

# Semi-Autogenous Grinding Mill (SAG) Overload Forecasting Using Gram Penalized Matrices in a CNN

R. Hermosilla<sup>1</sup> <sup>a</sup>, H. Allende<sup>1</sup> <sup>b</sup> and C. Valle<sup>2</sup> <sup>c</sup>

<sup>1</sup>Universidad Técnica Federico Santa María, Valparaíso, Chile

<sup>2</sup>Universidad de Playa Ancha de Ciencias de la Educación, Chile

**Keywords:** Overload Forecasting, SAG Mill Overload, Multivariate Times-Series Forecasting, Image Encoding for Time-Series, Time-Series CNN, CNN Explanation.

**Abstract:** In mining, detecting overload conditions is an opportunity to perform SAG mill to optimal operating conditions. With this in mind, several authors have prove using machine learning mechanisms to detect overloads. Our proposal establishes and tests a series of techniques to detect and forecast these events. Finally, we will look for an explanation of what the model considers for classification improving the phenomenon knowledge. Inspired by previous work and how operators classify overloads by analyzing behavior graphs of the most relevant variables, we proposed a framework that includes selection, encoding, and filtering improvement to finally discover the importance of the characteristics observed by our model using explanation techniques. Thus, using a group of novelty techniques, our advances exceed the results presented by other authors and by ourselves in previous publications, opening the door to a model based on attention in the future.

## 1 INTRODUCTION

In the mining context, comminution is the process that reduces the ore size. In the case of most Chilean mining, to obtain smaller particles for the next step in the copper extraction (flotation).

Usually, this process includes crushing and grinding equipment in a circuit composed of several stages.

SAG mill (semi-autogenous grinding) can process large amounts of ore in the so-called primary grinding stage. Usually, this process includes crushing and grinding equipment in a circuit composed of several steps. The efficient use of SAG mills means an increase in ore production.

A 1% increase in SAG mill treatment supposes a profit increase of up to US\$160 MM per year and between 200 and 400 tons of CO<sub>2</sub> emissions saving under the same electricity consumption (Wakefield et al., 2018; Pontt et al., 2012; Northey et al., 2013).

However, achieving these efficiency levels has the risk of overload, a condition that limits the operation. Divers authors have tried to identify through several techniques such as the theoretical analysis


of variables (Strohmayr and Valery, 2001; Powell et al., 2009), use of predictive control ((Forbes and Gough, 2003)), making simulation models (Varas et al., 2019), through linear programming (César and Daniel, 2009) or using machine learning techniques (Ko and Shang, 2011; McClure and Gopaluni, 2015; Hermosilla et al., 2021).


In our research, we have established a mechanism that allows the classification and early detection (forecast) of overloads as a multivariate binary classification model.


Inspired by the work performed in the generation of encodings based on distance matrices (Bardinas et al., 2018) and the analysis that expert metallurgists to determine the occurrence of overloads, we have proposed a deep convolutional neural network that uses the feature relationship using a multivariate approach of Gram's matrix as an encoder.

In other words, we proposed the generation of matrices with angular differences of the feature pair arrays as input for our model. Those matrices are penalized with a temporary degradation filter that penalizes the oldest data, trying to establish a mechanism for differentiating the temporal data used in each encoded time window, which has yielded promising results.

Finally, we uncover, through explanation mechanisms, the combinations of variables/time that most

<sup>a</sup>  <https://orcid.org/0000-0002-6032-4811>

<sup>b</sup>  <https://orcid.org/0000-0002-9899-0051>

<sup>c</sup>  <https://orcid.org/0000-0001-7158-2069>

affect the prediction of overloads.

The following chapters explain the overload phenomenon and how we have approached a solution. Section 2 explains the overload phenomenon and its importance in the comminution process. Section 3 will delve into our proposal and how we address it through a set of proposed methods. Section 4 shows our results. Section 5 shows our conclusions and future work.

## 2 OVERLOAD DETECTION AND RELATED WORK

Maximizing ore treatment is one of the main goals of a SAG mill operation. Though, this objective depends on several factors that can influence the occurrence of an overload event.

Factors such as the distribution of solids, hardness, excessive load volume, particle size distribution, the number of steel balls, and wear, among others, can generate an overload condition.

### 2.1 Overload Types

In a SAG mill can found three kinds of overloads can occur (Fig. 1).

