measure by using the MatLab envelope() function to create envelopes for the data from Figure 5.7. The resulting envelopes can be seen in Figure 5.9. Again it is clear that registration does reduce the size of the envelope, and furthermore improves the fit of the envelopes to the curve.



(a) Unregistered signature (b) Registered signature

Fig. 5.9: Plot of smooth envelopes around the unregistered and registered signature.

5.5 Conclusion and further approaches

5.5.1 *Other Approaches*

5.5.1.1 Multi-Image Registration

The registration methods we have considered so far concern registering a pair of images accurately. Our problem, however, concerns registering 91 images. This raises new challenges, in particular the propagation of registration errors. Thus we seek a consistent set of registrations, so that for example the same feature across multiple images is always transformed to the same feature in a reference image.

This problem has been extensively explored in the context of registration of mosaic retinal images. In [3] a technique is described for registering multiple images of the retina, with only partial overlap between different images, in an attempt to image accurately a larger retinal area than would be possible with only one image. This use of multi-image registration reduced registration error from 1–3 pixels to subpixel accuracy. The idea is to solve an optimization problem that finds a sequence of registrations of all the images at once, with the coherence of this sequence forming part of the constraint set of the problem.

Adapting this technique for the purposes of our problem, we denote our images by $\{I_0, \dots, I_{90}\}$. We shall choose (WLOG) I_0 to be the reference image, and shall seek a sequence of transformations $\Theta = (\theta_1, \dots, \theta_{90})$ where θ_n is a registration of I_n with I_0. We shall suppose we can find pairwise registrations $\phi_{m,n}$ between I_m and I_n by the above methods, and find associated confidences $p_{m,n} \in [0,1]$ that quantify the accuracy of $\phi_{m,n}$. We then find Θ via the following algorithm.

(1) Identify frames that are very similar (e.g., I_m and I_n for m, n close together), which we call neighbours, and form a graph $G = (V, E)$ with edge set $E = \{(m,n) | I_m \text{ and } I_n \text{ are neighbours}\}$. If G is not a tree then we may wish to replace it with a spanning tree T.
(2) Initialise $\Theta = (\phi_{1,0}, \dots, \phi_{90,0})$. Also compute $\phi_{m,n}$ and $p_{m,n}$ for $(m,n) \in E$.
(3) Find Θ minimizing

$$\mathcal{E}(\Theta) = \sum_{m=1}^{90} p_{m,0} \|\theta_m(I_m) - I_0\|^2 + \sum_{(m,n)\in E} p_{m,n} \|\theta_m(I_m) - \theta_n(\phi_{m,n}(I_m))\|^2. \tag{5.1}$$

The first term in this sum is a measure of the accuracy of each θ_m as a registration with respect to the reference image. The second term enforces consistency among transformations, as it requires that identical structures in neighbouring images are mapped to the same structure in the reference image.

5.5.1.2 Hardware Modification

The following approach assumes that a single image can be taken in less than 10ms and that the resulting image would have negligible distortions. This approach suggests modifying the hardware so that every image for the 91 different wavelengths can be taken at the same time.

Let each light source be independently controlled and consist of a white diode, whose intensity can be controlled and modulated.

The intensity of the ith diode is denoted by

$$d_i(t) \quad \text{for } i = 1, \ldots, 91.$$

A colour filter corresponding to each wavelength is placed in front of each diode as in Figure 5.10. All light beams are then projected as a single beam using standard optical techniques. This beam is shined on the retina and the image is taken; the camera must be able to take 10,000 or more frames per second.

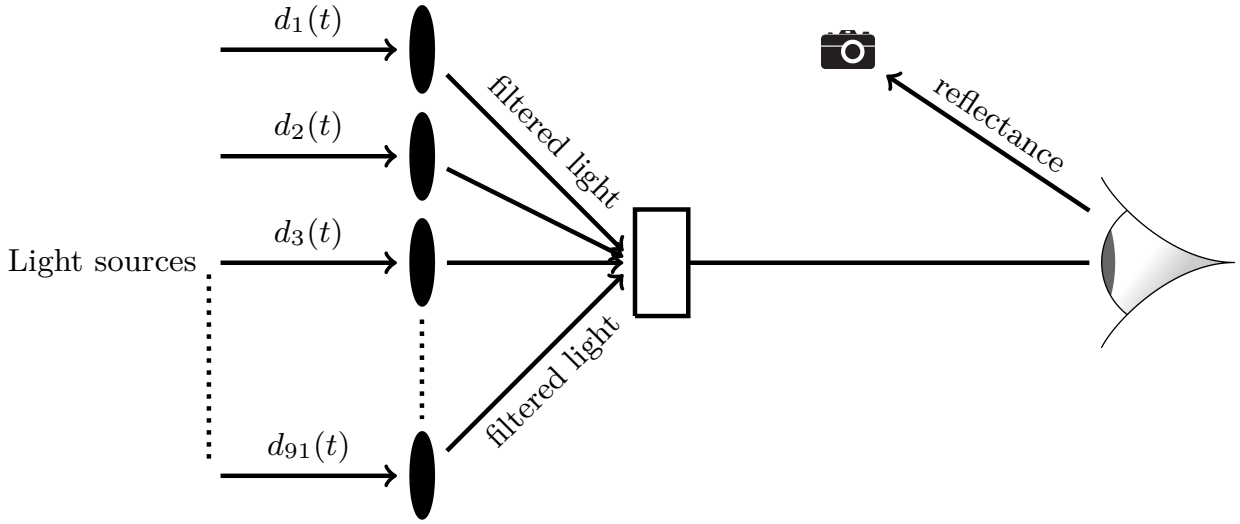


Fig. 5.10: Diagram of the proposed hardware change

The following steps can then be used to separate the light from the single output for each colour.

Denote the intensity of the captured light for a single pixel by $c(t)$, and the intensity for this pixel corresponding to the ith wavelength by $c_i(t)$. Modulate the intensity of the light source $d_i(t)$.

This can be achieved by using a sinusoidal wave as in

$$d_i(t) = A \sin(2\pi f_i t), \tag{5.2}$$

where f_i is the frequency of modulation, of the order of 1kHz. To obtain c_i a lock-in technique is used, Amplifier, which uses the orthogonality of the sine functions with different frequencies.

$$c_i(t) = \frac{1}{A}\int_{t=0}^{T} d_i(t)c(t)dt \tag{5.3}$$

Here are some comments on this method.

- For a frequency f_i close to 1kHz, the integration time is only a couple of milliseconds.
- Proper integration and calibration could yield accurate results if the f_is were only one or two Hertz apart from each other.
- It is important to identify the frequencies admitting the most noise from the environment and avoid them. It is usual for images corresponding to 60 Hz (or 50 Hz) and their odd multiples to be noisy, because electricity outlets run at those frequencies. More investigation is needed to determine any environmental contributions to noise.
- One strength of this technique comes from the fact that the frequencies and phase in d_i and c_i are matched. Calibration in the system is needed to find the best phase shift between c_i and d_i, but this only needs to be done once as long as the distances are fixed. Recall that the wavelength of light at 1kHz is 300 km, so the phase shift comes primarily from the delay in the processors.
- In general, having 10 data points per wavelength is enough to produce accurate results. The integration time and frequencies can be altered to manipulate the number of frames per second. For a 1kHz modulation frequency, that means 10,000 frames per second.
- The processing can be completely parallelized in order to keep the processing time to a minimum. The process that modulates the light source d_i computes the integration, this is independent from all other processors.
- This technique is well known and used (for example) in lock-in amplifiers. Lock-in amplifiers are expensive but once good frequencies are found, they don't need to be changed again and further purchases can be avoided. More details about lock-in amplifiers can be found in the user manual [11].

5.5.1.3 Optimal Transport

The image registration problem seeks to find the transformation (T) from one image (I_1) into another (I_2). Because of the change in the blood vessels during the cardiovascular cycle and because of the noise from the equipment, it is difficult to find a map such that $T(I_1) = I_2$. So an optimal map is sought that minimizes some measure of the difference between I_2 and $T(I_1)$, where each image is represented as a collection of densities (one density for each pixel).

Mathematical Formulation

Let I_1 denote the first image and I_2 denote the second image. In our case, if the resolution of the images $N_1 \times N_2$ is the same, then $I_1, I_2 \subset \mathbb{R}^2$ holds. Define the densities $f, g : I_1 \to [0,1], I_2 \to [0,1]$ such that for every $1 \le i \le N_1$ and $1 \le j \le N_2$, the quantity $f(x_{ij})$ denotes the intensity of the first image at x_{ij} and similarly for $g(x_{ij})$ in the second image.

Consider the loss function

$$c : (x, y) \in I_1 \times I_2 \mapsto c(x, y) \in \mathbb{R}_+,$$

where c encodes the cost needed to transport the intensity $f(x)$ to the pixel intensity $g(y)$. This is our measure of the difference between the corresponding pixels. Typically, c is the Euclidean or the H^1 norm.

The aim is to find the "transport map" T between f and g, where T exists. In other words, solve the optimization problem

$$T \in \arg\min_{\mathcal{M}} \mathcal{J}(T), \quad \text{where } \mathcal{J}(T) = \frac{1}{2}\int_{\mathbb{R}^2} |x - T(x)|^2 f(x)dx,$$

where $\mathcal{M}$ is the set of maps of the form

$$\int_{\mathbb{R}^2} \phi(y)g(y)\,\mathrm{d}y = \int_{\mathbb{R}^2} \phi(T(x))f(x)\,\mathrm{d}x, \quad \forall \phi \in \mathcal{C}^1(\mathbb{R}).$$

Monge–Ampère Equation

In the particular case of the L^2 metric, it is known (see [1]) that the optimal map T exists and is the gradient of some convex potential, i.e., $T = \nabla\Psi$ holds, where $\Psi \in \mathcal{C}^2$ holds and Ψ is convex. After some manipulations, we get the Monge–Ampère equation.

$$g\big(\nabla\Psi(x)\big)\det(\nabla^2\psi) = f(x), \quad \forall x \in \mathbb{R}^2 \tag{5.4}$$

It suffices then to solve the Monge–Ampère equation (5.4) to obtain Ψ and obviously T.

Numerical Methods

The registration problem is thus reduced to solving the PDE (5.4). It has to be noticed that this PDE is strongly nonlinear. Nevertheless, a wide family of numerical methods are available to solve it. Existing literature shows that a Newton solver is an efficient way to do this.

Key Points

- Take two (successive) images I_1 and I_2 defined on $\Omega \subset \mathbb{R}^2$.
- Introduce the densities $f, g\colon \Omega \to [0,1]$, where $f(x_{ij})$ denotes the intensity of I_1 at pixel x_{ij} (and the same for g and I_2).
- Consider the loss function

$$c\colon (x,y) \in \Omega \times \Omega \mapsto c(x,y) \in \mathbb{R}_+,$$

encoding the cost associated to the **transport** of $f(x)$ to $g(y)$. Typically this function is the Euclidean L^2 or the Sobolev H^1 norm.
- Solve the problem

$$T \in \arg\min_{\mathcal{M}} \mathcal{J}(T), \quad \text{where } \mathcal{J}(T) = \frac{1}{2}\int_{\mathbb{R}^2} |x - T(x)|^2 f(x)\,\mathrm{d}x$$

and $\mathcal{M}$ is the set of maps of the form

$$\int_{\mathbb{R}^2} \phi(y)g(y)\,\mathrm{d}y = \int_{\mathbb{R}^2} \phi(T(x))f(x)\,\mathrm{d}x, \quad \forall \phi \in \mathcal{C}^1(\mathbb{R}).$$

- T is the gradient of some convex potential, i.e., $T = \nabla\Psi$ holds, where $\Psi \in \mathcal{C}^2$ holds and Ψ is convex (see [1]).
- Numerically solve (5.4) (Newton solver, etc).

5.5.2 *Summary of Study Group Work*

This report highlights the complications for registering retina images taken at different wavelengths and assessing the quality of the registration. The intensity difficulties and the complicated geometry (Section 5.3.1)

have been investigated and approaches to dealing with this problem have been described. The group took two different approaches to optimizing the images: the first method looked at extracting the features first (Section 5.3.2). A Nelder–Mead transformation was then used to locate the parameters of the Möbius transformation mapping the source image of features to the target image of features. The second approach consisted of optimizing the transformation matching sections of the images to each other (Section 5.3.3). To assess the quality of the registration, analysis of the spectral signature was considered (Section 5.4).

Both approaches to the registration problem had good results in terms of finding the transformation but consumed a lot of time. To speed up computation Section 5.3.3 considers smoothing the extreme values, whereas Section 5.3.2 extracts the features, then works with these feature images. The feature extraction code that we used was fairly slow and the construction of the code has not been investigated. Run times could be shortened by improving the efficiency of the code structure and rewriting it into a more efficient programming language such as Julia, C, or Python.

Assessing the quality of the registration using the spectral signature (as considered in Section 5.4) appears to be a good idea. The independence of the spectral signature with respect to the optimization function improves its reliability as a measure of registration quality.

In Section 5.5.1 some other approaches to the problem have been proposed. These are approaches that could be of interest to the company but could not be investigated during the study group (for lack of time).

5.5.3 *Recommendations to the Company*

(i) As shown in Section 5.3.4, the company should consider rewriting its feature extraction code in a more efficient programming language in order to improve the run times for image registration.

(ii) The quality of the registration can be improved by using an optimization method to compute the parameters of a Möbius transformation (as opposed to a quadratic transformation). Using feature extraction may work well if the feature extraction code can be written more efficiently. Alternatively the company could consider optimizing on the image subregions directly but use smoothing first as shown in Section 5.3.3.

(iii) Futher investigation into multi-image registration and optimal transport could also yield some good registration results.

(iv) The company may wish to look at changing the hardware to a structure similar to that shown in Section 5.5.1.2. Depending on cost factors, this may be something to consider when setting up future tests.

References

1. Y. Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Comm. Pure Appl. Math.*, 44(4):375–417, 1991.
2. A. Calcagni, J. M. Gibson, I. B. Styles, E. Claridge, and F. Orihuela-Espina. Multispectral retinal image analysis: a novel non-invasive tool for retinal imaging. *Eye*, 25(12):1562–1569, 2011.
3. A. Can, C. V. Stewart, B. Roysam, and H. L. Tanenbaum. A feature-based technique for joint, linear estimation of high-order image-to-mosaic transformations: application to mosaicing the curved human retina. In *IEEE Conference on Computer Vision and Pattern Recognition (Hilton Head Island, 2000)*, volume 2, pages 585–591, Los Alamitos, CA, 2000. IEEE Comput. Soc.
4. A. Can, C. V. Stewart, B. Roysam, and H. L. Tanenbaum. A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human retina. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):347–364, 2002.
5. Ellex. White paper: Wavelength selection retinal photocoagulation: An overview of yellow, red and green wavelengths. Technical report, Ellex Medical Pty Ltd, 82 Gilbert Street Adelaide, SA, 5000 AUSTRALIA, 2011.
6. Y. Hirohara, Y. Okawa, T. Mihashi, T. Yamaguchi, N. Nakazawa, Y. Tsuruga, H. Aoki, N. Maeda, I. Uchida, and T. Fujikado. Validity of retinal oxygen saturation analysis: Hyperspectral imaging in visible wavelength with fundus camera and liquid crystal wavelength tunable filter. *Optical Review*, 14(3):151–158, 2007.

7. L. Laaksonen, E. Claridge, P. Fält, M. Hauta-Kasari, H. Uusitalo, and L. Lensu. Comparison of image registration methods for composing spectral retinal images. *Biomed Signal Process Control*, 36:234–245, 2017.
8. G. Lu and B. Fei. Medical hyperspectral imaging: a review. *J. Biomed. Opt.*, 19(1):010901, 2014.
9. N. Mouravliansky, G. K. Matsopoulos, K. Delibasis, and K. S. Nikita. Automatic retinal registration using global optimization techniques. In *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Hong Kong Sar, 1998)*, volume 20, pages 567–570, Piscataway, NJ, 1998. IEEE.
10. A. Myronenko and X. Song. Intensity-based image registration by minimizing residual complexity. *IEEE Trans. Med. Imaging*, 29(11):1882–1891, 2010.
11. Stanford Res. Syst, Sunnyvale, CA. *DSP Lock-in Amplifier, SR830 Model*, 2011. An optional note.
12. C. V. Stewart, C.-L. Tsai, and B. Roysam. The dual-bootstrap iterative closest point algorithm with application to retinal image registration. *IEEE Trans. Med. Imaging*, 22(11):1379–1394, 2003.
13. J.-P. Thirion. Image matching as a diffusion process: an analogy with Maxwell's demons. *Med. Image Anal.*, 2(3):243–260, 1998.

6
Eighth Montreal Industrial Problem Solving Workshop—Rio Tinto Report

Diana Jovmir, Nathalie Ayi, Audrey Poterie, Chris J. Budd, Seong-Hwan Jun,
Nonvikan Karl-Augustt Alahassa, Samira Amraoui, Slim Ibrahim, Tae Yoon Lee, Chu Pheuil Liou,
Catherine Poissant, Viviane Rochon Montplaisir, Loïc-Anthony Sarrazin-McCann, Pierre Duchesne,
Richard Arsenault, and Marco Latraverse

Abstract Rio Tinto uses a complex hydrological model to make ensemble predictions (ESP) for expected freshet volumes that are vital to planning the management of their hydro-electrical power plants. The ESP predictions, however, show an under-dispersion problem that is apparent in Talagrand histograms used for evaluating model performance. This problem has been successfully solved for the winter season (see [1]) but the summer season presents unique challenges. During the workshop, we proposed four different approaches to correcting the under-dispersion problem in the summer: optimization of the initial state in the summer, transfer function models, Gaussian processes, and one-dimensional modelling.

6.1 Introduction

Rio Tinto operates six hydro-electrical dams throughout the Saguenay–Lac-Saint-Jean region, providing 90% of the energy used at their aluminum smelting plants. Since hydro-power generation is a function of the flow rate through the turbines, which in turn depends on water levels in the reservoirs, some planning must go into water level management. Part of day to day operations consists of predicting inflows (which integrate to expected water-levels in the reservoirs) for the next 14 days. This allows the compamy to plan for optimal outflows in advance in order to maximize electricity production as well as minimize inconveniences for neighbouring communities and other users of the reservoirs.

D. Jovmir · N. K.-A. Alahassa · S. Ibrahim · T. Y. Lee · C. P. Liou · P. Duchesne
Université de Montréal

N. Ayi
Université Pierre et Marie Curie

A. Poterie
INSA Rennes

C. J. Budd
University of Bath

S.-H. Jun
University of British Columbia

S. Amraoui
Université de Nice Sophia-Antipolis

L.-A. Sarrazin-McCann · C. Poissant · V. Rochon Montplaisir
Polytechnique Montréal

R. Arsenault · M. Latraverse
Rio Tinto

The quantity of water flowing into a reservoir is a function of the geography of the catchments and the soil properties, as well as weather and precipitation, and thus can't be predicted exactly.

6.1.1 *Hydrological Model*

Rio Tinto uses the CEQUEAU model (a complex time-dependent hydrological model) to estimate resulting inflows; nevertheless it requires daily manual adjustments.

The model takes temperature and precipitation forecasts as inputs (I_t). It then uses initial conditions (x_t) as well as other parameters and information about the geography of the terrain (θ) to derive two additional state variables: $\mathbf{UW_t}$ (the water level in the aquifer) and $\mathbf{SW_t}$ (the amount of water found in the soil). During the winter and spring season, the dominant initial condition is the amount of snow accumulated on the ground ($\mathbf{SN_t}$). This variable is measured at different sites throughout the winter but measurement errors can be substantial. The observed inflows (Q_t) can then be modelled as

$$Q_t = M(x_t, \theta, I_t) + \epsilon_t, \tag{6.1}$$

where $M(\cdot)$ represents the estimated outputs calculated by the hydro model and ϵ is an error term stemming from uncertainty in weather observations, initial conditions, as well as modelling errors.

In reality, the observed inflows rarely match the estimated inflows. Before making predictions for the following two weeks, an experienced technician has to adjust the values of initial conditions until the estimated inflows match observed inflows. This adjustment ensures that initial conditions are as close to reality as possible before making predictions for a new cycle. The method by which these adjustments are made, however, is mostly based on the intuition of the individual technician and thus cannot be replicated.