In all cases of overload, mill loses the *cataract*<sup>4</sup> property generating a vicious circle that leads to the ineffectiveness of the equipment. The volumetric overload occurs when the SAG mill is overfed. Ore's hardness, size, or composition can produce this condition.

When excess water attenuates the fall's effect, it is declared an overload by slurry pooling, affecting the ore's size reduction, producing accumulation and next, a volumetric overload. When the pulp covers the lifters, a freewheeling overload is declared. The pulp on the lifters avoids the ore rising, inhibiting the cascade.

Also, composition, distribution, and load fill can influence an overload occurrence. We can search this behavior by observing the power consumed but only at the beginning of an overload (Apelt et al., 2001), without enough foreseeing.

The distinct factors and the complex relations of the features underlying the overload make obtaining a physical forecast overload system hard. However, establishing a model based on data that learn this complex relationship is possible. Thus, our research focused on obtaining an overload forecast model based

<sup>4</sup>Ore rises to the top of the mill, breaking on the impact zone.

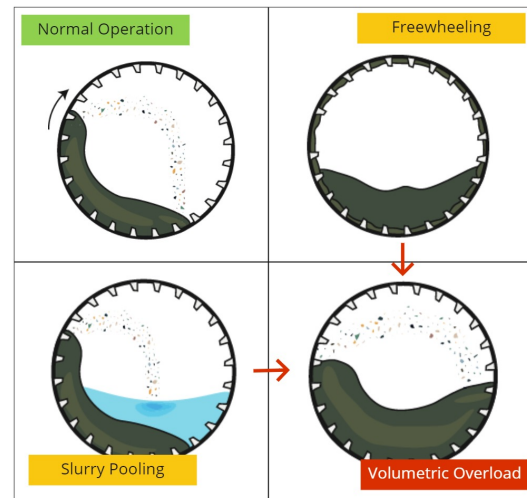


Figure 1: SAG mill overloads.

on a time-series classification with a multivariate approach.

### 2.2 Related Work

According to our research, two approaches exist at least to address the SAG mills' overloads and understand the problems associated with their operation.

One seeks to generate an adequate operation from the control point of view, and the other focuses on seeking the occurrence of overloads.

Among these investigations, ((Salazar et al., 2014)) establish a control and prediction mechanism for the behavior of the mill using a method called Multiple Input/Multiple Output Predictive Control Model (MIMO MPC).

The authors prove the relationship between the operating variables based on load tonnage.

His work also allowed the establishment of simulated operating conditions of an overload state, keeping the controlled variables without relevant variation.

On the other hand, (Wang et al., 2020) allowed to establish overload-free operating conditions through the control of the speed of the mill, generating an optimal operating space describing the relationship between speed, ore quantity, and particle size.

However, one of the nearest research to our line of study is published by (McClure and Gopaluni, 2015), which experiments with techniques like KPCS, SVM, and LLE to classify overloads (not predict them). Likewise, our previously published research advanced toward constructing an overload prediction framework using deep networks.

Now, we have improved results by better understanding the underlying phenomenon and the inclu-

sion of elements, such as the degradation or penalty filter.

In the previous work, we intuited could be explained the network in some way.

In this proposal, we have included a way to explain the network and analyze the results, allowing contrasting the model's decisions with experts' opinions.

### 3 OVERLOAD FORECASTING PROPOSED METHOD

As we have pointed out, we propose approaching our overload prediction problem through a multivariate binary classification model based on deep convolutional neural networks.

So, we have to overcome several difficulties, such as adequately selecting operation variables, generating a representative encode, or attending a highly unbalanced scenario (less than 2% of the labeled data corresponds to overloads).

Furthermore, we will add a filter that has given us promising results penalizing past data and the application of an interpretation method called Grad-CAM++ (Chattopadhyay et al., 2018) to determine and check the dependency of the selected relationships with our target.

#### 3.1 Base Definition

Let  $\mathbf{y} = [y_1, \dots, y_T]^T$  with  $T \in [1, \dots, T']$  as the set of values binaries that describe the overload condition every time  $t$  ( $y_t = 1$  as an overload at time  $t$ ). We will assume the matrix  $T' \times N$  as the matrix  $\mathbf{X} = [\mathbf{x}_1 | \dots | \mathbf{x}_{T'}]^T$  that expresses the collection of vectors  $\mathbf{x}_t$  formed by the  $N$  available and selected features. That is,  $x_t^{(j)}$  denotes the value of the  $j$ -th characteristic at time  $t$ .