Once the initial conditions of the hydrological model are adjusted, predictions can be made. Since long-range weather forecasts can be inaccurate, Rio Tinto uses ensemble predictions. Historical weather data, available for the prediction period since 1953, is fed into the model, thus producing 65 different predicted scenarios for the upcoming weeks. Since a fair amount of weather data is available, the company is fairly confident that their predictions will cover most possible outcomes. The next step in the decision process takes the different scenarios as inputs and makes an optimal decision on how to manage water levels to maximize power output. It is important, however, that the input scenarios be equally likely. This assumption can be tested (in retrospect) using a Talagrand histogram [6] (or PIT graph).

6.1.2 *Talagrand Histograms*

To test the assumption of equal probability for each member of the ensemble predictions, the sum of inflows over the 14 days of the prediction period is calculated and compared with the observed water accumulation during this time. This comparison is made for each year for which weather data is available and the quantile in which the observed value lies is recorded. The observation should fall in all quantiles of the predicted scenarios with equal probability. A histogram of the observed quantiles should appear flat (thus following a uniform distribution). The resulting histogram is called a Talagrand Histogram (see [6]) or PIT graph, and it is the tool used by Rio Tinto to assess the performance of their hydrological model.

6.1.3 *Problem*

It has been observed that the ensemble predictions output by the model appear to be consistently under-dispersed. This translates into predictions that often fall above or below the observed inflows and are reflected

in a concave PIT graph (more density on the edges and less in the middle). The problem is further complicated by the fact that the model response to initial conditions is season-specific. During the spring, snow melt is largely responsible for all inflows into the reservoir. During the summer, however, all initial conditions have a considerable effect on inflows and the initial conditions themselves vary much more quickly according to weather conditions.

6.1.4 *Solutions*

Rio Tinto hydrologists solved this problem in an elegant way for the winter/spring melt season [1]. Their method considers measurement errors that occur when setting the snowpack (SN_t) variable as part of initial conditions. With the help of historical information, the distribution of the SN_t variable is approximated. The distribution is then resampled a number of times (e.g., 10) and ensemble predictions are then produced as before for each of 10 sets of initial conditions. This method then yields 610 (61×10) projected outcomes (corresponding to years 1954 to 2014) that are then input into the decision process. The predictions made using this method are found to be more adequately dispersed than in the previous approach (which did not address the variability in snowpack measurements).

This solution, however, has not yet been successfully adapted to summertime predictions, because of the complexity of having to consider all other important initial conditions during the summer.

In the following sections we will describe the four different approaches that we explored during the week in our attempts to correct the under-dispersion of ensemble forecasts for the summer.

- In Section 6.2 we adapt the method for adding variability to winter predictions to the summer season.
- In Section 6.3 we adopt a post-production approach, where we attempt to fix model predictions using time series methods. We estimate the bias of the predictions using a transfer function model (with some or all initial conditions as explanatory variables) and then apply this model to modify the ensemble predictions, in the hope that variability will be more accurately described.
- In Section 6.4 we use Gaussian processes to model the relationship between the inputs and outputs of the CEQEAU model.
- In Section 6.5 we produce a simple one-dimensional hydrological model that can then be used to test various data assimilation approaches.

6.2 Optimization in the Initial State in Summer

6.2.1 *Explanation of the Winter Method*

Rio Tinto uses the CEQUEAU model to produce ensemble streamflow predictions (ESP). This model estimates the freshet volume, which is the hydrological variable of interest, based on parameters θ (calibrated on the data), x (the initial conditions), and I (the climate inputs). It is usually denoted by $M(x, \theta, I)$.

The problem with this model is that the ESP forecasts are often under-dispersed. This feature can mainly be explained by the fact that the hydrological CEQUEAU model is purely deterministic. Thus it does not take into account the fact that, in the real world, some variability may be observed, even when using similar climate and/or state variables. Also the model uses only years on record for the climate variables. Hence the spectrum of possible outcomes is limited.

To correct the under-dispersion, the idea is to find a way to reintroduce the missing variability into the ESP forecast members. This is actually the start of the method developed for the winter season and introduced in the paper [1]. This method will be henceforth called the ΔV method.

First, during the winter and in the sub-basin of the Lac-Saint-Jean watershed in central Quebec, the dominant hydrological variable related to the freshet volume is the snow water equivalent (SWE) on the catchment. This state variable will be used to reintroduce some variability.

We denote by y the observed freshet volume and by t a particular year. Thus $\varepsilon(x_t \mid \theta, I_t)$ represents the error between the observed freshet volume and the freshet volume estimated with the CEQUEAU model (which is denoted by $M(x_t, \theta, I_t)$). Hence we have the following relationship.

$$y_t = M(x_t, \theta, I_t) + \varepsilon(x_t, \theta, I_t) \tag{6.2}$$

In order to estimate the model error based on historical simulations, the former equation was modified by conditioning on the climate input and the parameter θ. This leads to the equation

$$y_t = M(x_t \mid \theta, I_t) + \varepsilon(x_t \mid \theta, I_t). \tag{6.3}$$

The method used to reintroduce the missing variability into the ESP forecast members follows the steps described below.

(1) **Hindcast Step.** For each year t ($t = 1954, \ldots, 2014$),

   (a) We run the hydrological model with fixed parameter set θ using observed climate data I_t for the required ESP duration.
   $\Rightarrow$ We obtain $M(x_t \mid \theta, I_t)$.
   (b) We calculate the error term $\varepsilon(x_t \mid \theta, I_t)$ based on (6.3).
   (c) We have identified previously the scalar value (SWE) in the vector x with which we will play. We apply a correction to SWE in order to reduce the error. We denote by DSWE (Delta snow value equivalent) the corrected value of SWE in x.
   $\Rightarrow$ We obtain DSWE_t.

   The previous process is repeated for each year $t = 1954, \ldots, 2014$: we obtain $\{\mathrm{DSWE}_{1954}, \ldots, \mathrm{DSWE}_{2014}\}$. We can then model the DSWE distribution (in a parametric or non parametric way).

(2) **Prediction Step**

   (a) For each year, we pick a random sample from the DSWE distribution and add it to the SWE value, i.e., we update x_t: we obtain the updated value $\tilde{x}_t$, for $t = 1954, \ldots, 2014$.
   (b) For each year, we perform ESP with historical climatology, i.e., we run the hydrological model.
   $\Rightarrow$ We obtain $\{M(\tilde{x}_t, \theta, I_t)$ for $t = 1954, \ldots, 2014\}$.

   The two previous steps are repeated ten times, leading to 61×10 predictions that we denote by $\{M^j(\tilde{x}_t, \theta, I_t), j = 1, \ldots, 10, t = 1954, \ldots, 2014\}$. In this fashion we obtain a variability-corrected ESP forecast.

To assess the validity of the model, Talagrand histograms are used (see [6]). Such a histogram is obtained by

(1) ordering the 10 predictions of the freshet volume, for each year t;
(2) taking the percentile q_t of the observed freshet volume y_t;
(3) repeating these two last steps for each year;
(4) drawing the histogram of $\{q_{1954}, \ldots, q_{2014}\}$.

The validity of the model is then assessed by analysing the histogram shape. Indeed, a U-shape shows that the model tends to under-estimate the true freshet volume because the true freshet volume more frequently corresponds to the extreme percentiles. On the contrary, a "dome"-shape shows over-dispersion. Consequently, an efficient model results in a flat histogram, i.e., the distribution of the true freshet volumes $\{y_{1954}, \ldots, y_{2014}\}$ is uniform according to $\{M^j(\tilde{x}_t, \theta, I_t), j = 1, \ldots, 10, t = 1954, \ldots, 2014\}$.

The hypothesis that the distribution of the true freshet volumes $\{y_{1954}, \ldots, y_{2014}\}$ is uniform according to $\{M^j(\tilde{x}_t, \theta, I_t), j = 1, \ldots, 10, t = 1954, \ldots, 2014\}$ can also be tested by using the non-parametric test of Kolmogorov-Smirnov at the reference level of 5%. Indeed, if the p-value of the test is inferior to 5%, the hypothesis is rejected, otherwise it is not rejected.

6.2.2 *Extension to the Summer*

Our aim is to apply this so-called ΔV method to the summer period. The first obvious obstacle is that the parameter with which we play during the winter (the SWE), is of course not available in the summer. Thus the first step is to identify the parameter to be used for the summer period.

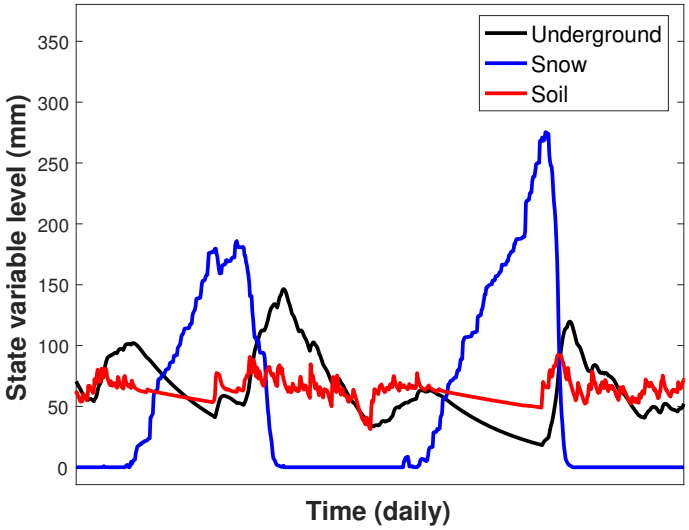


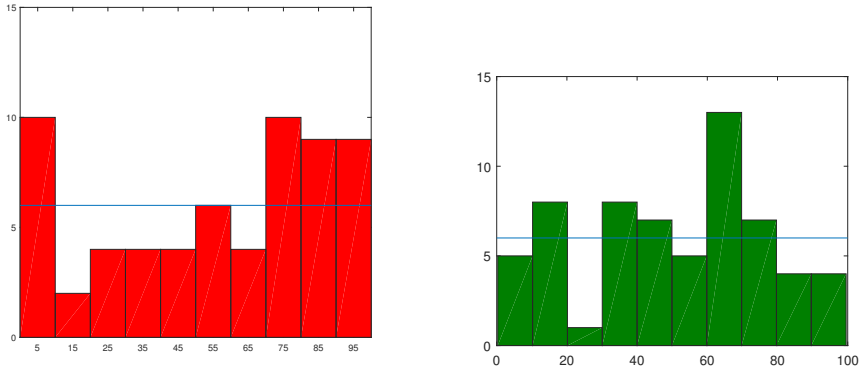
Fig. 6.1: Evolution of the state variable levels over two years.

The different state variables are the snow and the level of water under the ground and on the soil. As can be seen in Figure 6.1, the snow has a behaviour that is pretty smooth. Thus our idea is to pick a variable having a similar behaviour. For this reason, we disregard the soil and decide to focus on the underground. Our first (naive) approach was to apply the previous method, the ΔV method, and replace the SWE by the underground water level (UWL). Unfortunately, this was not very conclusive. If we compare the Talagrand histograms before and after applying the ΔV method, the histogram after correction does not seem to be flatter than the one before correction. Note that the Kolmogorov-Smirnov test does not reject the hypothesis of a uniform distribution. It is well known, however, that the test can have insufficient power with small samples, i.e., its probability of rejecting the null hypothesis is low even when the null hypothesis is false.

This outcome can be explained in the following way: during the winter, there is mainly one phenomenon (snow accumulation), while in the summer, the situation is more complex. Indeed, the evolution of the state variables actually depends on the level of the water already present in the soil and under the ground: if the area is very dry, a rainfall will not have the same consequences as if the area is already wet (in which case the rainfall could lead to flooding).

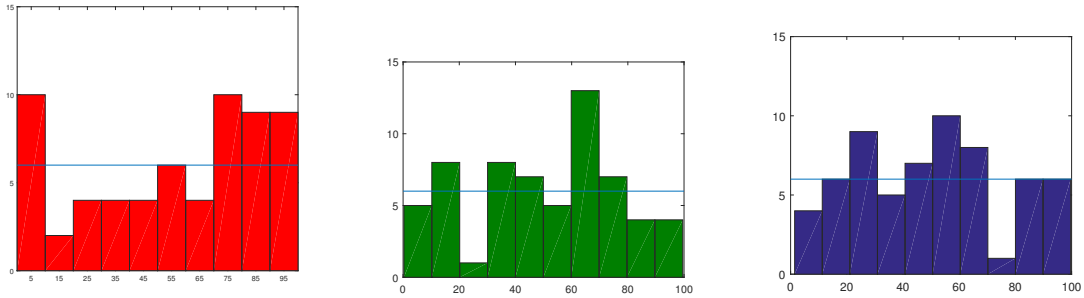
We adapt our method to the summer as follows. In the **Hindcast Step**, the first two steps are the same. Next, when building the DUWL distribution, we actually split the distribution into three distributions according to the initial state (dry, medium, or wet). The thresholds for splitting are chosen by an empirical method based on the empirical distribution of the data. Then the **Prediction Step** is performed for each DUWL distribution.

This approach seems to give better results (as can be seen in Figure 6.3). Indeed, the Talagrand histogram showing the result obtained by using both the ΔV method and the state of the soil is flatter.



Not corrected (p_{value}=0.05) Corrected (p_{value}=0.34)

Fig. 6.2: Application of the ΔV method: comparison of the models before and after applying the ΔV method.



Not corrected (p_{value}=0.05) Corrected (p_{value}=0.34) Conditioned corrected (p_{value}=0.17)

Fig. 6.3: Application of the ΔV method and conditioning: comparison of the models before and after applying the ΔV method and conditionning.

6.2.3 *Discussion*

We conclude that this first approach, rather less naive than the previous one, shows promising results that need to be investigated in more detail.

6.3 Time Series Approach

Treating the hydrological model as a black box, we could choose to propose modifications to the inputs or the outputs of the model. Since the model output has to be adjusted by changing levels of state variables daily in order to match observations, we explored the idea that we might be able to model this adjustment using a time-series transfer function model.

ARMAX models are appropriate for predicting future values in a time-series as a function of past and present values of one or more exogenous variables. Let Q_t^{sim} be the series of model predictions for a given time period and Q_t^{obs} be the observed water levels for the same period.

We decided to focus on modelling the "error" series, D_t, as a function of initial conditions series and their lags, where

$$D_t = Q_t^{\text{obs}} - Q_t^{\text{sim}}. \tag{6.4}$$

We initially chose to use, as an exogenous variable in an ARMAX model, the series whose generic term (P_t) is the sum over the past two weeks of precipitation values. It would also be possible to explore other initial conditions series or to use several variables.

Such a model takes the form

$$D_t = \frac{\nu(B)B^d}{\omega(B)} P_t + n_t, \tag{6.5}$$

where n_t, the error term, is assumed to be a stationary time series of mean 0 that is uncorrelated with P_t but might display an ARMA(p, q) autocorrelation pattern. The notation B^d refers to the backshift operator (instructing us to choose the previous value in the series), i.e.,

$$Bx_t = x_{t-1} \quad \text{and} \quad B^d x_t = x_{t-d}.$$

In this way, $\nu(B)$ and $\omega(B)$ are polynomial operators of finite order, as shown in the following formulas.

$$\nu(B) = \nu_0 + \nu_1 B + \cdots + \nu_q B^q$$
$$\omega(B) = \omega_0 - \omega_1 B - \cdots - \omega_p B^p$$

When these polynomial operators are applied to the explanatory time-series P_t, they describe the lags and their associated coefficients that are relevant in modelling the response series.

The general form of the ARMAX model for the D_t series can then be written in the following (perhaps more intuitive) form:

$$\omega(B)D_t = \nu(B)B^d P_t + \omega(B)n_t \implies D_t = \sum_{j=1}^{p} \omega_j D_{t-j} + \sum_{i=d}^{q+d} \nu_i P_{t-i} + \omega(B)n_t, \tag{6.6}$$

whree p and q are finite lags.

We can interpret this expression as saying that the difference between predicted water levels and observed water levels depends on past values of precipitation (or other initial conditions series) and past values of itself in a predictable way. This idea is philosophically aligned with the current practice of making adjustments to the initial conditions series each day in order to match the predicted and observed water levels. If we can estimate how the difference series (D_t) reacts to different values of initial conditions, we might be able to predict the discrepancy directly (instead of adjusting initial conditions to eliminate it) and simply add this predicted difference to the model prediction to obtain a less biased and more adequately dispersed set of ensemble predictions.

For our model we chose P_t as the exogenous variable and included lags of up to 7 days. The remaining noise model, n_t, was defined as having an ARMA$(2, 2)$ shape. In the future, other available variables could be included in the model; variable selection would have to be performed in order to be able to include larger lags, going all the way back to the previous years values (possibly). Other available state variables and initial conditions series are shown in Figure 6.4.

6.3.1 *Simulation*

For this project, we had access to data going back to 1953. For the estimation part of the model, we made use of weather and initial conditions time series (T_t, P_t, SU_t, and SW_t), the output of the hydrological model before corrections (Q_t^{sim}), and observations of actual water levels for each year (Q_t^{obs}). Our prediction date was July 15th and we used data points occurring before this date to estimate model parameters for

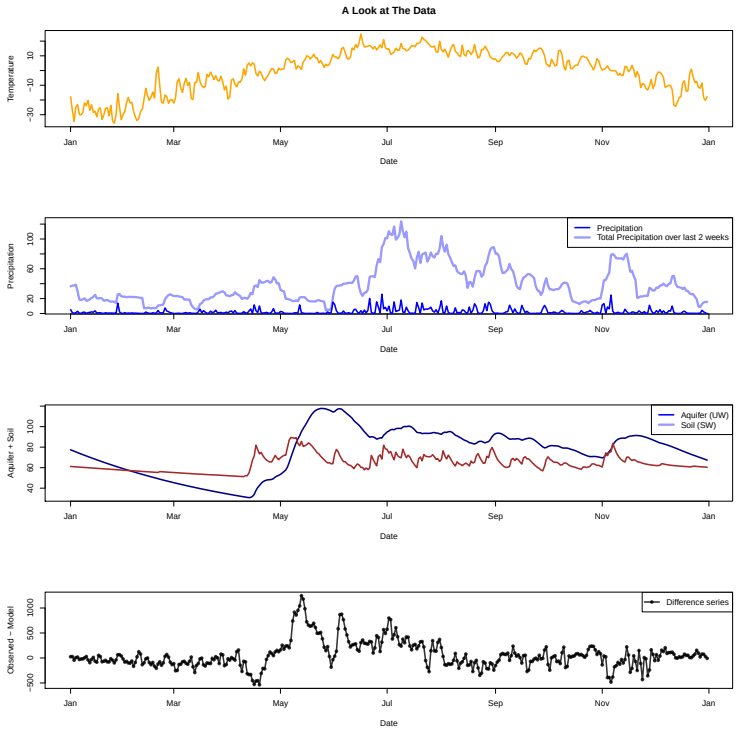


Fig. 6.4: Time-series of initial conditions and state variables (from year 1994) used by the CEQEAU hydro model to calculate inflow volumes in the three upper panels. These variables could be used as exogeneous variables in an ARMAX model to estimate the difference series (D_t), shown in the lower panel. We ultimately chose to base our model on the sum of precipitation values over the past two weeks (in panel 2), that we refer to as P_t.

each year. We used the **armax** function of the **TSA R** [4] package to fit a transfer function model to the difference (D_t) series for each year, using the precipitation series P_t as the exogenous variable as in Equation 6.5. Parameters in the model were allowed to vary from year to year, though in future work some effort should be put into finding out whether it is possible to come up with a general solution that can remain constant from year to year.

After the estimation, the model was transformed into a "regression" style model as in (6.6). The reason for this is that the **armax** function lacks a **predict** method but the **arima** function from the **stats** [7] package in **R** will accept a regression model with correlated errors and produce predictions. In this way, 63 different predictions for the D_t series were made for the 15 days following July 15th, each one using the precipitation series for each year since 1953.

Recall that

$$D_t = Q_t^{\text{obs}} - Q_t^{\text{sim}}.$$

We then assume that the difference series is related to our predicted values as in

$$D_t = D_t^{\text{pred}} + \epsilon_t,$$

where ϵ_t is assumed to follow a $\text{N}(0, V_\epsilon)$ distribution. We then retrieve a prediction for the expected water levels by adding the predicted D_t^{pred} series for a particular year to the model prediction output for the respective year:

$$Q_t^{\text{obs}} = Q_t^{\text{sim}} + D_t^{\text{pred}} + \epsilon_t \implies Q_t^{\text{pred}} = Q_T^{\text{sim}} + D_t^{\text{pred}}.$$

In a subsequent step, we add 10 random white noise series to each prediction line to simulate the random noise, ϵ_t, which is sampled from a normal distribution of mean 0 and variance V_ϵ (this variance was estimated during the estimation step). In this way, we obtain 650 different scenarios, each of which having the same probability of occurring. Figure 6.5 shows the data used and the possibilities obtained.

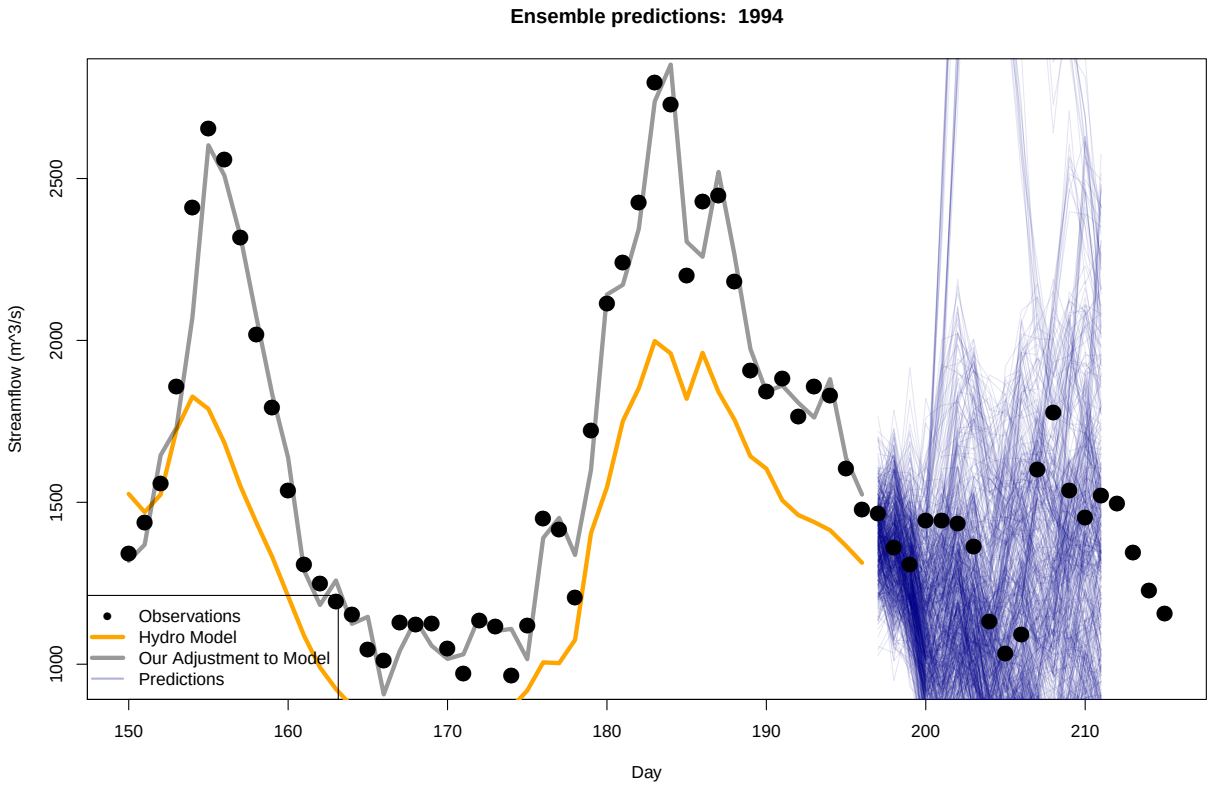


Fig. 6.5: Example of ensemble predictions for the year 1994 as output by the hydrological model with our **armax** modifications. The blue lines each represent a predicted scenario based on weather data for one of the past 65 years with 10 different error series added on for a total of 650 possibilities.

The total volume of inflows for the prediction period is then calculated for each scenario and compared to the observed inflow during this interval in order to produce the diagnostic PIT diagrams. Our results are illustrated in Figure 6.6.

6.3.2 *Results and Discussion*

The simple ARMAX model that we implemented during the workshop seems promising, as it improved the dispersion of the ensemble predictions (as seen in Figure 6.6). There is much room for improvement, however.

First of all, most of the data present in the initial conditions time series would occur in the winter and spring (spring run-off is the dominant variable all the way up into May). If we wanted to characterize summer season data only, we would be left with a few data points for each year: hence we decided not to take that route.

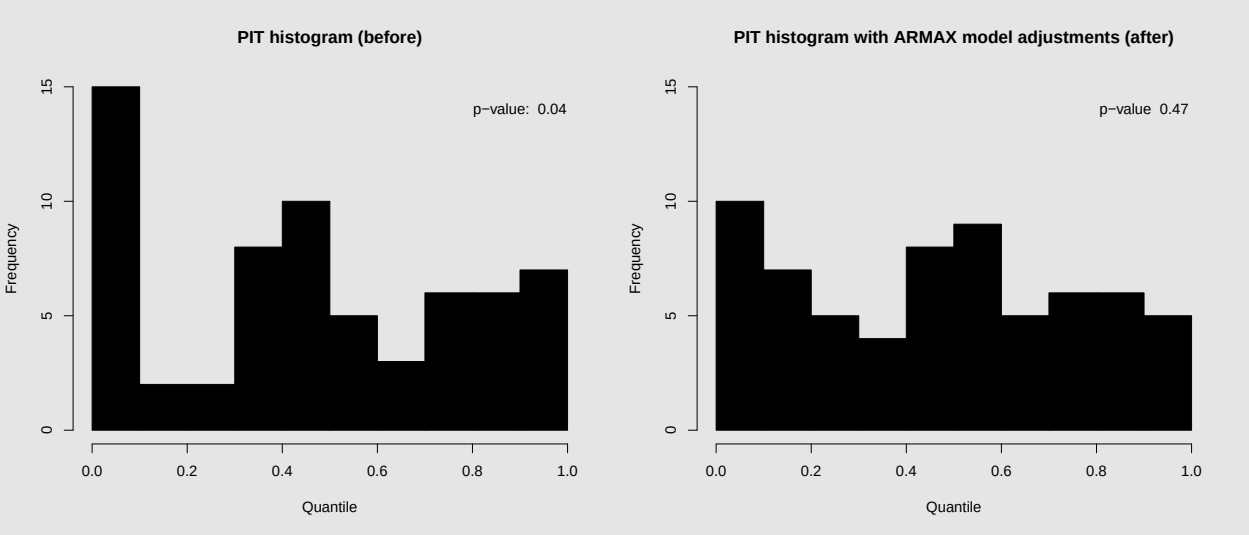


Fig. 6.6: PIT histograms for predictions before time series adjustment (left) and after (right). A Kolgomorov–Smirnoff test yields p = 0.04 for the original predictions and p = 0.47 after the time-series adjustment was performed.

The second issue of concern is related to model selection. We allowed the correlated noise series n_t to follow an ARMA(p, q) model with p and q at most two, while the exogenous variable P_t could have an AR dependence of order 2 and MA terms up to order 7. Selection of variables was made in a heuristic manner, mainly by deciding what seems to work best among the options that are convenient to implement. In future work, however, a lot more focus should be put on variable selection. Since a lot of past data is available, we can imagine that lags of one year or more could play a significant role in model performance. Furthermore, since several initial conditions series are available, more than one exogenous variable could be used for fitting and predicting the difference series. It is computationally expensive to try to perform variable selection for such large sets of potential variables; therefore efficient algorithms for variable selection (such as the adaptive LASSO) would have to be investigated.

Another possibility to explore is related to the splitting of data. During the workshop, we decided to separate the data by year and fit data for each year separately. Two other possibilities deserve to be considered. The time series could be left "un-chopped" and we could attempt the fitting of a single model using all data points since 1953, allowing for long-range correlations, with lags of up to several years. Another possibility is to separate the data by year but fit all years together in a multivariate time series model, with the purpose of obtaining parameters that are applicable to all years. This approach is not guaranteed to succeed (because it would require the existence of some underlying "physical" model connecting the difference series to the initial conditions in a reliable way) but it deserves further investigation.

6.4 Gaussian Process Approach

CEQUEAU is a complex model and its inner workings are unclear. It is a deterministic code that takes as input the state variable and the weather information, denoted as I, and outputs the predicted flow $\hat{y}$. The Gaussian Process (GP) is a popular tool for modelling the relationship between the input and the output of a complex computer simulation code [2]. This section demonstrates potential avenues where GP can be utilized for a subset of problems posed during the workshop. We begin with a brief background on GP (for a

complete treatment of this topic, we refer the reader to [8]). We then present preliminary results. The report concludes with a discussion.

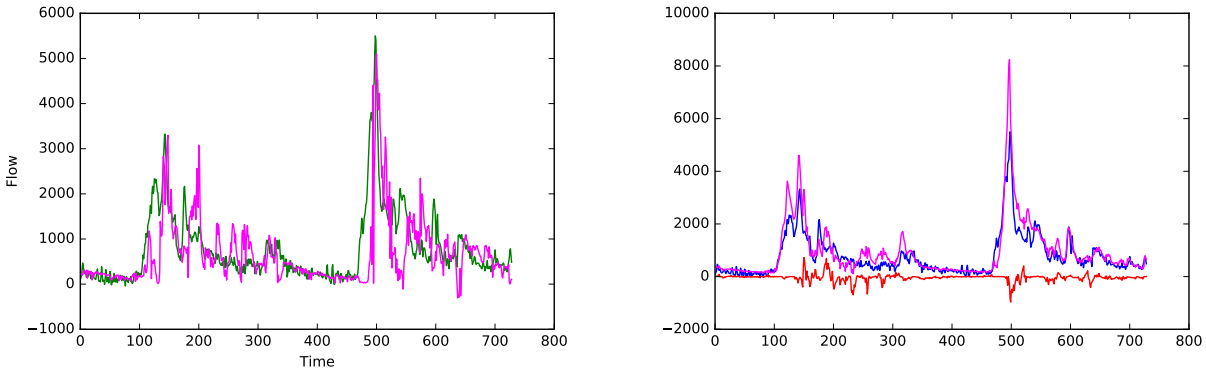
6.4.1 *Gaussian Process*

The first step is to view a complex computer code such as CEQUEAU as a black box function, $\hat{y}(x) = M(x)$, where x denotes the input to the computer code and $\hat{y}$ denotes the output. The dependence between any pair of inputs, x, x', is modelled using a covariance function, $K_\theta(x, x')$. The parameter θ captures the dependence between a pair of inputs x, x' and is estimated from the sample evaluations of the function: $\big(x_n, M(x_n)\big)$, for $n = 1, \ldots, N$, where N denotes the number of evaluations. For the CEQUEAU model, each point $x_n = (I_n, t)$ is composed of the state variables and the weather forecast, denoted by I_n and t (respectively), where t is the time (for example, day of the year). There are various choices for the covariance function; a popular choice is the radial basis function (RBF), also known as squared exponential kernel, which is suitable if the function $\hat{y}(x)$ is assumed to be a smooth function. The squared exponential kernel is given by

$$K(x, x') = \exp\left(-\frac{1}{2\theta} d(x, x')\right),$$

where $d(x, x')$ denotes the distance between the two inputs x, x'. Typically $d(x, x')$ is defined as $\|x - x'\|^2$. Since there is little reason to assume that the output of the CEQUEAU model varies widely when there is a small change in the input, the squared exponential kernel was used for obtaining the preliminary results.

6.4.2 *Results*



(a) (b)

Fig. 6.7: Prediction results on the unseen data. (a) Green indicates the calibrated values produced by CEQUEAU and magenta indicates the predicted values produced by GP. (b) Blue indicates the observed flows, magenta denotes the output of the CEQUEAU model plus the predicted residual values produced by GP, and red indicates the residuals produced by GP.

The points at which to evaluate the black box function are typically determined via carefully determined statistical design [9]. Some technical issues, however, prevented direct evaluation of the CEQUEAU model

during the workshop. Nonetheless we were provided access to calibrated output from the CEQUEAU model and the preliminary results were obtained by fitting GP to these data. Additional exploration consisted of fitting GP on the residuals obtained between the calibrated output of the CEQUEAU model and the observed flow.

GP was fitted to the calibrated CEQUEAU model using the data of the 10-year period from 1953-Jan-01 to 1962-Dec-31. The estimated parameter value is $\theta = 0.405$. This fitted model is used to make predictions for the 2-year period starting on 1963-Jan-01 and the result is displayed in Figure 6.7 (a). The prediction seems to model the output of CEQUEAU quite well in its ability to pick out the peaks and troughs. To improve upon this, we have also fitted the GP on the residual, using the formula $r(x_t) = y_t - \hat{y}(x_t)$, where y denotes the observed flow at time t. The prediction on the unseen data for a two-year period starting on 1963-Jan-01 is displayed in Figure 6.7 (b). This figure seems to indicate that modelling the residual leads to accurate modelling of the observed flows. It illustrates that the best use of GP may be its use in combination with CEQUEAU: that is, its best use may be to model the difference between the observed and the calibrated values output by CEQUEAU (rather than modelling the output of CEQUEAU directly). This conclusion, however, may be premature, considering that we did not have access to the CEQUEAU code.

6.4.3 *Discussion*

We will conclude this section with a discussion on solving one of the main difficulties underlying the use of the CEQUEAU model: determining a suitable initial condition, z_0, where z denotes the underlying state of the soil (i.e., $I = (z, w)$, where w denotes the weather data). The current procedure encapsulates a manual "trial-and-error" approach, where a technician tries multiple values of z_0 until the simulated flow appears close enough to the observed flow: for example, until the expression $\sum_{t=1}^{T} \|y_t - \hat{y}_t\|$ is *small* (where T denotes the total number of time points under consideration). The precise definition of "small" is unclear and the procedure relies heavily on the experience of the technician and his prior knowledge of how the CEQUEAU model works. The GP may provide a way of inferring the initial condition (z_0).

The GP model can be useful as a simulation tool. Starting from an arbitrary initial value, we can use the GP model to simulate the forecast for the next 14 days. It may be possible to simulate the future values using the GP for optimization of the initial state: that is, we can pose the problem of finding the suitable initial value as an optimization problem:

$$\hat{z}_0 = \arg\min_{z_0} \sum_{t=1}^{T} \big(y_t - \hat{y}(z_t \mid z_{t-1}, w_{t-1})\big)^2, \tag{6.7}$$

where $\hat{y}(z_t \mid z_{t-1}, w_{t-1})$ is simulated by the GP model. The solution to the above optimization problem may be obtained via Bayesian optimization techniques (see [3]). This optimization can be performed over the collection of past data. Also note that posing the problem of finding the initial condition as an optimization problem focuses the attention on optimality, in terms of the choice of the initial values (i.e., of minimizing the squared difference between the observed and the predicted values).

Note also that the state variables z_t are hidden variables, which evolve according to a hidden Markov model structure:

$$z_t = f(z_{t-1}, w_{t-1}) + \eta_t, \quad \text{for } t > 0, \tag{6.8}$$

where f is a function of the current state z_{t-1} and the weather condition w_{t-1} and η_t denotes the random fluctuation. The observed values depend upon the state variables:

$$y(z_t) = g(z_t) + \epsilon_t, \tag{6.9}$$

where ϵ_t denotes the random noise. Sequential Monte Carlo methods [5] are widely adopted for problems exhibiting a hidden Markov model structure. Formulating the problem of finding the initial condition in the context of SMC methods may be the next step towards formalizing the problem of inferring the initial value.

6.5 One-dimensional Model

The idea behind the development of a one-dimensional model is to produce a simple, but realistic, deterministic hydrological model that will allow us to develop an understanding of the water flux and can then be used to test various data assimilation approaches.

6.5.1 *The Model Fomulation*

In the one-dimensional model we consider the whole hydrological basin to be a series of connected *cells* C_i. In this model C_1 will be assumed to be the cell that is highest in altitude. It is also assumed that there is a positive free water flux F_i^n from cell C_i to cell C_{i+1}, with no flux into cell C_1. At time t_n the cell C_i will contain a quantity SL_i^n of ground water (i.e., water in the soil) and a quantity SW_i^n of snow. In addition, each cell will be subjected to a temperature T_i^n and a precipitation R_i^n, which can be obtained from measured data. This is illustrated in Figure 6.8.

We now make the following physical assumptions.

1. The free water flux is a function of the water in the soil, expressed by the following equation.

$$F_i^n = f(SL_i^n) \tag{6.10}$$

2. If $\boldsymbol{T_i^n > 0}$ holds, then the snow melts as the temperature rises and the melt water is added to the water in the soil SL_i^n. In addition the precipitation falls as rain and is also added to the soil water.
3. If $\boldsymbol{T_i^n \leq 0}$ holds, then the precipitation falls as snow and is added to the snow SW_i^n in the cell C_i.

We make the following modelling assumptions.

- The amount of snow that melts in one time unit is directly proportional to the temperature above freezing. All of the melted snow becomes "water in the soil."
- If the temperature is *positive*, then the amount of water in the soil added within a unit of time is directly proportional to the precipitation.
- If the temperature is *negative*, then the increase (within one unit of time) in the snow contained in the cell is directly proportional to the precipitation.

We may now formulate a mathematical model. If $T_i^n > 0$ holds, the equations are as follows.

$$SW_i^{n+1} = SW_i^n - \alpha T_i^n SW_i^n \tag{6.11}$$

$$SL_i^{n+1} = SL_i^n + \alpha T_i^n SW_i^n + \beta R_i^n + F_{i-1}^n - F_i^n \tag{6.12}$$

If $T_i^n \leq 0$ holds, the equations are as follows.

$$SW_i^{n+1} = SW_i^n + \gamma R_i^n \tag{6.13}$$

$$SL_i^{n+1} = SL_i^n + F_{i-1}^n - F_i^n \tag{6.14}$$

Let N denote the number of cells. As described above, we assume that $F_0^n = 0$ holds. The value

$$Q^n = F_N^n$$

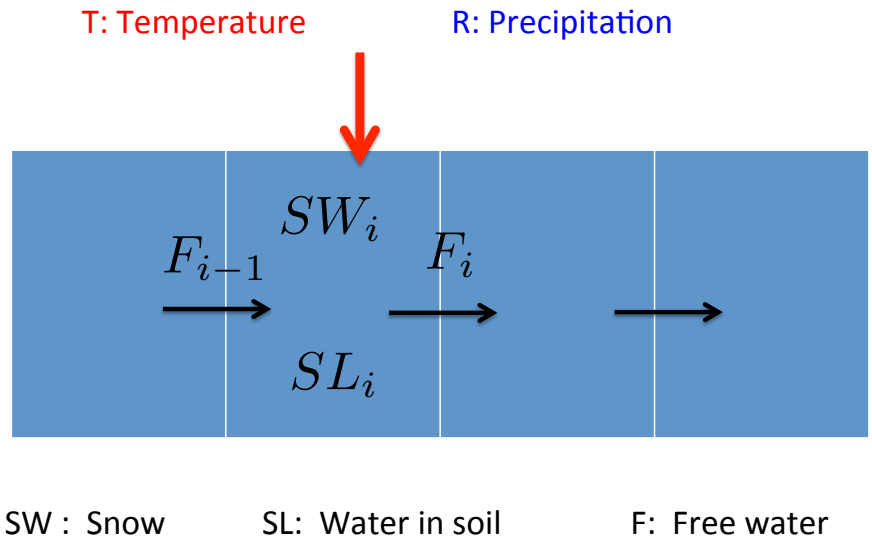


Fig. 6.8: A schematic of the one-dimensional cell model.

is then the *measured flux* that will be our primary measurement for the data assimilation calculation.