Expressed our approach as a multivariate time-series problem that symbolizes a stochastic process, we define our goal as overload forecasting  $y_{t+k}$ , thus, in  $k$  steps ahead of the current observation  $t$ . Given that the associated distribution of the underlying process is unknown, we assume our model will learn the relationship of values observed previously to forecast the overload occurrences. Then, we modeled the values' fit as  $p(y_{t+k}|x_t, \dots, x_{t-w-1})$  in  $w$  steps to the past.

#### 3.2 Feature Selection by Pairs

We propose to train a deep convolution neural network to learn the relationship of the available features

through a structure that includes pairing the variables with the highest relationship with the overload.

Inspired by the classification of overloads achieved by metallurgical experts, we intend to select those pairs that provide the most information regarding our target.

We solved this using Conditional Mutual Information (CMI) (Liang et al., 2019) to get a reduced number of pairwise relations, looking for pairs related to adding the maximum information possible about our target. We express CMI as follows:

$$I(x^{(i)}, x^{(j)}|y) = E_y[D_{kl}(P_{(x^{(i)}, x^{(j)}|y)} || P_{(x^{(i)}|y)} \otimes P_{(x^{(j)}|y)})] \quad (1)$$

where  $I(x^{(i)}, x^{(j)}|y)$  is the value expected respect to  $y$ , of the *relative entropy* or *Kullback-Leibler divergence*  $D_{kl}$  from conditional joint distribution  $P_{(x^{(i)}, x^{(j)}|y)}$  to the product  $\otimes$  of conditional marginals  $P_{(x^{(i)}|y)}$  and  $P_{(x^{(j)}|y)}$ .

Accordingly, let define  $\mathbf{z}_r = (z_r^{(1)}, z_r^{(2)})$  as a matrix with features pairs  $x^{(i)}$  and  $x^{(j)}$  selected from  $r$ th better CMI values (1).

Let  $\mathbf{z}_r = (z_r^{(1)}, z_r^{(2)})$  as a matrix with features pairs  $x^{(i)}$  and  $x^{(j)}$  selected from  $r$ th better CMI values (1).

Let  $\tilde{\mathbf{Z}}$  as an array with three dimensions  $w \times w \times R$ , with window length  $w$  and the number of selected pairs  $R$ , each composed by  $\tilde{z}_{tr}^{(i)}$ .

#### 3.3 Encoder by Matrices Inspired in Gram

Problems such as univariate time series have recently been solved using gram matrices (Wang et al., 2022); however, we will use it as a multivariate approach applying some changes. After selecting  $R$  pairs on top of CMI ranking time series  $\mathbf{X}$  using the technique *min-max scaler*, we transform each pair  $\mathbf{z}_1, \dots, \mathbf{z}_R$  to the space  $[-1, 1]$  as follow:

$$\tilde{z}_{tr}^{(i)} = \frac{z_{tr}^{(i)} - \min(\mathbf{z}_r^{(i)})}{\max(\mathbf{z}_r^{(i)}) - \min(\mathbf{z}_r^{(i)})}, i = 1, 2, t = 1, \dots, T, r = 1, \dots, R, \quad (2)$$

where minimum value  $\min(\mathbf{z}_r^{(i)})$  and maximum value  $\max(\mathbf{z}_r^{(i)})$  of the  $i$ th selected feature.

Then, we transform  $\tilde{z}_{tr}^{(i)}$  to polar coordinates through the cosine of the angle of the values, and obtaining the radius  $\rho$  using the timestamp  $ts_t$ , given by:

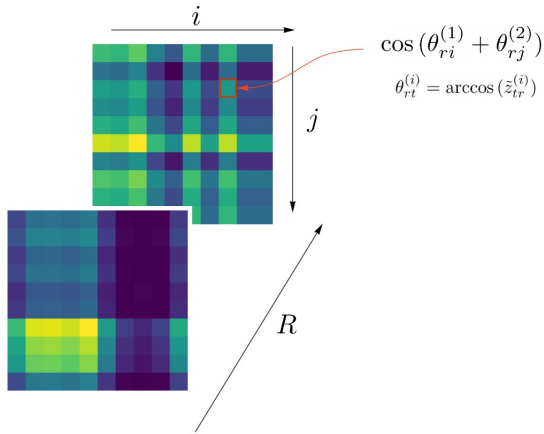


Figure 2: Image generated with GADF approach with  $R$  pairs/channels.

$$\theta_{rt}^{(i)} = \arccos(z_{tr}^{(i)}), -1 \leq z_{tr}^{(i)} \leq 1, \quad (3)$$

$$\rho_t = \frac{ts_t}{\max(ts)}, ts_t \in ts,$$

where the value of the timestamp is represented by  $ts_t$ , and the maximum value of all timestamps is wrote as  $\max(ts)$ .

The next step in our process is to apply a Gramian approach to the angle differences of each pair to obtain a three-dimensional array  $G$  with  $w \times w \times R$  dimensions. Here,  $g_{r(i,j)}$  represents each pixel in position  $(i, j)$  of  $r$ th Gram's matrix of  $G$ , as follows:

$$g_{r(i,j)} = \cos(\theta_{ri}^{(1)} + \theta_{rj}^{(2)}), i, j = 0, \dots, w - 1, \quad (4)$$

where  $\theta_{ri}^{(1)}$  and  $\theta_{rj}^{(2)}$  represents the transformed pair selected previously. We can understand  $G$  as the rendering of an image with  $R$  channels, where each image represents the angular variation of the top- $R$  CMI's ranking selected pairs (Figure 2).

### 3.4 Penalization over Time Sub-Matrices

As we can guess, overloads are a phenomenon that increases as we get closer to the moment of their appearance. Also, after making our models, we verified, using an explanation method (4.2), that in the cases in which overloads occur, our model considered the instants prior to the evaluated time to be more critical (Figure 3). Based on these antecedents, we define a filter that allows us to penalize those previous values in each channel of the encoder.

Let  $f = [0, \dots, w - 1]$  define the filter  $F = \frac{f+f^T}{2w}$ , which will generate a filter of dimension  $w \times w$  with

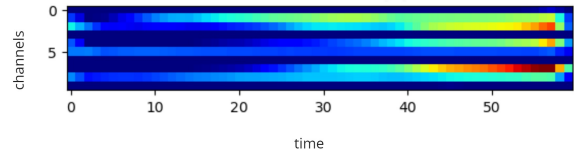


Figure 3: Importance by selected pairs over overload classification. Blue and red colors represent low and high importance, respectively.

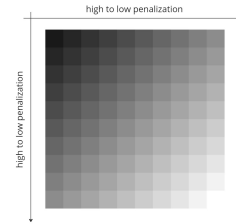


Figure 4: Filter applied in each channel.

values in  $\mathbb{R}$  in the interval  $[0, 1]$  which we will apply to each of the channels  $R$  of the generated images  $G$  (Figure 4). As we will see in chapter 4, this simple method allowed us to improve our model significantly.

### 3.5 Convolutional Neural Network

The set of images  $G$  are taken by a deep convolutional neural network (Fig. 5), made up of convolutional layers with  $n_1$  filters of  $k_1 \times k_1$  size and  $2 \times 2$  max-pooling, followed by a batch normalization layer.

We added once more time the same set of convolutional, max-pooling and batch normalization layers (with the same hyperparameters). All the convolutional layers have  $ReLU$  as an activation function and zero padding.

Finally, the last layer is fully connected with a flatter layer, followed by  $n_3$  nodes with  $Dropout$  dense layer, to end with one single neuron as output for binary classification using a *sigmoidal* activation function. Due to the highly imbalanced scenario found,

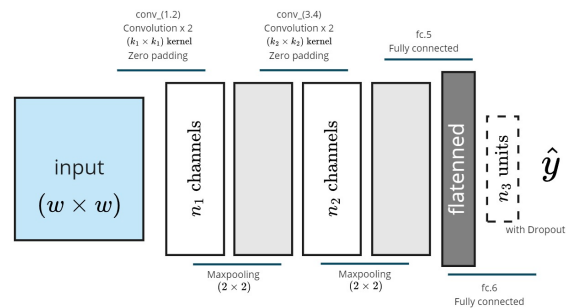


Figure 5: Convolutional Neural Network Architecture.

we use *focal binary crossentropy* as a cost function setting with class weights parameters.

## 4 EXPERIMENTS AND RESULTS

Obtaining a forecast with ten minutes of anticipation is our primary goal. Considering a 30 seconds resolution, we set  $k = 20$  in our target  $y_t + k$ . Also, we define  $w = 60$  as the offset into the past, ergo 30 minutes.

We have a dataset with around 700k labeled records strongly imbalanced (1.46% imbalanced ratio), corresponding to nearly twelve months of data.