6.5.2 *Implementation*

To implement this model, we assume that the time unit is *one day* and N is equal to 10. The values of the constants of proportionality above, and the function in (6.10), were chosen to get a reasonable fit to the measured data. Here are the choices we made.

$$\alpha = 1/15, \quad \beta = \gamma = 1, \quad f(S) = S/4. \tag{6.15}$$

The temperature and precipitation data were taken from those supplied. Initial values of SW and SL were estimated and the model was "spun up" by running it over several years before results were plotted. The resulting model was very easy to implement in Matlab. As output we took SW^n to be the *total* amount of snow in all of the cells combined on the nth day, and we also calculated Q^n. The results of these calculations in one year, together with the mean temperature T^n and mean precipitation R^n, are presented in Figure 6.9. Similarly, calculations over ten years are presented in Figure 6.10.

The results of a one-year simulation (displayed in Figure 6.9) show that the total Q has a sharp peak in the spring when the snow melts, and that this is the dominant effect on its value throughout the year. The

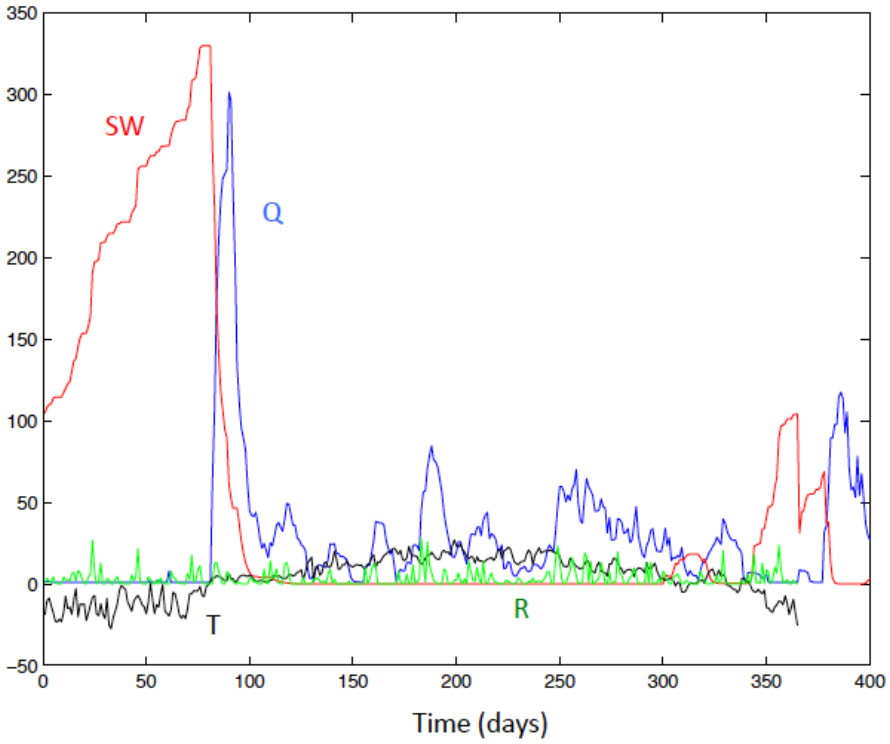


Fig. 6.9: Results for one year of the one-dimensional cell model: outputs Q (blue) and SW (red) and inputs T (black) and R (green).

(rather unpredictable) rainfall R has a lesser, though still significant, effect. A similar result can be seen in the ten-year simulation, and we see a variation in the total amount of snow SW from one year to the next.

6.5.3 *Data Assimilation*

Noting from the above calculation that the total amount of snow fall SW is the main factor determining Q, we ask the question: How well is it possible to estimate SW from the data? To answer this question, we conducted the following test for assessing accurately an initial value of SW from the noisy measured data of the total flux Q.

- Take an initial *estimated value* of SW^0 at time t^0 (assumed to be in winter) and distribute the snow evenly over all of the cells. Set the initial value of SL to 0 (this is a reasonable assumption for winter months).
- Run the model to generate two years of values for the total flux Q^n.
- Add noise to Q to yield the noisy measured data Q_{Noise}.
- Make a shift ($SW^0 \rightarrow SW^0 + \delta SW$) and generate a new time history of the total flux (Q_δ).
- Find the δSW minimizing the *error* between Q_{Noise} and Q_δ.

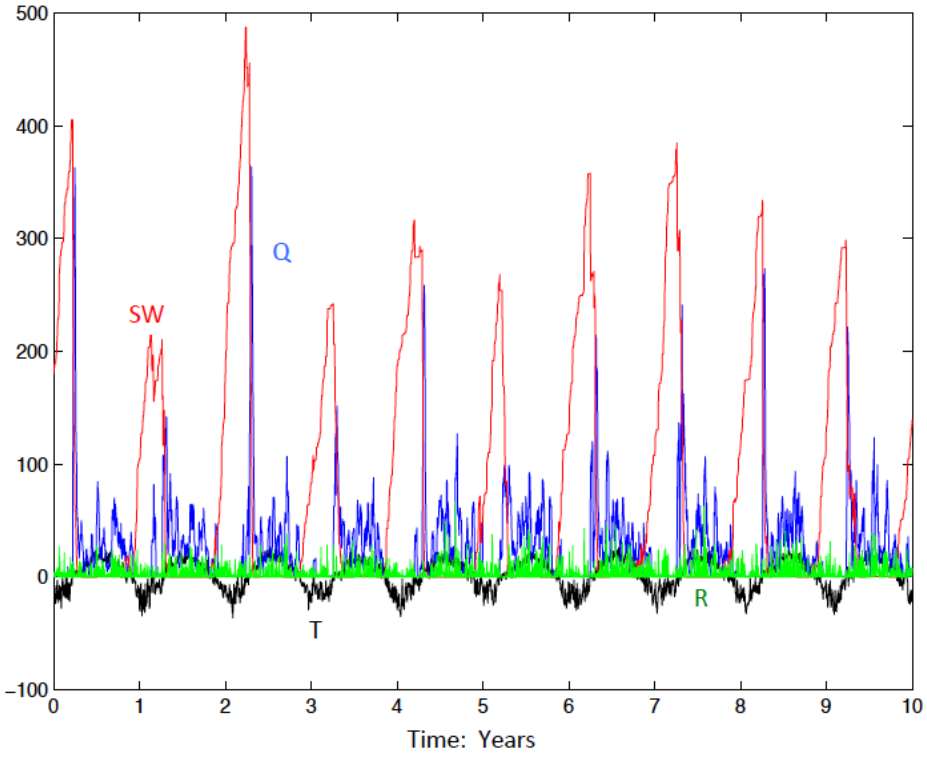


Fig. 6.10: Results for ten years of the one-dimensional cell model.

An example of the difference between Q and Q_{noise} is given in Figure 6.11.

We considered two measures of the error:

$$E_1 = \left(\int (Q_{\text{Noise}} - Q_\delta) \, \mathrm{d}t \right)^2 \tag{6.16}$$

and

$$E_2 = \int (Q_{\text{Noise}} - Q_\delta)^2 \, \mathrm{d}t. \tag{6.17}$$

In the case of E_1 we choose δSW so that $E_1 = 0$ holds. In the case of E_2 we choose δSW to minimize its value.

We ask the question: Which of these two error measures leads to the best estimate of SW?

We can test this procedure by considering a number of realizations of Q_{Noise} with random noisy data. If the data assimilation procedure is working well, then over all of these realizations we should see a mean of $\delta SW = 0$. A measure of the performance of this algorithm is given by the standard deviation of the resulting estimate (denoted by σ). Accordingly we take 100 realizations, with a Gaussian noise added to Q.

The resulting histogram of the values of δSW is shown in Figure 6.12 together with estimates of the mean μ and standard deviation σ. Two plots are displayed in Figure 6.12: (a) the estimate for error measure E_1

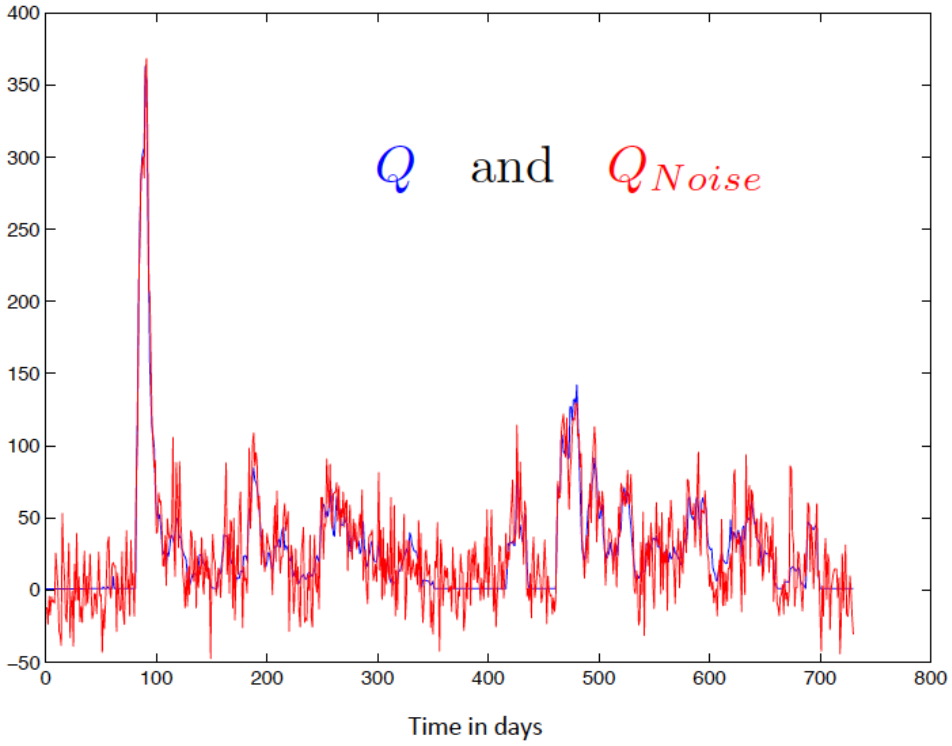
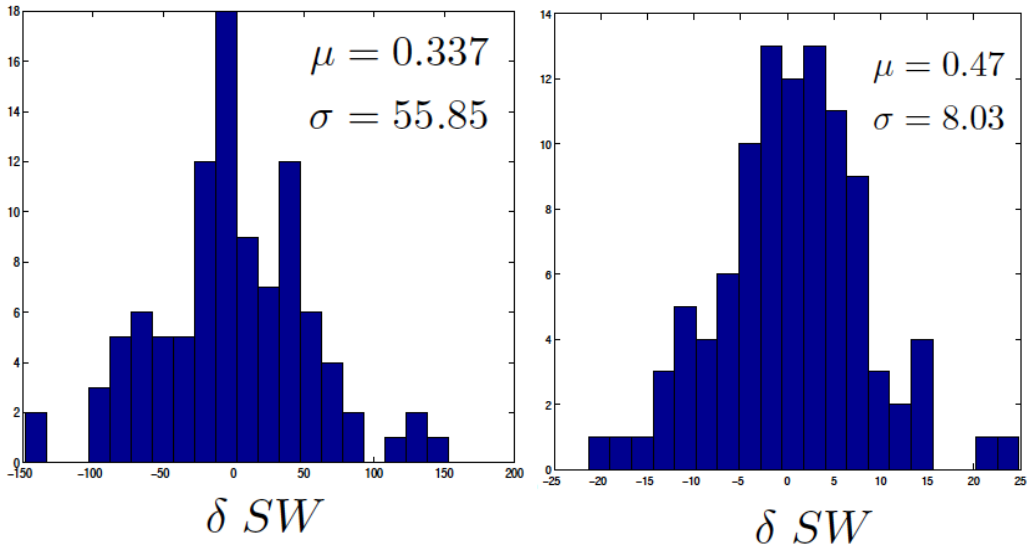


Fig. 6.11: The output Q of two years of the one-dimensional cell model, together with a noisy perturbation Q_{Noise}.

and (b) the estimate for error measure E_2. We can see that whilst the mean of the estimate is close to zero in both cases, the standard deviation of the error for measure E_1 is *much* higher than that for measure E_2. We conclude that in the data assimilation routine the measure E_2 yields a much better estimate for the initial snow value SW^0.

6.5.4 *Discussion*

Whilst very simple, the one-dimensional model gives surprisingly realistic-looking results for the variation in the amount of snow and the total flux. It is also useful in testing the two error measures used in the data assimilation calculation to estimate the initial snow value from the measured flux Q. Further tests of the simple model could help see how well the future values of the flux (the focus of our interest) can be predicted. We recommend this model be used for estimating the effectiveness of further data assimilation procedures. Its simplicity allows it to be used for a large number of realizations and to test many schemes, in a manner that may not be possible with the more complex CEQUEAU model. It would also be interesting to carry out a comparison between the predictions of the one-dimensional model and these full simulations.



(a) E1 (b) E2

Fig. 6.12: The histogram of the estimates of δSW given by the two error measures E_1 and E_2. The mean and standard deviation are also given. We see that the error measure E_2 gives a much better estimate of SW than E_1

.

6.6 Acknowledgements

We would like to thank Richard Arsenault and Marco Latraverse from Rio Tinto for bringing this interesting problem to our attention, the Université de Montréal for the delicious lunches, Dr. Odile Marcotte for organizing the workshop, and Prof. Pierre Duchesne for coordinating our team. We would also like to thank Diana Jovmir for taking the main responsibility for the writing of this report.

References

1. Richard Arsenault, Marco Latraverse, and Thierry Duchesne. An efficient method to correct under-dispersion in ensemble streamflow prediction of inflow volumes for reservoir optimization. *Water Resources Management*, 30(12):4363–4380, 2016.
2. Derek Bingham, Pritam Ranjan, and William J Welch. Design of computer experiments for optimization, estimation of function contours, and related objectives. In J. F. Lawless, editor, *Statistics in Action: A Canadian Outlook*, chapter 7. CRC Press, Boca Raton, FL, 2014.
3. Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arXiv:1012.2599.
4. Kung-Sik Chan and Brian Ripley. *TSA: Time Series Analysis*, 2012. R package version 1.01.

5. Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In Dan Crisan and Boris Rozovskiĭ, editors, *The Oxford Handbook of Nonlinear Filtering*, chapter 24, pages 656–704. Oxford Univ. Press, Oxford, 2011.
6. Thomas M. Hamill. Interpretation of rank histograms for verifying ensemble forecasts. *Monthly Weather Review*, 129(3):550–560, 2001.
7. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017.
8. Carl Edward Rasmussen and Christopher KI Williams. *Gaussian Processes for Machine Learning*. Adapt. Comput. Mach. Learn. MIT Press, Cambridge, MA, 2006.
9. Jerome Sacks, William J. Welch, Toby J. Mitchell, and Henry P. Wynn. Design and analysis of computer experiments. *Statist. Sci.*, 4(4):409–435, 1989.

7

Arbitrage Strategy Between Next-Day Delivery Prices and Real-Time Delivery Prices of Electricity Megawatts on the Physical California Market

submitted by CWP Energy, Montreal, Inc.

Fabian Bastin, Marina Chugunova, Betul Zehra Karagul, Manuel Morales, Nazim Regnard, Yiran Wang, and Farshid Zoghalchi

7.1 Introduction

CWP Energy is a private company involved in the physical and financial electricity markets. In particular it carries out daily imports and exports of electricity between diverse physical markets in North America (e.g. NYISO, NEISO, MISO, ERCOT, SPP, OntarioISO), thus contributing to a secure equilibrium between the supply of, and demand for, electricity (which must be ensured in real time) and to the reliability of the North American electricity network.

Electricity is indeed an asset that cannot be stored, and the demand for electricity at a given time and in a given location (called "demand node" of the network) will be satisfied only if (i) the corresponding quantity is produced (at a node called "production node") and (ii) transmitted from the production node to the demand node (iii) simultaneously. ISO (Independent System Operators) are non-profit public entities in charge of a geographic area including one or several North American states (for instance NYISO for New York State, NEISO for New England). Every day the ISO ensure that they have an accurate forecast of the next-day electricity demand and generation capacity.

In deregulated markets such as the North American markets mentioned above, the electricity delivery price is determined through a matching between the supply and demand curves, provided every five minutes by the various market players in the network (electricity distributors, generators, importers and exporters of electricity). Every five minutes each player provides the relevant ISO with a curve expressing the relationship between the price and amount of electricity. The ISO has only to aggregate these curves and select the buying or selling price as the point of intersection of the aggregated supply curve and the aggregated demand curve.

The real-time (RT) price for a given time (hour) corresponds to the price selected from the curves provided one hour beforehand. In practice many computations of this kind are carried out at the network nodes, resulting in a price for each node. If there is no congestion in the network, however, all nodes will carry the same prices. This is not true if there is congestion, since a congested line cannot transmit enough electricity to satisfy all the connected demand nodes.

The extreme volatility of the RT price complicates the management of production units and distributors. In order to secure their purchases and sales, the ISO have designed a day-ahead (DA) delivery mechanism,

Fabian Bastin · Manuel Morales · Nazim Regnard · Yiran Wang
Université de Montréal

Marina Chugunova
Claremont Graduate University

Betul Zehra Karagul
Hacettepe University

Farshid Zoghalchi
University of Toronto

as follows: (i) at 10:30 AM local time, the participating market players send to the ISO 24 supply or demand curves for the nodes of relevance to them, (ii) the ISO selects a price for each of these nodes by aggregating these supply (or demand) curves (iii) while taking into account his own internal demand forecast, the various production capacities, and congestion. The selected price is called the DA price. The market player who buys (respectively sells) at the DA price for a certain hour is committed to buying (respectively producing) the agreed number of megawatts the next day.

The gap between the actual demand and the DA demand, as well as the gap between the actual production and the DA production, are adjusted through a buyout or resale mechanism at the RT price. In practice, since the market is deregulated and the number of network nodes is huge, the distributors or generators can use the day-ahead bidding mechanism to manipulate the market and introduce a bias in the RT-DA spread, defined as the DA price minus the RT price.

To reduce or even eliminate this market power, the North American ISO have completed the DA delivery mechanism by introducing a virtual bidding mechanism that allows financial firms without any physical asset to speculate on the RT-DA spread at any node [2]. These virtual bidding contracts allow any physical network node to buy (respectively sell) a certain number of megawatts at the DA price and to resell (respectively buy out) automatically the same number at the RT price. No physical flow takes place when such a contract is carried out. A virtual supply curve and a virtual demand curve are simply added to the real supply and demand curves (respectively) at the time when the day-ahead bidding takes place [1].

This project aims to design an automated algorithm for trading virtual products at three important nodes of the physical market called CAISO (California Independent System Operator): SP15, NP15, and SP26, as represented on Figure 7.1. The California electricity system is characterized by the prevalence of renewable energy production, such as solar and wind power, resulting in large production volatility but also potentially significant gains from the signed contracts.

At 10:30 AM on a given day and for a given node, the algorithm must decide which hours on the next day warrant a short, long, or neutral position (i.e., no position at all). This algorithm should result in profits on an annual, trimestrial, and monthly basis, while satisfying criteria on the maximum daily loss. The data provided by CWP Energy contains loads (actual value, day-ahead value, and two-days-ahead value), wind and solar production (actual value and day-ahead forecast value), and price (DA price, RT price, and RT-DA spread) of CAISO, from January 1, 2014, to August 2017, 7, for each hour.

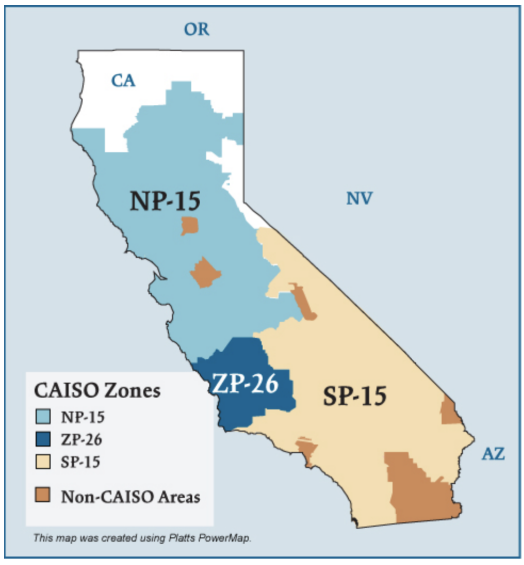


Fig. 7.1: CAISO map.