We selected 48 of 72 features linearly independent. With these 48 features we start our process of selecting the  $K = 10$  top pairs corresponding to  $\{(fin, gru), (pot, dm_-), (ton, dm_-), (fin, int), (gru, int), (pre, vel), (pot, vel), (pot, pre), (pre, dm_-), (int, pre)\}$ <sup>5</sup>.

After selecting our top feature pairs, we build the  $G$  matrices using our Gram's matrices approach. Due to time-missing values, this process reduced the number of matrices to around 685k, reducing our imbalanced ratio to 1.38%.

We setted the network hyperparameters with  $n_1 = 256$ ,  $n_2 = 128$ ,  $n_3 = 512$  nodes for each layer using the grid search method. Using the same way we setted  $(k_1 \times k_1) = (k_2 \times k_2) = (3 \times 3)$  for the kernels size. We setted the optimizer as *Adam* with learning rate ( $lr = 0.001$ ).

To train we use a cross-validation technique proposed by (Bergmeir and Benítez, 2012) for time series.

### 4.1 Evaluation Methods

Table 1 shows the average sensitivity, specificity, and F1 score with the respective standard deviation to the model based on LLE+LDA as suggested by (McClure and Gopaluni, 2015), we have also included the results of our previously proposed model (Hermosilla et al., 2021), and the results of this proposal, without filter ( $\star$ ) and with penalization filter. As can be seen, the proposed model with the applied filter exceeded the previous results significantly.

### 4.2 GradCAM++ CNN Explanation

The authors (Chattopadhyay et al., 2018) proposed a generalized method called *Grad-CAM++* that is formulated by explicitly modeling the contribution of

<sup>5</sup>fin: fine granulometry, gru: coarse granulometry, dm\_-: milliseconds from the last maintenance, int: intermediate granulometry, pre: pressure, vel: velocity, ton: total tonnage.

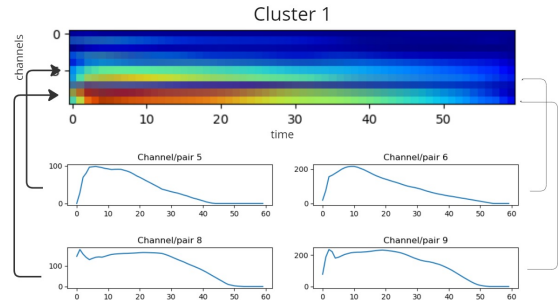


Figure 6: Example of cluster generated with a explanation model based on GradCAM++.

each pixel in the feature maps of a CNN to the final output. With this, it is possible to look at the elements to which the network pays attention in each case. To do this, they reformulate the structure of the  $w_c^k$  weight for  $c$  class in  $k$  layer of the network:

$$w_c^k = \sum_i \sum_j \alpha_{ij}^{kc} \cdot \text{relu} \left( \frac{\partial Y^c}{\partial A_{ij}^k} \right), \quad (5)$$

where *relu* is the Rectified Linear Unit activation function,  $\alpha_{ij}^{kc}$  are the weights of the gradients for a particular pixel  $i, j$ , and  $Y^c$  are the values of the particular class to analyze.  $w_c^k$  will capture the importance of the activation map  $A^k$ .

Using this method, we characterized the overloads to understand feature pairs directly influencing the target. Have been clustered the results to achieve an analysis of the leading classification groups of the model.

In Figure 6, we can see an example of how the model considers the pairs of channels 5, 6, 8, and 9 most relevant in the periods close to the overload occurrence.

Precisely channels 5 ( $\{pre\}, \{vel\}$ ) and 6 ( $\{pot\}, \{vel\}$ ) are relationships, which experts use to classify overloads. However, channels 8 and 9, with pairs  $(pre, dm_-), (int, pre)$ , seem particularly relevant to the model and offer an opportunity to use new relations to the study of overload genesis.

An interesting observation is that even having penalized the times far from the moment  $t$ , in all cases, the model highlights those instants as essential for the classification, which makes us suppose that the information contained in the first moments of the characteristics matrix  $G$  is essential for forecasting.

Table 1: Average and standard deviation over  $S$  test sets.