The report is organized as followed. We first introduce basic concepts in energy trading and then we examine daily patterns in different seasons. Next we select and calculate certain variables to build the model and try to reduce random noise by clustering similar observations. Finally we propose a preliminary analysis of the bidding optimization problem.

7.2 Energy trading

The most fundamental aspect in energy trading is the choice of a position. At day j, the bidding decision on hour h at day $j+1$ is either

A Short position sell electricity at price $d^*_{j+1,h}$ or
A Long position buy electricity at price $d^*_{j+1,h}$,

where $d^*_{j+1,h}$ is the price per MW. Based on the biddings, a day-ahead (DA) price (denoted by $d_{j+1,h}$) is determined at day j (but after the decision concerning $d^*_{j+1,h}$), and the contract is selected if

$$d^*_{j+1,h} < d_{j+1,h} \quad \text{(short)}$$
$$d^*_{j+1,h} > d_{j+1,h} \quad \text{(long)}.$$

Since the electricity cannot be stored, an actor who has been awarded a contract will have to buy (sell) electricity immediately at the RT price $r_{j,h}$ when he (she) has taken a short (long) position. Assuming that we trade $q_{j+1,h}$ megawatts, the payoff will therefore be

$$q_{j+1,h}(d^*_{j+1,h} - r_{j+1,h})\mathbb{1}_{d^*_{j+1,h}<d_{j+1,h}}$$

when we are in a short position, while for a long position, the payoff will be

$$q_{j+1,h}(r_{j+1,h} - d^*_{j+1,h})\mathbb{1}_{d^*_{j+1,h}>d_{j+1,h}}.$$

The spread is defined as

$$\tilde{s}_{j+1,h} \overset{def}{=} d_{j+1,h} - r_{j+1,h}.$$

The difficulty resides in the fact that $d_{j+1,h}$ and $r_{j+1,h}$ are unknown at the bidding time, i.e., when fixing $d^*_{j+1,h}$. In the California market, we have that $E[\tilde{s}_{j+1,h}] \approx \$3$. The spread distribution, however, has a heavy left tail; the spread realization can be very negative, close to -$1000 in the California market. Such a situation is an advantage when choosing a long position, but a problem when choosing a short position, and it can be shown that because of the contract selection process, a naive strategy consisting of always taking a short position results in losses in the long term. There remains the possibility of taking a neutral position, i.e., of not making any bid at all, but in this case there will be a gain of zero. Such a strategy can however be selected as a risk-averse option, when the chances of important losses are high.

7.3 Seasonal patterns

Both the loads and price of electricity can vary a lot from one season to the next because of the heat and cooling effects and the mutable weather conditions (which affect the wind and solar energy production). As shown in Figures 7.2 and 7.3, the average production curves change over the year, Summer and Spring being windier and the solar production benefiting from more daylight. Thus we cannot use similar strategies in all seasons to cope with this problem. Using historical data, we try to evaluate the accuracy of the weather

forecast and to recover the market players' reaction to the known information when establishing their own DA supply or demand curves.

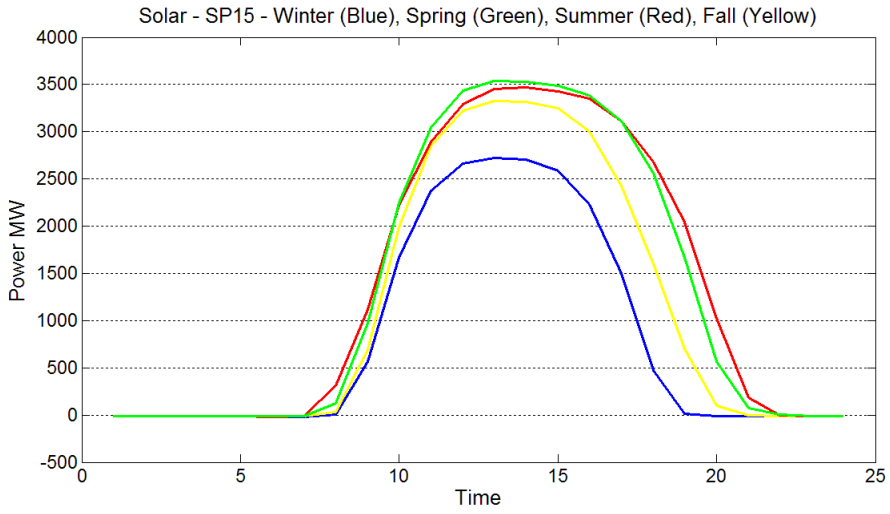


Fig. 7.2: Average solar production at node SP15.

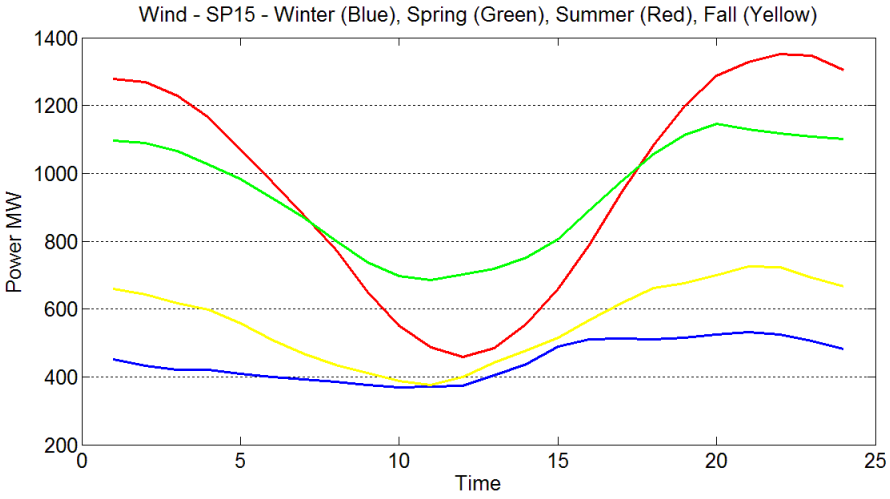


Fig. 7.3: Average wind production at node SP15.

As we can observe in Figure 7.4, the DA-RT price spread exhibits different seasonal patterns. We can also observe intra-day variations. In particular, the price forecast seems less accurate in the evening, between 17:00 to 24:00, corresponding to the after-work period. Winter is the only season for which the electricity price is always underestimated during the evening.

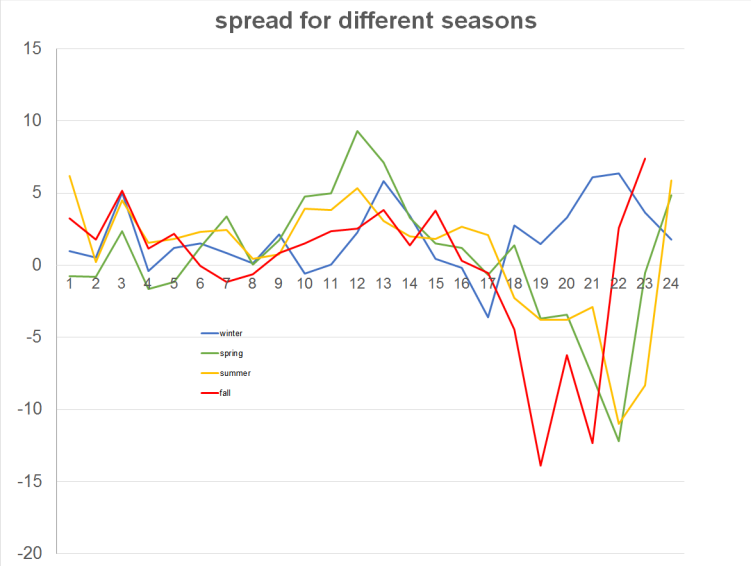


Fig. 7.4: Average hourly DA-RT price spread for the four seasons.

7.4 Variables

We aggregate the raw data into the following basic variables.

$$
\begin{aligned}
l_{j,h} &= \text{actual load of day } j, \text{ hour } h \\
d_{j,h} &= \text{day-ahead price of day } j, \text{ hour } h \\
r_{j,h} &= \text{real time price of day } j, \text{ hour } h \\
l_{j,h,1} &= \text{1-DA forecast load of day } j, \text{ hour } h \\
l_{j,h,2} &= \text{2-DA forecast load of day } j, \text{ hour } h \\
w_{j,h} &= \text{actual wind production of day } j, \text{ hour } h \\
w_{j,h,1} &= \text{1-DA forecast wind production of day } j, \text{ hour } h \\
w_{j,h,2} &= \text{2-DA forecast wind production of day } j, \text{ hour } h \\
s_{j,h} &= \text{actual solar production of day } j, \text{ hour } h \\
s_{j,h,1} &= \text{1-DA forecast solar production of day } j, \text{ hour } h \\
s_{j,h,2} &= \text{2-DA forecast solar production of day } j, \text{ hour } h \\
o_{j,h} &= \text{actual outage of day } j, \text{ hour } h \\
o_{j,h,1} &= \text{1-DA forecast outage of day } j, \text{ hour } h \\
o_{j,h,2} &= \text{2-DA forecast outage of day } j, \text{ hour } h
\end{aligned}
$$

The market players have to decide their supply or demand curves for day j in day $j-1$, based on the available information at this time. We assume that they use the most recent information only, i.e., the observations at day $j-2$. The past prediction errors and spreads have therefore to be computed for the day $j-2$.

$$\tilde{s}_{j-2,h} = \text{spread of day } j-2, \text{ hour } h$$
$$e^l_{j-2,h} = l_{j-2,h} - l_{j-2,h,1}$$
$$e^w_{j-2,h} = w_{j-2,h} - w_{j-2,h,1}$$
$$e^s_{j-2,h} = s_{j-2,h} - s_{j-2,h,1}$$

The predicted load and production evolutions are as follows.

$$\epsilon^l_{j,h} = l_{j,h,1} - l_{j-2,h}$$
$$\epsilon^{l,2}_{j,h} = l_{j,h,2} - l_{j-2,h}$$
$$\epsilon^w_{j,h} = w_{j,h,1} - w_{j-2,h}$$
$$\epsilon^s_{j,h} = s_{j,h,1} - s_{j-2,h}$$

Figures 7.5 and 7.6 show the average forecast errors of solar and wind production, respectively. More detailed representations are displayed in Figures 7.7 and 7.8.

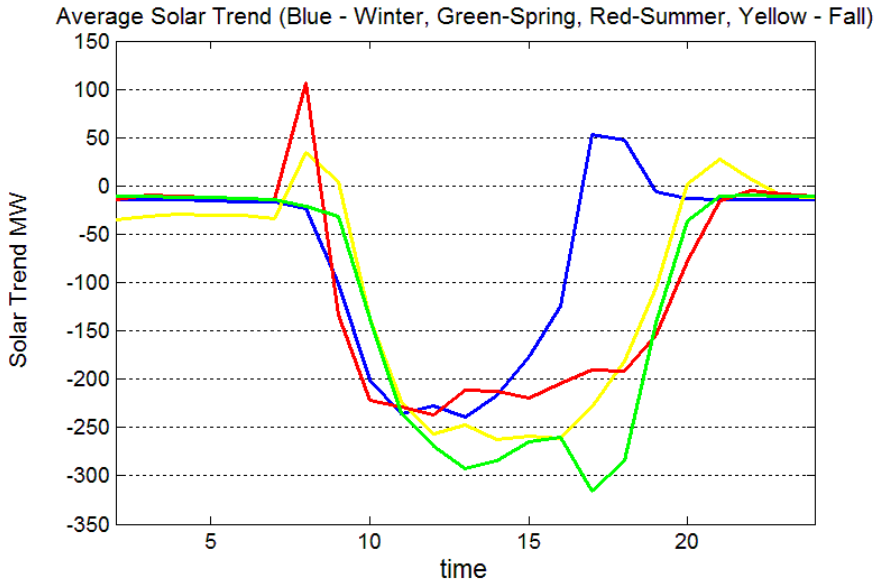


Fig. 7.5: Solar production forecast error.

7.5 Clustering

The previous graphs exhibit the presence of noise in the data, not only in the forecast of solar and wind production but also in the price spread. This makes it more difficult to predict the spread distribution and isolate the spikes (if there is a positive spread and no spike, then a short strategy should be used). On the contrary, for a negative spread, a long strategy is better. Without additional information, it is therefore difficult to make accurate predictions and to reduce the noise. An alternative strategy that we have explored

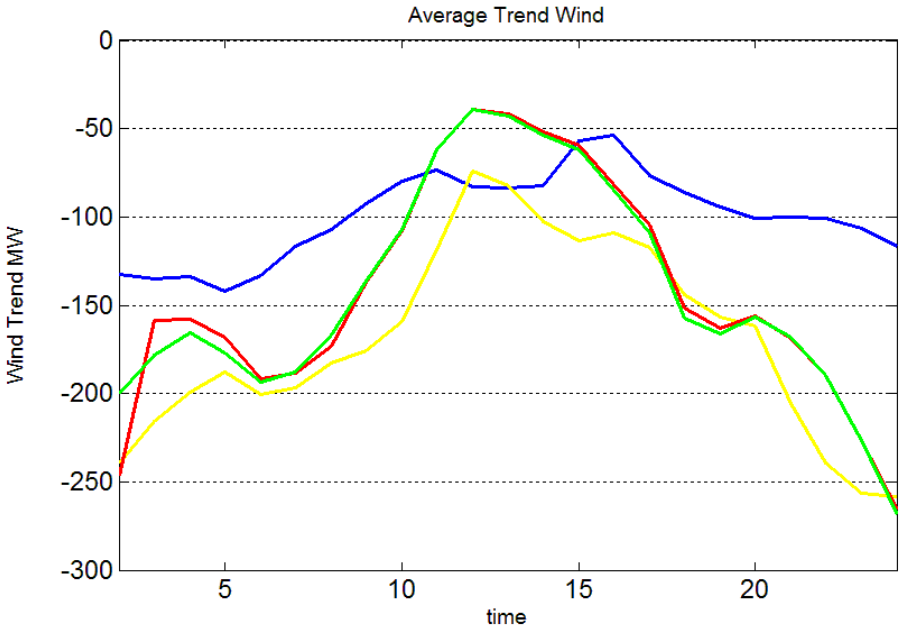


Fig. 7.6: Wind production forecast error.

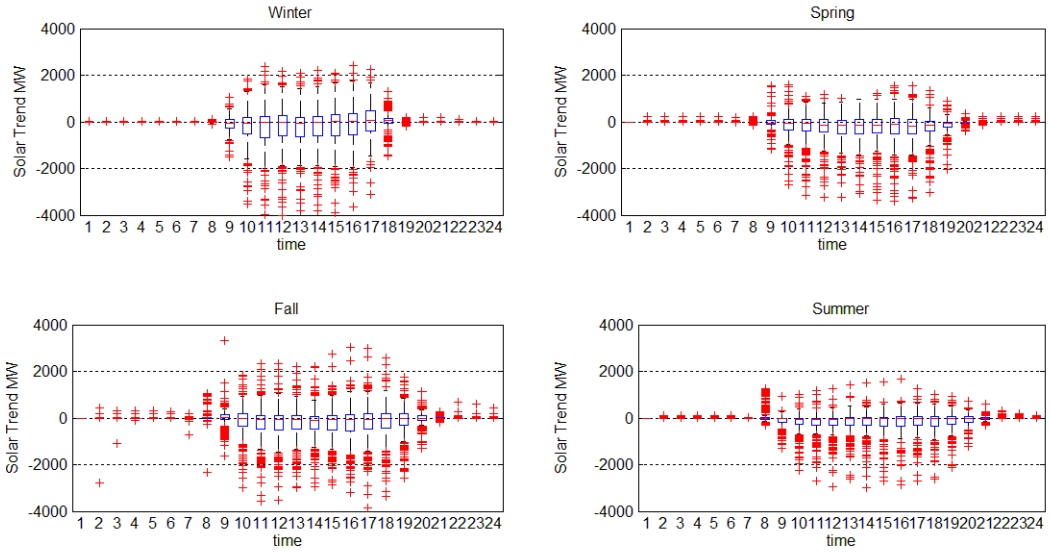


Fig. 7.7: Solar forecast errors.

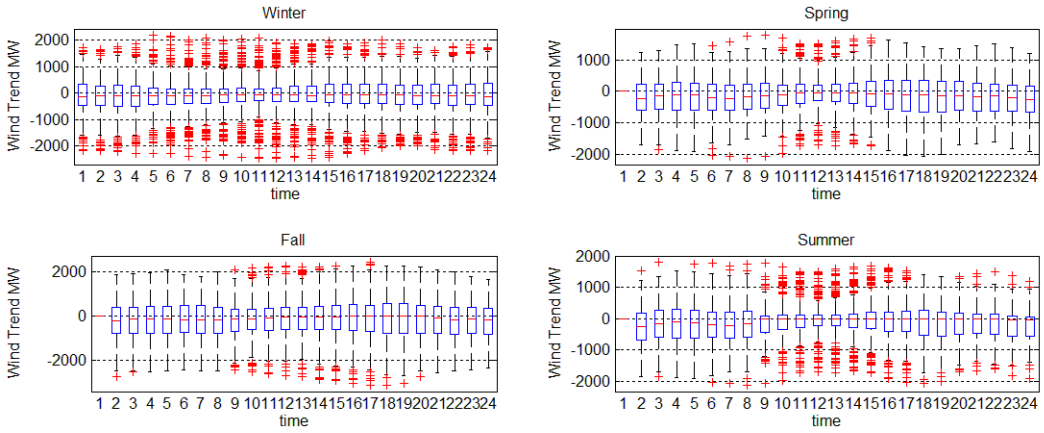


Fig. 7.8: Wind forecast errors.

is to use a classification, the rationale being that such a classification will enable us to find in the past a similar situation on a particular day and examine what happened the next day.

We chose to apply clustering, a set of techniques used to group data into subsets of similar observations called clusters. There are many ways of defining the notion of cluster and hence many different clustering algorithms. We used the k-means clustering algorithm, which aims to partition n observations into k clusters so as to minimize the within-cluster sum of squares (WCSS). After various experiments, we decided to split the data of each season into six clusters, based on the values of the variables mentioned before. Fewer clusters, for instance four, did not provide convincing results since they presented spreads that were too large.

We tried to isolate clusters in which most observations had the same sign. Also, for a short position, the cluster should include spreads that are positive most of the time and without negative spikes. Figures 7.9–7.12 display the spread distributions of the different clusters in each season.

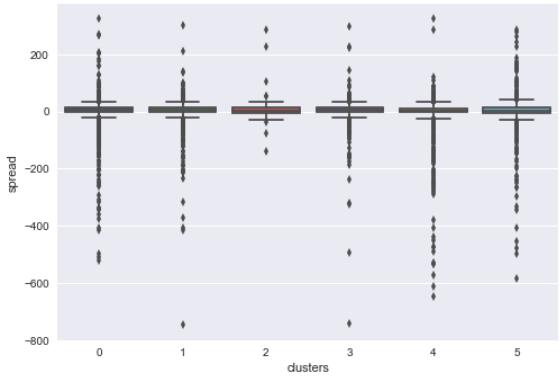


Fig. 7.9: Spring spread distribution.

We also had a look at the relationship between spread and DA price. For reasons of brevity, we only give Winter as an example (see Figure 7.13), but the other seasons show similar patterns.

The clusters offer a visual tool to detect the absence of large negative spikes, favouring a short position, while long positions should be prevalent when the spread average is small, close to 0, with the presence

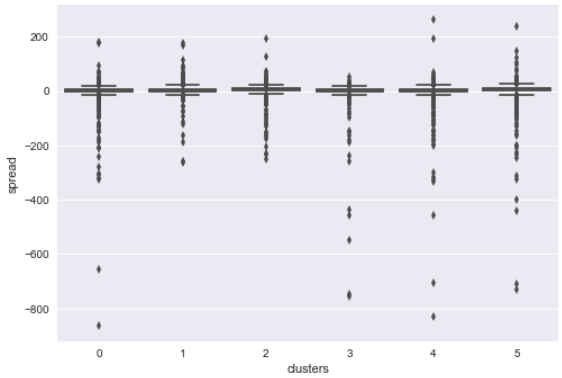


Fig. 7.10: Summer spread distribution.