Model	Sensitivity	Specificity	F1
LLE+LDA	0.555( $\pm$ 0.425)	0.453( $\pm$ 0.321)	0.440( $\pm$ 0.298)
Previous proposed model	0.643( $\pm$ 0.227)	0.894( $\pm$ 0.106)	0.609( $\pm$ 0.084)
Proposed model*	0.653( $\pm$ 0.124)	0.889( $\pm$ 0.178)	0.742( $\pm$ 0.098)
Proposed model	<b>0.871(<math>\pm</math>0.134)</b>	0.838( $\pm$ 0.102)	<b>0.859(<math>\pm</math>0.113)</b>

## 5 CONCLUSIONS AND FUTURE WORKS

In this work, we propose improving the methods to arrive at an overload forecasting model in a complex, multivariate, and highly unbalanced problem using a Gram matrix-based encoding.

We take advantage of the benefits of the CNNs to generate a model that allows us to know the relationships of these matrices with the overload.

The experimental results show that we have overcome the approach in previous works and state of the art.

Using an explanation method called Grad-CAM++, we established some interesting study sets for expert review, for example, the relationship between pressure and timing of equipment maintenance and fine grain size and pressure to explain some overloads.

Also, this behavior could allow us to increase the forecast distance. In the future, we will integrate the care information in the same network to generate a model specialized mainly in those elements that most influence the occurrence of overloads.

## REFERENCES

- Apelt, T., Asprey, S., and Thornhill, N. (2001). Inferential measurement of sag mill parameters. *Minerals engineering*, 14(6):575–591.
- Bardinas, J., Aldrich, C., and Napier, L. (2018). Predicting the operating states of grinding circuits by use of recurrence texture analysis of time series data. *Processes*, 6(2):17.
- Bergmeir, C. and Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213.
- César, G. Q. and Daniel, S. H. (2009). Multivariable model predictive control of a simulated sag plant. *IFAC Proceedings Volumes*, 42(23):37–42.
- Chattopadhyay, A., Sarkar, A., Howlader, P., and Balasubramanian, V. N. (2018). Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE.
- Forbes, M. G. and Gough, B. (2003). Model predictive control of sag mills and flotation circuits.
- Hermosilla, R., Valle, C., Allende, H., Lucic, E., and Espinoza, P. (2021). Semi-autogenous (sag) mill overload forecasting. In *Iberoamerican Congress on Pattern Recognition*, pages 392–401. Springer.
- Ko, Y.-D. and Shang, H. (2011). Sag mill system diagnosis using multivariate process variable analysis. *The Canadian Journal of Chemical Engineering*, 89(6):1492–1501.
- Liang, J., Hou, L., Luan, Z., and Huang, W. (2019). Feature selection with conditional mutual information considering feature interaction. *Symmetry*, 11(7):858.
- McClure, K. and Gopaluni, R. (2015). Overload detection in semi-autogenous grinding: A nonlinear process monitoring approach. *IFAC-PapersOnLine*, 48(8):960–965.
- Northey, S., Haque, N., and Mudd, G. (2013). Using sustainability reporting to assess the environmental footprint of copper mining. *Journal of Cleaner Production*, 40:118–128. Special Volume: Sustainable consumption and production for Asia: Sustainability through green design and practice.
- Pontt, J., Valderrama, W., Olivares, M., Rojas, F., Robles, H., L’Huissiers, S., and Leiva, F. (2012). Uso eficiente de la energía en procesos mineros. *Centro de automatización para la industria minera, Chile*.
- Powell, M., Van der Westhuizen, A., and Mainza, A. (2009). Applying grindcurves to mill operation and optimisation. *Minerals Engineering*, 22(7-8):625–632.
- Salazar, J.-L., Valdés-González, H., Vyhmesiter, E., and Cubillos, F. (2014). Model predictive control of semi-autogenous mills (sag). *Minerals Engineering*, 64:92–96.
- Strohmayr, S. and Valery, W. (2001). Sag mill circuit optimisation at ernest henry mining. In *Proceedings of the SAG 2001 Conference, Vancouver, BC, Canada*, volume 30.
- Varas, P., Carvajal, R., and Agüero, J. C. (2019). State estimation for sag mills utilizing a simplified model with an alternative measurement. In *2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, pages 1–7. IEEE.
- Wakefield, B., Lindner, B., McCoy, J., and Auret, L. (2018). Monitoring of a simulated milling circuit: Fault diagnosis and economic impact. *Minerals Engineering*, 120:132–151.
- Wang, J., Li, S., Ji, W., Jiang, T., and Song, B. (2022). A t-cnn time series classification method based on gram matrix. *Scientific Reports*, 12(1):1–14.
- Wang, X., Yi, J., Zhou, Z., and Yang, C. (2020). Optimal speed control for a semi-autogenous mill based on discrete element method. *Processes*, 8(2):233.