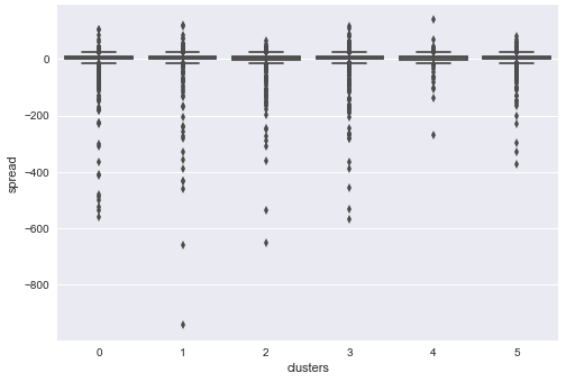


Fig. 7.11: Fall spread distribution.

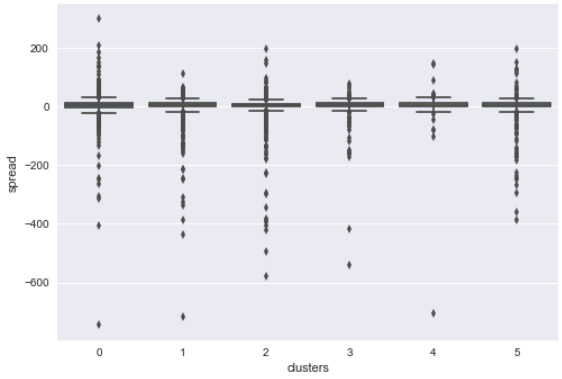


Fig. 7.12: Winter spread distribution.

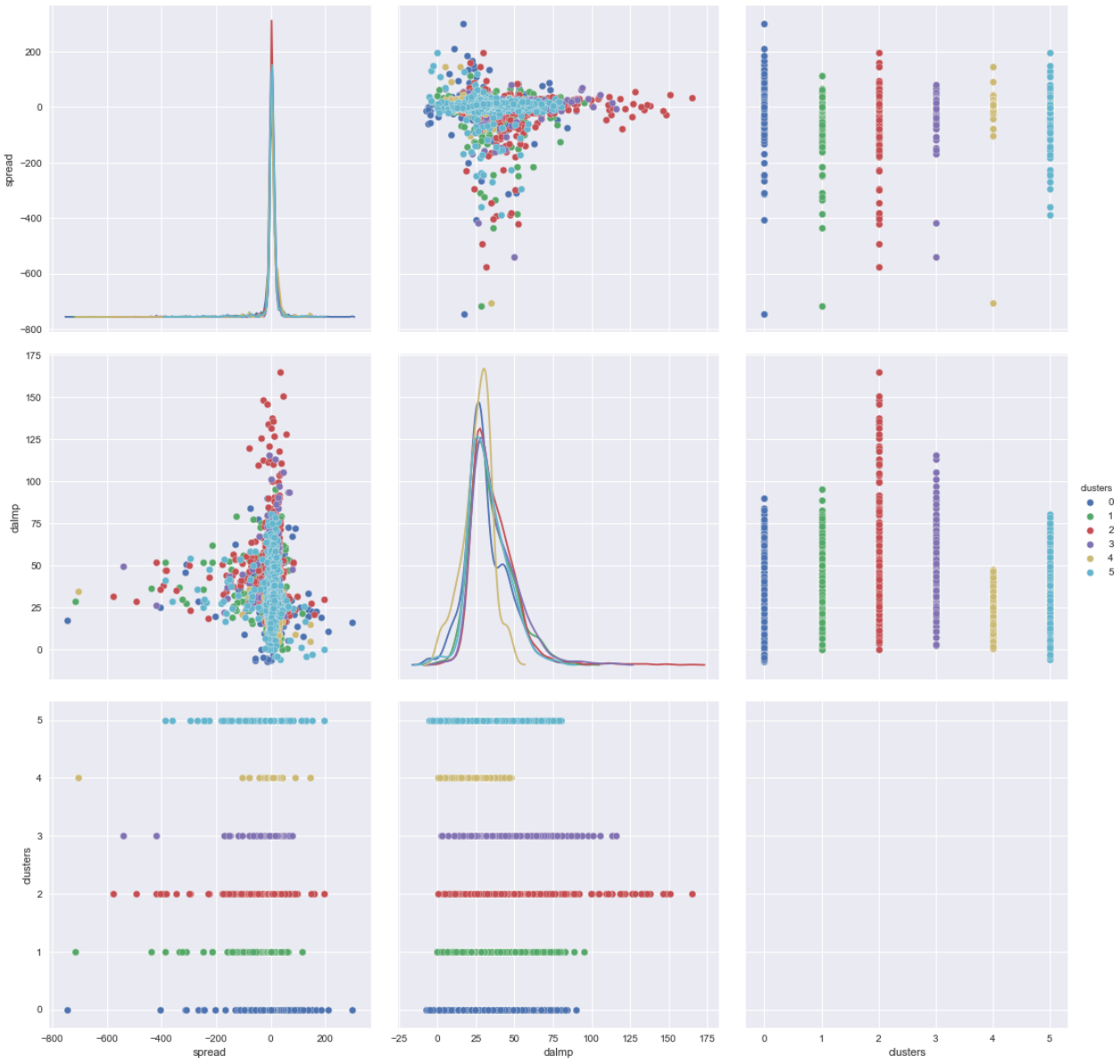


Fig. 7.13: Relationship between DA price and spread during the Winter.

of large negative spikes. While the current results could be improved, it seems likely that the classification approach will be able to provide guidelines to the decision-maker.

7.6 Bidding optimization problem

Ultimately, our goal is to maximize the expected hourly profit realized on the next day, that is to solve the problem

$$\begin{aligned}\max_{d^*_{j+1,h},q_{j+1,h},y^s_h,y^l_h} \quad & E[y^s_h(q_{j+1,h}(d^*_{j+1,h}-r_{j+1,h})\mathbb{1}_{d^*_{j+1,h}<d_{j+1,h}})+\\ & \quad y^l_h(q_{j+1,h}(r_{j+1,h}-d^*_{j+1,h})\mathbb{1}_{d^*_{j+1,h}>d_{j+1,h}})]\\ \text{s.t. } & y^s_h, y^l_h \in \{0,1\}\\ & 0 \le q_{j+1,h} \le u_{j+1,h}\\ & \text{risk aversion constraints,}\end{aligned}$$

where the superscript s stands for short and l for long, and $u_{j+1,h}$ is an upper bound on the quantity of electricity that we are ready to buy or sell. The binary variables y^s_h, y^l_h indicate whether we have adopted a short or a long position at the hour h of the day $j+1$. We have ignored the subscript $j+1$ to simplify the notation.

Note that the current program allows one to take a neutral position at the hour h by setting both y^s_h and y^l_h to zero, but also to adopt both a short and a long positions. We can prevent the latter situation by adding the constraint

$$y^s_h + y^l_h \le 1.$$

A small variant consists of enforcing different bounds depending on the chosen position. The program then becomes

$$\begin{aligned}\max_{d^*_{j+1,h},q_{j+1,h},y^s_h,y^l_h} \quad & E[y^s_h(q^s_{j+1,h}(d^*_{j+1,h}-r_{j+1,h})\mathbb{1}_{d^*_{j+1,h}<d_{j+1,h}})+\\ & \quad y^l_h(q^l_{j+1,h}(r_{j+1,h}-d^*_{j+1,h})\mathbb{1}_{d^*_{j+1,h}>d_{j+1,h}})]\\ \text{s.t. } & y^s_h, y^l_h \in \{0,1\}\\ & 0 \le q^s_{j+1,h} \le u^s_{j+1,h}\\ & 0 \le q^l_{j+1,h} \le u^l_{j+1,h}\\ & \text{risk aversion constraints.}\end{aligned}$$

The risk aversion contraints could be as follows:

$$E[\text{loss} \,|\, \text{loss} > \kappa] \le \lambda,$$

where κ and λ are predetermined constants.

The mathematical program we have just given is a stochastic program, which cannot be solved exactly. The solution of this program is beyond the scope of the present report, but an avenue to be explored is the optimization of a sample average approximation of the problem (see for instance Chapter 5 in [3]).

7.7 Future work

We have performed a preliminary analysis on the raw data and built a basic model using these data. While the results we obtained are promising, many improvements can be proposed. The choice of variables for the clustering approach should be investigated again, as there might exist other variables providing a better description of the market; also a more concise model might be proposed. Moreover alternative clustering techniques should be tested. Secondly, we should translate each cluster into a policy decision, incorporating a qualitative approach and human input. The next step would be to automatize the bidding strategy using stochastic programming. Thirdly, we should validate the proposed methods, for instance by performing backtests on historical data, and doing out-of-sample validation to prevent overfitting. Finally, we should include more details in the model, such as congestion and renewable energies other than wind energy and solar energy.

Reference

1. John Birg, Ali Hortaçsu, Ignacia Mercadal, and Michael Pavlin. Limits to arbitrage in electricity markets: A case study of MISO. Working Paper CEEPR WP 2017-003, MIT Center for Energy and Environmental Policy Research, January 2017.
2. Ignacia Mercadal. Dynamic competition and arbitrage in electricity markets: The role of financial players, January 2016.
3. Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lecture Notes in Stochastic Programming*. SIAM, Philadelphia, PA, 2009.

8
Modelling of the Friction Stir Welding Process for Aluminum Alloys

Kirk Fraser, Sean Bohun, Xiulei Cao, Huaxiong Huang, Kate Powers, Aina Rakotondrandisa, Mohammad Samani, and Zilong Song

Abstract In this report, we present the results from our modelling exercise during and shortly after the Eighth Montreal Industrial Problem Solving Workshop (IPSW). Kirk Fraser from the National Research Council of Canada presented a challenging problem. When the friction stir welding (FSW) process is used in practice, defects appear under uncertain operating conditions. The question was posed as to whether it is possible to understand the basic mechanisms of the defect generation so that it could be avoided. During the workshop, simplified models were developed for heat generation and plastic deformation and attempts were made to find analytical and numerical solutions, which were continued for a short period after the workshop had ended. We found that the unrealistic solution obtained during the workshop was due to the choice of a balance between inertial forces and plastic stress. In contrast, when inertia is negligible, physically reasonable solutions can be obtained for plastic deformation irrespective of the compressibility of the material. We investigated the dominant physical processes that drive the FSW process: the solutions from the simplified one-dimensional heat model and a two-dimensional non-Newtonian fluid model are presented here. The temperature distribution matches the results using direct numerical simulation and the experimental measurements. We did not have sufficient time, however, to solve the time-dependent one-dimensional thermal-plastic-elastic model. This latter problem has been left for future research.

8.1 Introduction

In many industrial applications, joining aluminum alloys and other metal workpieces can be accomplished through various approaches, such as adhesive bonding and welding. FSW is a relatively new technology

Kirk Fraser
NRC-CNRC

Sean Bohun
University of Ontario Institute of Technology

Xiulei Cao · Zilong Song
York University

Huaxiong Huang
York University and Fields Institute

Kate Powers
University of Bath

Aina Rakotondrandisa
Université de Rouen

Mohammad Samani
The Hospital for Sick Children

(patented in 1991), with the advantages of being more environmentally friendly and producing better joints than traditional approaches. The technique has gained popularity in aerospace, automotive, and marine industries.

FSW is a solid-state welding process (cf. Figure 1), where a hardened steel tool rotates and presses into the metal plates [1] that are to be welded [2]. The steel tool consists of a pin and a shoulder. Axial force is applied to the tool, to keep it in contact with the material, and the rotation of the pin and shoulder against the workpiece generates frictional heating that softens the aluminum. The material is not heated beyond the melting temperature as in traditional welding but rather reaches a temperature of about 80–90% of the melting temperature. While in this state, the heating and rotation of the tool induce a large plastic deformation and the metal in the plastic region behaves like a fluid, ultimately joining the plates.

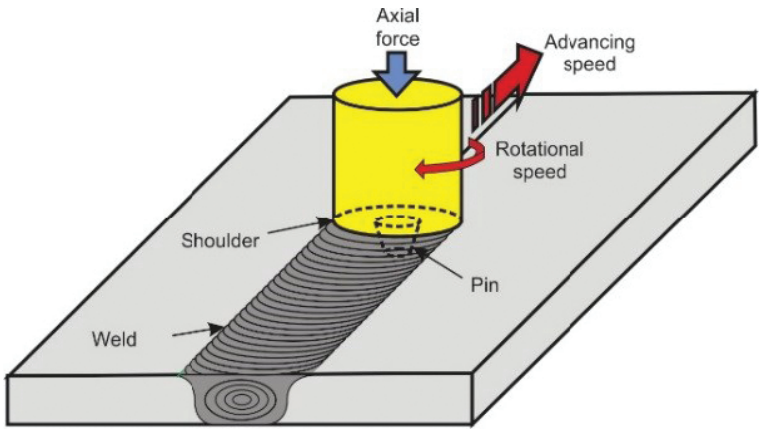


Fig. 8.1: Schematic illustration of the FSW process.

It has been observed experimentally that when the applied pressure or the rotational speed is too high, material may flow out of the tool to form uneven surface deposits. On the other hand, when the applied pressure or the rotational speed is too low, or the tool moves too fast, an incomplete joint may occur. The objective of this project is to understand the underlying mechanisms and provide guidance for improving the process. In particular two elements play an important role in the FSW process: heat generation and large plastic deformation. Revealing the mechanisms or influencing factors for these two aspects can shed light on the size of the plastic region and the possible generation of defects during the process.

During the workshop, we built a three-dimensional (3D) general plastic/elastic model coupled with heat transfer. With axisymmetry and the plane strain assumption, the general model was reduced to a one-dimensional (1D) model. To keep the problem tractable, we studied the heat generation and plastic deformation processes separately even though in practice they are interrelated. For the heat generation and temperature distribution problem we used a numerical method, while for the plastic models, analytical solutions were developed for two separate 1D scenarios. This work was continued after the workshop and allowed us to investigate the role of several key physical parameters inducing plastic deformation, such as tool pressure.

[1] We focus on aluminum plates in this report.

8.2 A General Model

In this section, we first introduce a general 3D model for the FSW process and illustrate various mechanisms involved in the process and how they are coupled.

We assume that the aluminum workpiece can be divided into two regions: an elastic and a plastic region. In the elastic region, away from the steel tool, we assume that the deformation is governed by linear elasticity. By ignoring the body forces, the force balance reads

$$\nabla \cdot \boldsymbol{\sigma} = \vec{0}, \tag{8.1}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress and the constitutive relation is given by

$$\boldsymbol{\sigma} = \lambda \operatorname{tr}(\boldsymbol{\varepsilon})\mathbf{I} + \mu\boldsymbol{\varepsilon} = \frac{E\nu}{(1+\nu)(1-2\nu)}\operatorname{tr}(\boldsymbol{\varepsilon})\mathbf{I} + \frac{E}{2(1+\nu)}\boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} = \frac{1}{2}(\nabla\vec{u} + \nabla\vec{u}^{\top}), \tag{8.2}$$

where $\boldsymbol{\varepsilon}$ is the strain tensor, $\vec{u}$ is the displacement vector, λ and μ are Lamé constants, E is Young's modulus, and ν is Poisson's ratio (see Table 1 for parameter values). The temperature field T is governed by the heat equation

$$\rho C_p \frac{\partial T}{\partial t} = \nabla \cdot (k\nabla T), \tag{8.3}$$

where ρ is the density, C_p is the specific heat capacity, and k is the thermal conductivity (see Table 1).

Parameter	Value	Units	Parameter	Value	Units
k	237	J/s.m.K	σ_{y0}	240	MPa
E	70	GPa	T_R	20	^{o}C
ν	0.35		T_{melt}	605	^{o}C
ρ	2700	kg/m^3	m	1.34	
C_p	897	J/kg.K			

Table 8.1: Thermal-physical properties of aluminum.

In the plastic region, near the tool, the force balance and incompressibility condition are

$$\nabla \cdot \boldsymbol{\sigma} = \rho\frac{D\vec{v}}{Dt}, \qquad \nabla \cdot \vec{v} = 0, \tag{8.4}$$

where $\vec{v} = \dot{\vec{u}}$ is the velocity and $\frac{D}{Dt} = \frac{\partial}{\partial t} + \vec{v}\cdot\nabla$ the material time derivative. For the constitutive relations, we adopt a perfect plasticity model with von Mises yield function [3, see chap 8], and more precisely

$$\dot{\boldsymbol{\varepsilon}} = \frac{\Lambda}{\sigma_y}(\boldsymbol{\sigma} - \frac{1}{3}\mathrm{Tr}(\boldsymbol{\sigma})\mathbf{I}) \equiv \frac{\Lambda}{\sigma_y}\boldsymbol{\sigma}^{\mathrm{dev}}, \qquad \sigma_y^2(T) = \frac{3}{2}\mathrm{tr}\left(\boldsymbol{\sigma}^{\mathrm{dev}}\boldsymbol{\sigma}^{\mathrm{dev}}\right), \tag{8.5}$$

where σ_y is the yield stress depending on temperature T and Λ is a Lagrange multiplier that in general can be a scalar function of position and time. We use a simple form of Johnson-Cook constitutive model for the yield stress [2, 4]:

$$\sigma_y^2(T) = \sigma_{y0}\left(1 - \left(\frac{T - T_R}{T_{melt} - T_R}\right)^m\right), \tag{8.6}$$

where σ_{y0} is the yield stress at room temperature, T_R and T_{melt} are the room and melt temperatures (respectively), and m is the thermal-softening exponent (see Table 1). The temperature field is governed by

the equation

$$\rho C_p \frac{DT}{Dt} = \nabla \cdot (k \nabla T) + \mathrm{tr}(\boldsymbol{\sigma}\dot{\boldsymbol{\varepsilon}}), \tag{8.7}$$

where the last term represents heat generation due to plastic flow.

In summary, we have presented a model for both the plastic and elastic regions of the material. To complete this model, we require suitable boundary conditions like convection of heat for temperature and displacement or traction conditions for the mechanical field, and propose connection conditions on the free boundary between the elastic and plastic regions. These conditions will be illustrated in later sections for the 1D case. In the governing equations above, the coupling of the mechanical and thermal fields comes mainly from two terms: $\sigma_y(T)$ and $\mathrm{tr}(\boldsymbol{\sigma}\dot{\boldsymbol{\varepsilon}})$.

8.3 Heat Generation

In this section, we will present a 1D model for the temperature field developed by our team. The simulation of the temperature field is credited to Mohammad Samani.

In this 1D model (cf. Figure 2), we consider the case where the shoulder of the tool presses on the material with some axial force N_{axial} and the shoulder and the pin rotate at a constant frequency ω. The axisymmetric case is considered and the temperature in the aluminum material is denoted by $T = T(r,t)$, where r denotes the radial distance of a point in the material from the pin axis (and $r_p < r < L$ holds). The temperatures of the pin and the shoulder are denoted by $T_p(t)$ and $T_s(t)$, respectively. In this model the effect of the velocity $\vec{v}$ in the plastic region is neglected: it will be studied in the future.

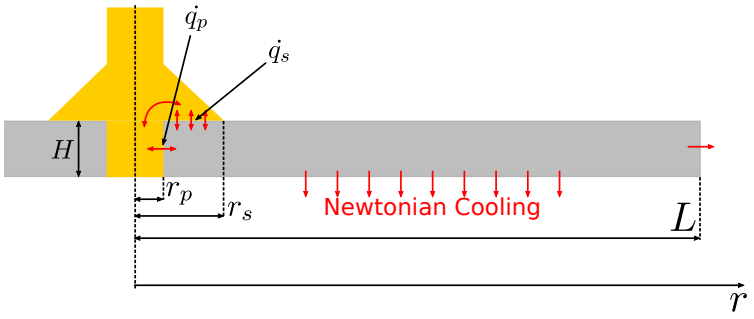


Fig. 8.2: Schematic illustration of the 1D heat generation model.

The governing equation is

$$\rho C_p \frac{\partial T}{\partial t} = \frac{1}{r}\frac{\partial}{\partial r}\left(rk\frac{\partial T}{\partial r}\right) + \dot{q}_s + \dot{q}_B, \quad r_p < r < L, \tag{8.8}$$

where $\dot{q}_s$ is the heat generation due to friction between the shoulder and the material and $\dot{q}_B$ the cooling due to the convection of heat towards the base material. They are given by the following formulas (respectively):

$$\begin{aligned} \dot{q}_s(r) &= f\frac{N_{\text{axial}}}{\pi(r_s^2 - r_p^2)} 2\pi\omega r \frac{1}{H} - h_p(T - T_p)\frac{1}{H}, \quad r_p < r < r_s, \\ \dot{q}_B(r) &= -h_B(T - T_B)\frac{1}{H}, \quad r_p < r < L, \end{aligned} \tag{8.9}$$

where f is the frictional coefficient between aluminum and the tool, H is the thickness of the material, h_B is the convection coefficient between the workpiece and the supporting material, and T_B is equal to the room temperature (see Table 2 for the parameter values). The temperatures of the pin and the shoulder are governed by the following equations (respectively):

$$\begin{aligned} \rho_p c_p V_p \frac{\mathrm{d}T_p}{\mathrm{d}t} &= \dot{q_p} + h_{ps}(T_p - T_s) + h_p(T_p - T(r_p)), \\ \rho_s c_s V_s \frac{\mathrm{d}T_s}{\mathrm{d}t} &= -h_{ps}(T_p - T_s) + h_s(T_s - T|_{(r_p,r_s)}), \end{aligned} \tag{8.10}$$

where $T|_{(r_p,r_s)}$ is the mean temperature of the material in the interval, $V_p = \pi r_p^2 H$ is the volume of the pin, V_s is the volume of the shoulder (we set $V_s = 100V_p$), c_p and c_s are the heat capacities of the pin and shoulder (respectively), h_p, h_s, and h_{ps} are some convection coefficients, and $\dot{q_p}$ is the heat generation caused by the rotation of the pin. The heat generation $\dot{q_p}$ is given by the equation

$$\dot{q_p} = \sigma_{r\theta} v_\theta A_p, \quad v_\theta = \gamma 2\pi\omega r_p, \quad A_p = \pi r_p^2 + 2\pi r_p H, \tag{8.11}$$

where $\sigma_{r\theta}$ and v_θ are the shear stress and velocity (respectively) and A_p is the surface area of the pin (cf. Table 2).

Parameter	Value	Units	Parameter	Value	Units
N_{axial}	10	kN	$\sigma_{r\theta}$	100	MPa
r_p	5	mm	h_p, h_s, h_B	10^4	$\mathrm{J/s \cdot m^2 \cdot K}$
r_s	15	mm	h_L	5000	$\mathrm{J/s \cdot m^2 \cdot K}$
L	100	mm	ρ_s	8170	$\mathrm{kg/m^3}$
H	10	mm	c_p, c_s	418	J/kg.K
ω	1400/60	1/s	h_{ps}	10^5	$\mathrm{J/s \cdot m^2 \cdot K}$
f	0.5		γ	0.9	

Table 8.2: Data used in the simulation of the temperature field.

The boundary conditions are

$$\begin{aligned} -k\frac{\partial T}{\partial r} &= h_p(T - T_p), \quad \text{at } r = r_p, \\ -k\frac{\partial T}{\partial r} &= h_L(T - T_R), \quad \text{at } r = L, \end{aligned} \tag{8.12}$$

and the initial value of any of the temperatures is set equal to the room temperature. Figure 3 shows the temperature field inside the aluminum workpiece, given the data in Table 2. The temperature in the region under the shoulder is higher, suggesting that the shoulder is mostly responsible for the heating. Since σ_y is a decreasing function of T, the region under the shoulder will first yield due to a combination of pressure and shear stresses. The time for reaching the desirable temperature in the workpiece matches experimental observations.

8.4 A One-Dimensional Plastic-Elastic Model

In this section, we consider a 1D plastic-elastic problem where the yield stress σ_y is assumed to be a constant. For the plastic deformation, we consider two different plastic models. Our first plastic model follows from the general framework of the previous section, which uses the von Mises yield criterion and assumes that the plastic flow is incompressible. The second plastic model assumes that the material is compressible with a

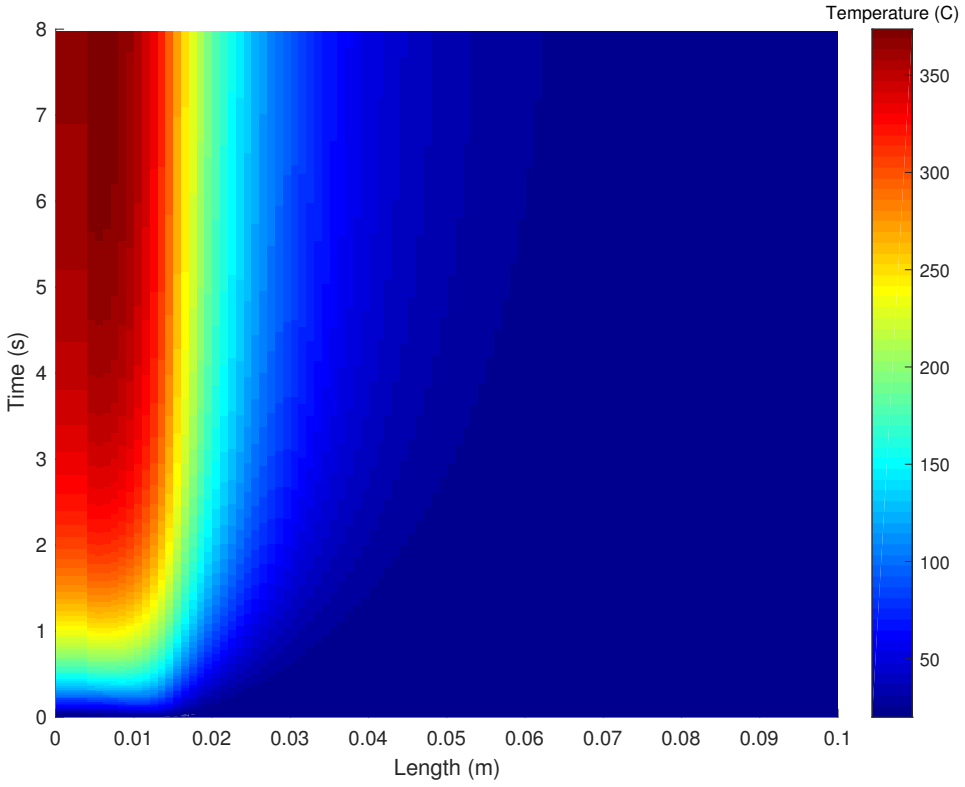


Fig. 8.3: Evolution of the temperature field.

simpler yield criterion: it is used in a model for plastic deformation in a gun barrel problem [3]. The work in the present section was mainly carried out by the two faculty members (Sean Bohun and Huaxiong Huang), a student (Kate Powers), and a postdoc (Zilong Song), with valuable input from Kirk Fraser. Zilong Song reexamined the model and produced most of the results presented in this section after the workshop.

We consider a plane strain problem and use polar coordinates (r, θ) and z for the thickness of plate in the formulation. By a 1D model we mean that all the functions are functions of r only. The displacement in the elastic region is $\vec{u} = \langle u_r(r), u_\theta(r), 0\rangle$, and the velocity in the plastic region is $\vec{v} = \langle v_r(r), v_\theta(r), 0\rangle$. In the following, we will consider an infinite domain $r_p < r < \infty$, which is divided into two regions: the plastic region $r_p < r < r_b$ and the elastic region $r_b < r < \infty$, where r_b is the location of the free boundary between them. We will study the elastic and plastic problems separately and match the solutions by using jump conditions on the plastic-elastic interface (a free boundary).

8.4.1 *1D Elastic Model*

In the elastic region $r_b < r < \infty$, the force balance becomes

$$\frac{\partial \sigma_{rr}}{\partial r} + \frac{1}{r}(\sigma_{rr} - \sigma_{\theta\theta}) = 0, \qquad \frac{\partial \sigma_{r\theta}}{\partial r} + \frac{2}{r}\sigma_{r\theta} = 0, \tag{8.13}$$

with the simplified constitutive relations

$$\begin{aligned}
&\sigma_{rr} = (\lambda + 2\mu)\varepsilon_{rr} + \lambda\varepsilon_{\theta\theta}, \quad \sigma_{r\theta} = 2\mu\varepsilon_{r\theta}, \\
&\sigma_{\theta\theta} = (\lambda + 2\mu)\varepsilon_{\theta\theta} + \lambda\varepsilon_{rr}, \quad \sigma_{zz} = \lambda(\varepsilon_{\theta\theta} + \varepsilon_{rr}), \\
&\varepsilon_{rr} = \frac{\partial u_r}{\partial r}, \quad \varepsilon_{r\theta} = \frac{1}{2}\left(\frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}\right), \quad \varepsilon_{\theta\theta} = \frac{u_r}{r},
\end{aligned} \tag{8.14}$$

where $\varepsilon_{rr}, \varepsilon_{r\theta}, \varepsilon_{\theta\theta}$ and $\sigma_{rr}, \sigma_{r\theta}, \sigma_{\theta\theta}$ are respectively the in-plane components of the strain tensor ε and the stress tensor σ, and σ_{zz} is the stress component along the thickness direction z.

The boundary conditions are

$$\begin{aligned}
&u_r = u_\theta = 0, \quad \text{as } r \to \infty \\
&\sigma_{rr}(r) = \sigma^b_{rr}, \quad \sigma_{r\theta}(r) = \sigma^b_{r\theta}, \quad \text{at } r = r_b,
\end{aligned} \tag{8.15}$$

where $\sigma^b_{rr}, \sigma^b_{r\theta}$ are unknown constants to be determined by matching the solutions in the plastic region.

The solution of the above linear elastic problem can be readily obtained. The two displacements and non-zero stresses are given by

$$\begin{aligned}
&u_r = -\sigma^b_{rr}\frac{1+\nu}{E}\frac{r_b^2}{r}, \quad u_\theta = -\sigma^b_{r\theta}\frac{1+\nu}{E}\frac{r_b^2}{r}, \\
&\sigma_{rr} = \sigma^b_{rr}\frac{r_b^2}{r^2}, \quad \sigma_{\theta\theta} = -\sigma^b_{rr}\frac{r_b^2}{r^2}, \quad \sigma_{r\theta} = \sigma^b_{r\theta}\frac{r_b^2}{r^2}.
\end{aligned} \tag{8.16}$$

One can see that the displacement decays like r^{-1} while stresses decay like r^{-2}.

8.4.2 *1D Incompressible Plastic Model*

In this plastic model we use the same formulation as in the previous section but reduce it to one dimension. We consider the steady state (quasi-static process) and the term $\partial\vec{v}/\partial t$ will be neglected. As this is a 1D problem, the partial derivatives are actually total derivatives but we retain the partial derivative notation.

In the plastic region $r_p < r < r_b$, the force balance and incompressibility conditions are the following.

$$\begin{aligned}
&\rho\left(v_r\frac{\partial v_r}{\partial r} - \frac{v_\theta^2}{r}\right) = \frac{\partial\sigma_{rr}}{\partial r} + \frac{1}{r}\left(\sigma_{rr} - \sigma_{\theta\theta}\right) \\
&\rho\left(v_r\frac{\partial v_\theta}{\partial r} + \frac{v_r v_\theta}{r}\right) = \frac{\partial\sigma_{r\theta}}{\partial r} + \frac{2}{r}\sigma_{r\theta} \\
&\frac{\partial v_r}{\partial r} + \frac{v_r}{r} = 0
\end{aligned} \tag{8.17}$$

The constitutive relations and yield criterion are as follows.

$$\begin{aligned}
&\sigma_{rr} = p + \frac{\sigma_y}{\Lambda}\frac{\partial v_r}{\partial r}, \quad \sigma_{\theta\theta} = p + \frac{\sigma_y}{\Lambda}\frac{v_r}{r} \\
&\sigma_{r\theta} = \frac{\sigma_y}{2\Lambda}\left(\frac{\partial v_\theta}{\partial r} - \frac{v_\theta}{r}\right), \quad p = \frac{1}{3}\left(\sigma_{rr} + \sigma_{\theta\theta} + \sigma_{zz}\right) \\
&(\sigma_{rr} - p)^2 + (\sigma_{\theta\theta} - p)^2 + (\sigma_{zz} - p)^2 + 2\sigma_{r\theta}^2 = \frac{2\sigma_y^2}{3}
\end{aligned} \tag{8.18}$$

The boundary conditions are

$$\begin{aligned} &v_\theta = \gamma 2\pi\omega r_p, \quad \sigma_{r\theta} = f\sigma_{rr} \quad \text{at } r = r_p, \\ &v_\theta = 0, \quad p = p_b \quad \text{at } r = r_b, \end{aligned} \tag{8.19}$$

where p_b is assumed to be known for the time being, ω is the rotational frequency of the pin, $0 < \gamma < 1$ is a dimensionless scalar, and f is the frictional coefficient.

In this system, there are 4 unknowns: v_r, v_θ, Λ, and p (or equivalently σ_{rr}). One can easily express v_r and Λ analytically with two integrating constants c_1, c_2.

$$v_r = \frac{c_1}{r}, \quad \Lambda = \left(\frac{3\sigma_y^2 c_1^2}{r^4(\sigma_y^2 - 3\sigma_{r\theta}^2)} \right)^{\frac{1}{2}}, \quad \sigma_{r\theta} = \frac{\rho c_1 v_\theta}{r} + \frac{c_2}{r^2} \tag{8.20}$$

Then the system is reduced to two equations for v_θ and σ_{rr}.

$$\frac{\partial v_\theta}{\partial r} = \left(\frac{1}{r} + \frac{2\Lambda\rho c_1}{\sigma_y r} \right) v_\theta + \frac{2\Lambda c_2}{\sigma_y r^2} \tag{8.21}$$

$$\frac{\partial \sigma_{rr}}{\partial r} = \frac{2\sigma_y c_1}{\Lambda r^3} - \frac{\rho c_1^2}{r^3} - \frac{\rho v_\theta^2}{r} \tag{8.22}$$

In principle, the two functions and integrating constants c_1, c_2 can be determined by the four boundary conditions. Note that σ_{rr} and p are related by $\sigma_{rr} = p - \frac{\sigma_y c_1}{\Lambda r^2}$.

This system of two ODEs was solved during the workshop by Kate Powers using the shooting method. To solve the system one chooses a value of c_1 and then solves (8.21), iterating on c_2 until the boundary condition $v_\theta(r_b) = 0$ is satisfied. Armed with values of c_1 and c_2, equation (8.22) is solved and c_1 may be updated so that the condition $\sigma_{r\theta}(r_p) = f\sigma_{rr}(r_p)$ holds. The solution is obtained when these nested processes converge simultaneously. Figure 8.4 shows the resulting MATLAB plots, with parameters $\omega = 1000/60$ 1/s, $\gamma = 1/2$, $f = 1/5$, $r_b = 3r_p = 0.015$ m, $p_b = 0$. We observe that these results are not realistic because the radial velocity is extremely high, implying that the material is moving away from the pin and creating a large hole.

To investigate the reason for the unrealistic solution, we carry out a dimensional analysis. We use a prime to denote a dimensionless quantity, and adopt the scales

$$\begin{aligned} &r = r_p r', \quad (\boldsymbol{\sigma}, p) = \sigma_y(\boldsymbol{\sigma}', p'), \quad (v_r, v_\theta) = V(v_r', v_\theta'), \quad \Lambda = \bar{\Lambda}\Lambda', \\ &c_1 = C_1 c_1', \qquad c_2 = C_2 c_2', \end{aligned} \tag{8.23}$$

where $V, \bar{\Lambda}, C_1, C_2$ are some characteristic values. If we balance inertia and stress by choosing

$$\rho V^2 = \sigma_y, \qquad C_1 = r_p V, \qquad \bar{\Lambda} = V/r_p, \qquad C_2 = \sigma_y r_p^2, \tag{8.24}$$

then all the terms in the two ODEs are in the same order. In this situation, from the data in Table 1, $\sigma_y \sim 10^8$ Pa, $\rho \sim 10^3$ kg/m^3, and the resulting speed is of the order of 10^2–10^3 m/s, explaining the large velocity of v_r in Figure 8.4. For the FSW technique, balancing the inertial forces with the characteristic yield stress requires the corresponding characteristic speed to be very large. The fact that these large velocities are not experienced indicates that inertial forces do not dominate. Other scales are also possible for this system.

A more reasonable scaling would be to choose the characteristic speed to match the axial velocity of the rotating pin radius, resulting in

$$V = 2\pi\omega r_p, \qquad C_1 = r_p V, \qquad \bar{\Lambda} = V/r_p, \qquad C_2 = \sigma_y r_p^2 \tag{8.25}$$

and with typical values of $\rho = 2.7 \times 10^3$ kg/m^3, $\sigma_y = 10^8$ Pa, $r_p = 0.5 \times 10^{-2}$ m, so that $V = 0.5236$ m/s holds. Scaling in this fashion indicates that even with the significant rotation rates involved in the process, $\rho V^2 \ll \sigma_y$ holds, meaning that the inertial terms are dominated by the yield stress. In practice this implies that the inertial terms with ρ in the governing equations can be neglected.

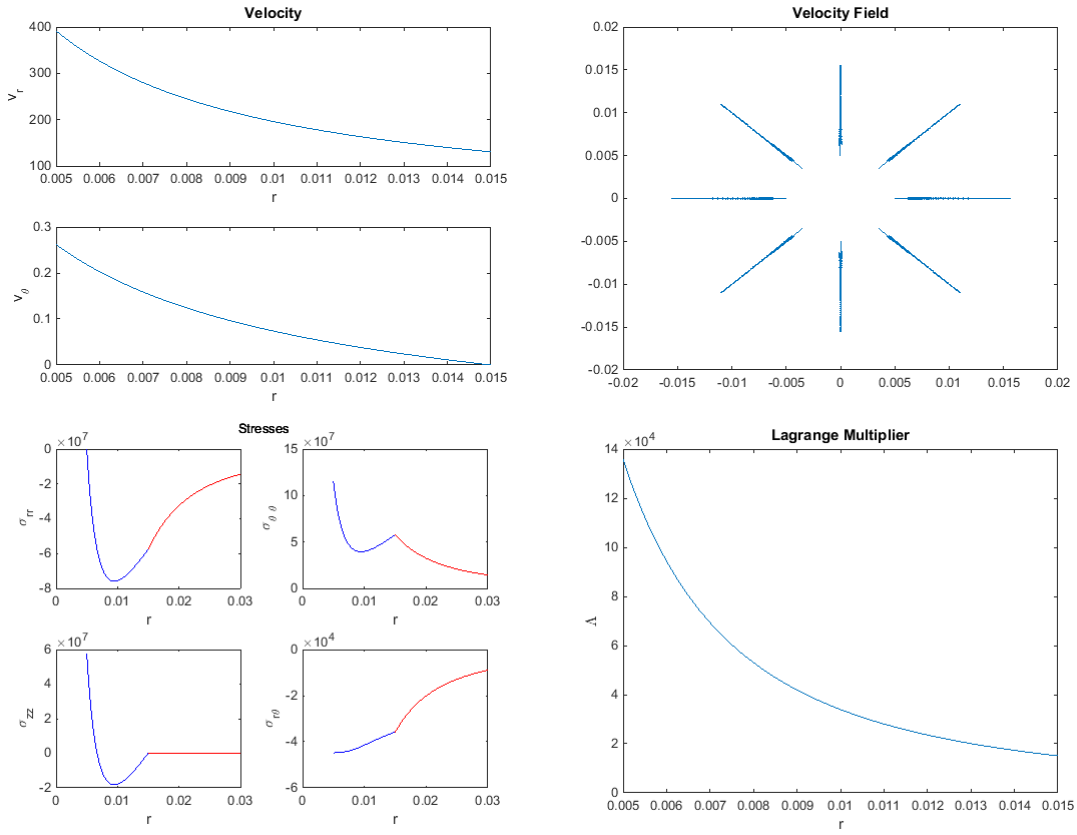


Fig. 8.4: Numerical results for the solution of the 1D incompressible plastic model by the shooting method. The red lines in the stress plot are the stresses in the elastic region, which we can see match up at the boundary.

By neglecting all the inertial terms, the system can be solved analytically with four integrating constants. We now present the solution in dimensionless quantities and for convenience the prime is removed (e.g., in the following formula r stands for r').

$$
\begin{aligned}
&v_r = \frac{c_1}{r}, \quad \sigma_{r\theta} = \frac{c_2}{r^2}, \quad \sigma_{rr} = p - \frac{c_1}{\Lambda r^2}, \quad \Lambda = \left(\frac{3c_1^2}{r^4 - 3c_2^2}\right)^{1/2}, \\
&v_\theta = \frac{\sqrt{3}c_1^2}{c_2 r}\left(\frac{r^4 - c_2^2}{c_1^2}\right)^{1/2} + c_3 r, \\
&\sigma_{rr} = \frac{c_1}{\sqrt{3}|c_1|r^2}\left(r^2 \log\left(r^2 + \sqrt{r^4 - c_2^2}\right) - \sqrt{r^4 - c_2^2}\right) + c_4.
\end{aligned}
\tag{8.26}
$$

The four $O(1)$ integrating constants can be determined easily from four boundary conditions. For example, with the same data $\gamma = 1/2, f = 1/5, p_b = 0, r_b = 3$ as in the solution obtained earlier (but in dimensionless form), we get

$$
c_1 = 1.5447, \quad c_2 = -0.3652, \quad c_3 = 7.3195, \quad c_4 = -1.6685. \tag{8.27}
$$

This numerical example shows that the original quantities v_θ and v_r are of the same order as $V = 2\pi\omega r_p$, which is more realistic than the previous calculations (cf. Figure 6 for the stress distribution).

To complete the present subsection, we match the plastic solution with the elastic one. We adopt the jump conditions at the free boundary r_b, i.e.,

$$v_\theta = 0, \quad [\sigma_{rr}] = [\sigma_{r\theta}] = [\sigma_{\theta\theta}] = 0, \quad r = r_b, \tag{8.28}$$

which implies that $p_b = 0$ holds. We need one extra condition to solve the system, since r_b is not known. One possibility is to consider a quasi-static process, from which the evolution of the free boundary r_b follows:

$$\frac{dr_b}{dt} = v_r(r_b), \qquad r_b(0) = 2r_p, \tag{8.29}$$

where the initial position of the free interface is set to be the circumference of the shoulder $r_s = 2r_p$. Figure 5 shows the evolution of r_b with parameters $\gamma = 1/2, f = 0.2$ on the boundary. Figure 6 shows the stress distributions in both the elastic and plastic regions for two cases: $r_b = 2$ and $r_b = 3$.

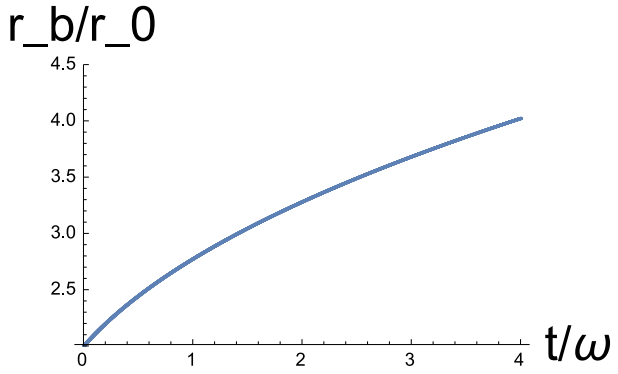


Fig. 8.5: Evolution of r_b.

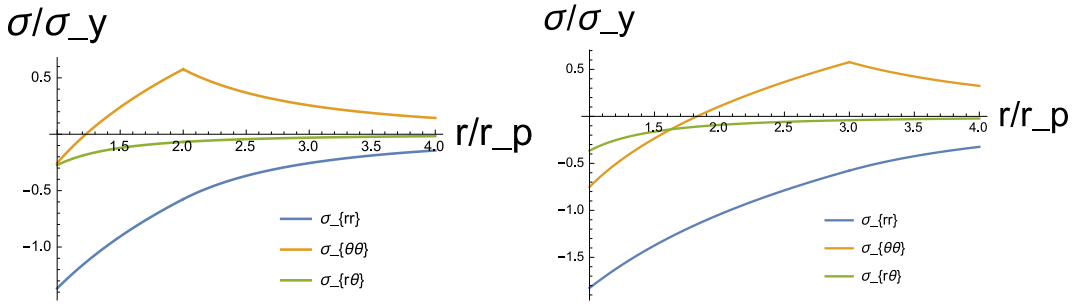


Fig. 8.6: Distribution of stresses with $r_b = 2$ and $r_b = 3$.

Another possibility is to use approximate conditions from the pressure of the shoulder on the top surface of the material. Since there is a singularity at the circumferential boundary of the shoulder, we cannot use this approximation for the whole region. We only use such conditions at the two ends, namely at $r = r_p$ and $r = r_b$. Or we set $p(r_p) = \sigma_{zz}(r_p) = P_0$, where P_0 is the pressure by the axial force generated by the shoulder.

The condition at $r = r_b$ is $p(r_b) = \sigma_{zz}(r_b) = 0$; it is consistent with the previous condition $p_b = 0$. With the extra condition at r_p, we can determine r_b: for example, if $P_0 = -\sigma_y$ holds, we obtain $r_b/r_p = 2.349$.

Numerical results show that the larger the pressure P_0, the smaller the radial velocity v_r. Thus the pressure has the effect of preventing the material from going away from the pin. Figure 7 shows the dependence of r_b/r_p on the parameters $-P_0/\sigma_y$ and f. The plastic region becomes larger when pressure increases, which indicates that pressure from the shoulder plays an important role in yielding.

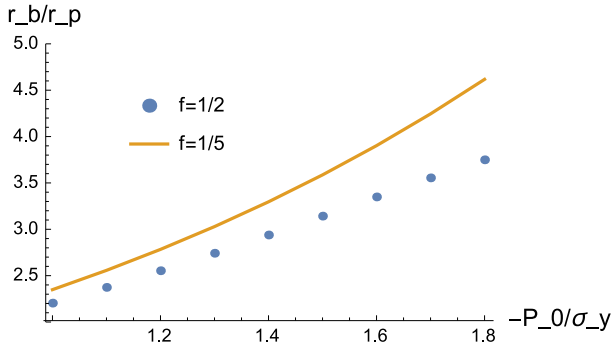


Fig. 8.7: Dependence of r_b/r_p on P_0/σ_y and f.

8.4.3 *1D Compressible Plastic Model*

The constitutive relations for the previous plastic model imply the incompressibility condition, which may be inconsistent with the plane strain assumption (implying uniformity in the thickness direction). In this subsection, we relax the incompressibility assumption and consider the force balance with a simpler yield criterion, which is used for a gun barrel problem in [3].

The inertial terms are assumed to be negligible, so the governing equations and yield criterion are

$$\frac{\mathrm{d}\sigma_{rr}}{\mathrm{d}r} + \frac{1}{r}(\sigma_{rr} - \sigma_{\theta\theta}) = 0, \qquad \frac{\mathrm{d}\sigma_{r\theta}}{\mathrm{d}r} + \frac{2}{r}\sigma_{r\theta} = 0, \qquad \frac{1}{4}(\sigma_{rr} - \sigma_{\theta\theta})^2 + \sigma_{r\theta}^2 = \sigma_y^2(T), \tag{8.30}$$

where the yield stress $\sigma_y(T)$ can be a function of the temperature T and hence a function of r. Being consistent with the previous boundary conditions and matching conditions, we take

$$\begin{aligned} &\sigma_{rr} = P_r, \quad \sigma_{r\theta} = f\sigma_{rr} \quad \text{at } r = r_p, \\ &[\sigma_{rr}] = [\sigma_{r\theta}] = 0 \quad \text{at } r = r_b, \\ &(\sigma_{rr}^b)^2 + (\sigma_{r\theta}^b)^2 = \sigma_y^2 \quad \text{or} \quad \sigma_{rr}^b + \sigma_{\theta\theta}^b = 0 \quad \text{at } r = r_b, \end{aligned} \tag{8.31}$$

where $P_r < 0$ holds and $|P_r|$ is the physical pressure on the surface of the pin. The continuity of $\sigma_{\theta\theta}$ is tacitly assumed in the last equation.

Solutions to this system are readily obtained.

$$\begin{aligned} \sigma_{r\theta}(r) = fP_r\frac{r_p^2}{r^2}, \quad \sigma_{rr}(r) &= \pm\int_{r_p}^{r}\frac{2}{s}\sqrt{\sigma_y^2(s) - \sigma_{r\theta}^2(s)}\,\mathrm{d}s + P_r \\ \sigma_{\theta\theta}(r) &= \sigma_{rr}(r) \pm 2\sqrt{\sigma_y^2(r) - \sigma_{r\theta}^2(r)} \end{aligned} \tag{8.32}$$

A later calculation showed that the solution with the positive sign is the correct one. In the special case where σ_y is constant, the explicit solution is

$$\begin{aligned} \sigma_{rr}(r) &= f(r) - f(r_p) + P_r, \\ f(r) &= \sigma_y \log\left(\sigma_y\left(\sqrt{r^4\sigma_y^2 - f^2P_r^2r_p^4} + r^2\sigma_y\right)\right) - \frac{1}{r^2}\sqrt{r^4\sigma_y^2 - f^2P_r^2r_p^4}. \end{aligned} \tag{8.33}$$

The three matching conditions uniquely determine the free boundary r_b and the two matching parameters $\sigma^b_{rr}, \sigma^b_{r\theta}$ of the elastic solution.

Indeed, the above solution is a generalization of the gun barrel problem. For the special case $f = 0$, i.e., the case where plasticity is induced by pressure only, we recover the solution in the gun barrel problem [3, pp. 351-352]. In this case, the plastic solution reads

$$\sigma_{r\theta} = 0, \qquad \sigma_{rr} = 2\sigma_y \ln\left(\frac{r}{r_p}\right) + P_r, \qquad \sigma_{\theta\theta} = 2\sigma_y \ln\left(\frac{r}{r_p}\right) + P_r + 2\sigma_y \tag{8.34}$$

and the yielding position r_b is explicitly given by

$$r_b = r_p \exp\left(\frac{-P_r}{2\sigma_y} - \frac{1}{2}\right), \tag{8.35}$$

which indicates plastic deformation when $|P_r| > \sigma_y$ holds.

For the data $\left\{f \to 1/2, P_r \to -1.5 \times 10^8 \text{ Pa}, \sigma_y \to 10^8 \text{ Pa}, r_p \to 0.005 \text{ m}\right\}$, we get a numerical value of r_b satisfying $r_b \approx 0.007062$m. Figure 8 shows the stress distribution for both the elastic and plastic regions. Figure 9 shows the dependence of r_b/r_p on the parameters $-P_r/\sigma_y$ and f. As the pressure P_r or frictional coefficient f increases, the plastic region becomes larger.

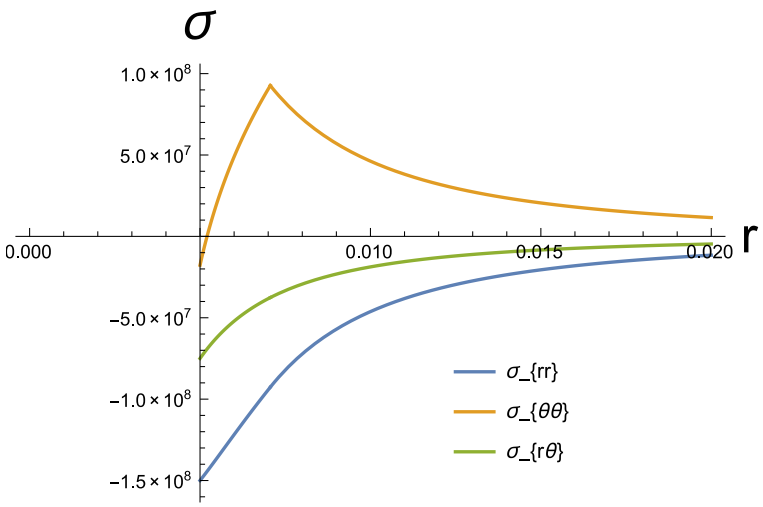


Fig. 8.8: Distribution of stresses predicted by plastic model II.

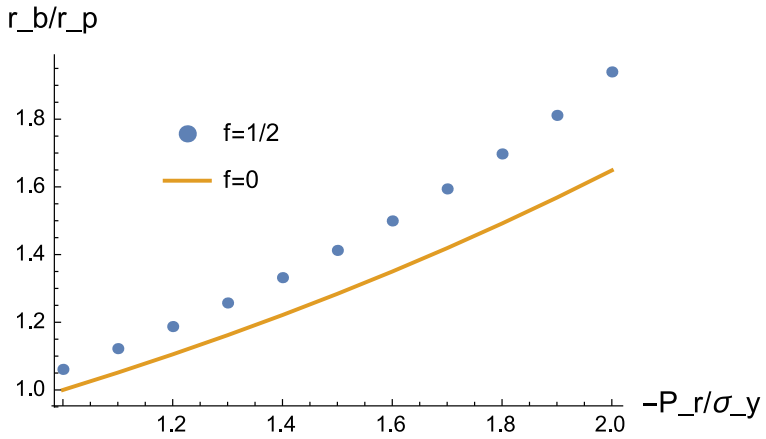


Fig. 8.9: Dependence of r_b/r_p on the parameters $-P_r/\sigma_y$ and f.

8.5 Direct Numerical Solutions

During the workshop, direct numerical simulations were carried out. One was based on a 2D non-Newtonian fluid model for material flow in the plastic region [1], and the other was based on the previous 1D dynamical system. We achieved partial success but also encountered some problems in obtaining desirable results.

We consider the 2D non-Newtonian fluid model for the velocity $\vec{v} = \langle v_r(r), v_\theta(r), 0\rangle$ coupled with a heat equation,

$$\begin{aligned} &\nabla \cdot (2\mu\dot{\varepsilon}) - \nabla p = 0, \quad \nabla \cdot \vec{v} = 0, \\ &\rho C_p \frac{DT}{Dt} = \nabla \cdot (k\nabla T) + \alpha \mathrm{tr}\,(2\mu\dot{\varepsilon}\dot{\varepsilon})\,, \end{aligned} \tag{8.36}$$

where α is a coefficient ranging from 0.9 to 1 (assuming that the mechanical power does not entirely dissipate in the form of heat because of hardening) and the effective viscosity μ depends on the strain rate through the Norton-Hoff law:

$$\mu = K(T)(2\mathrm{tr}(\dot{\varepsilon}\dot{\varepsilon}) + 3\gamma^2)^{\frac{m(T)-1}{2}}. \tag{8.37}$$

In the Norton-Hoff law the two coefficients K and m are functions of temperature, and play a role similar to that of yield stress (which depended on temperature) in the previous model. The finite element method is used to solve this system with some typical initial and boundary conditions. Figure 8.10 shows the distribution of temperature and two velocities. We encountered a problem related to the stability of the stress components.

The second simulation is based on the previous dynamical 1D plastic model. After some manipulations, the system is simplified to four equations in four unknowns: $(v_r, v_\theta, \Lambda, p)$.

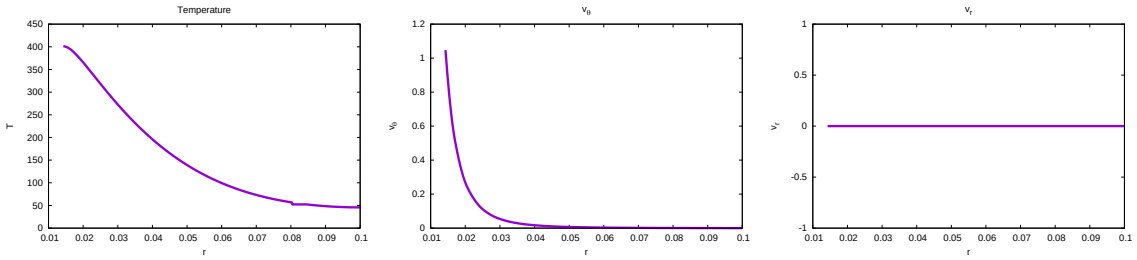


Fig. 8.10: Distribution of temperature and velocity predicted by the non-Newtonian fluid model.

$$
\begin{aligned}
&\rho\frac{\partial v_r}{\partial t}+\left(\rho v_r-\frac{\sigma_y}{\Lambda r}\right)\frac{\partial v_r}{\partial r}-\frac{\partial}{\partial r}\left(\frac{\sigma_y}{\Lambda}\frac{\partial v_r}{\partial r}\right)+\frac{\sigma_y}{\Lambda r^2}v_r=\frac{\rho}{r}v_\theta^2+\frac{\partial p}{\partial r}\\
&\rho\frac{\partial v_\theta}{\partial t}+\left(\rho v_r-\frac{\sigma_y}{\Lambda r}\right)\frac{\partial v_\theta}{\partial r}+\frac{\partial}{\partial r}\left(\frac{\sigma_y}{2\Lambda r}v_\theta-\frac{\sigma_y}{2\Lambda}\frac{\partial v_\theta}{\partial r}\right)+\left(\frac{\rho v_r}{r}+\frac{\sigma_y}{\Lambda r^3}\right)v_\theta=0\\
&\Lambda=\sqrt{\frac{3}{2}\left(2\left(\frac{v_r}{r}\right)^2+\frac{1}{2}\left(\frac{\partial v_\theta}{\partial r}-\frac{v_\theta}{r}\right)^2\right)}\\
&-\frac{\partial}{\partial r}\left(r\frac{\partial p}{\partial r}\right)=-\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\left(\frac{\sigma_y}{\Lambda r}v_r\right)+\frac{2\sigma_y}{\Lambda r}v_r+\rho r v_r\frac{\partial v_r}{\partial r}-\rho v_\theta^2\right)
\end{aligned}
\tag{8.38}
$$

The boundary conditions are as follows.

$$
\begin{aligned}
v_\theta=\gamma 2\pi\omega r_p,\quad \sigma_{rr}=P_r\quad &\text{at } r=r_p,\\
v_\theta=0,\quad v_r=0\quad &\text{at } r=r_b
\end{aligned}
\tag{8.39}
$$

The finite volume method was used to solve this system, but we encountered a problem regarding stability when solving the pressure equation.

8.6 Future Work and Discussion

In this report, we have investigated separately the heat generation and plastic deformation problems. The simulation of the temperature field and 1D analytical results provide some insights into the mechanism of the FSW process. Future work will address the coupling of the two aspects. For example, the effect of yielding material is not considered in the 1D temperature model, and we need to estimate the stress power $\mathrm{tr}(\boldsymbol{\sigma}\dot{\boldsymbol{\varepsilon}})$ based on the velocity in the plastic deformation. On the other hand, the dependence of yield stress on temperature is a crucial factor in the plastic-elastic deformation, and this should be taken into account to determine the location of the free boundary or the size of the yielding region.

In the plastic-elastic model, we have used a plane strain formulation, i.e., we assumed that everything is uniform along the thickness direction. This causes some inconsistency between yielding conditions and incompressibility conditions. It suggests that the component v_z or $\partial v_\theta/\partial z$ is important. A plate model will be needed to reduce the 3D problem to a 2D one, by keeping some information on these components.

Reference

1. Eric Feulvarch, J-C Roux, and J-M Bergheau. A simple and robust moving mesh technique for the finite element simulation of friction stir welding. *Journal of Computational and Applied Mathematics*, 246:269–277, 2013.
2. Kirk Fraser, Lyne St-Georges, and Laszlo I Kiss. A mesh-free solid-mechanics approach for simulating the friction stir-welding process. In *Joining Technologies*. InTech, 2016.
3. Peter Howell, Gregory Kozyreff, and John Ockendon. *Applied solid mechanics*, volume 43. Cambridge University Press, 2009.
4. Rafal Smerd. Constitutive behavior of aluminum alloy sheet at high strain rates. Master's thesis, University of Waterloo, 2005.