

# On-Line Optimization of Cone Crushers using Extremum-Seeking Control

Khalid Tourkey Atta\*, Andreas Johansson, and Thomas Gustafsson

**Abstract**—This article demonstrates the ability of on-line optimization of cone crushers, specifically maximization of the total throughput of the crusher by adjusting the eccentric speed ( $\omega$ ). The on-line optimization was based on the Extremum-Seeking Control (ESC) approach, which is advantageous when optimizing systems with unknown time varying characteristics. Two types of gradient based approaches are tested in simulation, the traditional Band-pass filters method and a method utilizing the Extended Kalman Filter (EKF). Both methods perform satisfactory, demonstrating the good potential of ESC for online-optimization of cone crushers. To deal with unwanted behavior of the EKF based approach for situations when the gradient is not correctly estimated, a modification is suggested based on detecting this condition and accommodating for it.

## I. INTRODUCTION

Size reduction (comminution) is a vital part in any mineral processing plant and the cone crusher is widely used for this purpose. And as any size-reduction process, most of the energy consumed will appear as sound and heat, and a small amount will be used to produce smaller particles [1].

The objectives of crusher optimization depend on whether the crusher produces the end product or is part of a comminution chain. One example of optimization criteria, to be considered here, is throughput maximization, in case the crushing is the bottle neck in the total plant production.

As pointed out in [2], until the late 90's the dependence of the capacity and the product size distribution on the operating speed was not fully appreciated, possibly because of the practical difficulties associated with the required experiments.

Due to this lack of quantitative models for the effect of the main manipulated variables (Eccentric Speed ( $\omega$ ) and Closed side Setting (CSS)), on-line optimization of crushers by Closed-Loop control has only recently begun to attract interest. Recent modeling efforts by Evertsson and co-workers [3], [4], [5] shows that these variables have a significant influence on the throughput which is also confirmed in experimental studies (e.g.[6]).

The Pioneering work of on-line optimization of cone crushers is [2], where a finite state machine is presented to maximize the output by producing small positive and negative steps in speed. The difficulties that arise when attempting to use  $\omega$  to maximize the output are related to limitations in the range and the rate of change of  $\omega$ . The slow step response makes it difficult to speed up the process

of tracking the optimal values, which may change quickly with the properties of the input material, apart from changes due to wear.

Extremum Seeking Control (ESC) is a control concept for single objective on-line optimization. Many different applications for ESC can be found in the literature e.g. internal combustion engines, grinding processes, and solar cells [7]. In this article, we will apply two algorithms for ESC on a dynamic model of a cone crusher. To the best of our knowledge, ESC has not previously been considered for crusher control.

### A. Extremum Seeking Control Concept

Plant Optimization is the main objective of any control system. Most controllers have a certain input called Set-Point, which defines the required value of a variable in the plant. On the other hand, in any plant, there are some variables that measure the performance of the plant, e.g. the total production, the production of waste, the emission of gases (ex. CO<sub>2</sub>) and has no predefined set point, but is always required to be maximized or minimized.

Extremum Seeking Control (ESC) requires some prior knowledge of the dynamics, but not the exact conditions of optimality, neither location nor values, and these are also allowed to be varying in time, with unknown dynamics. Consider a plant with a non-linear and time-varying behavior defined as

$$\frac{dx}{dt} = f(x, u(t)) \quad (1a)$$

$$y = h(x) \quad (1b)$$

where  $x$  is a vector representing the state of the plant,  $u$  is a vector of manipulated (input) variables, and  $y$  is a scalar representing the output variable (performance) of the plant. It is assumed that the steady-state values of  $y$ , denoted  $y_{ss}$ , has a unique maximum with respect to a constant input  $u$ , denoted as  $u_{ss}$ , for any fixed  $t$ . In the following, we will consider  $u$  as a scalar and limit our scope to single input systems.

The extremum-seeking controller, as illustrated in Fig 1, will try to drive the system into its optimal value ( $y^*$ ) by finding the optimal value for  $u$  ( $u^*$ ), even though the values of  $y^*$  and  $u^*$  are unknown and varying with time.

Many ESC methods have been suggested, and can be categorized in many different manners, e.g. Static, Analog, and Numerical Methods [8], or perturbation and non-perturbation methods. Some methods only require the knowledge of the output of the system ( $y$ ), like most of the perturbation methods, while some other methods require the state of the

Authors are with the Control Engineering Group, Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, 971 87 Luleå, Sweden

\*Corresponding author. Tel: +46 (0)920 49 2567, Fax:+46 (0)920 49 2801, Email: Khalid.Atta@Ltu.se

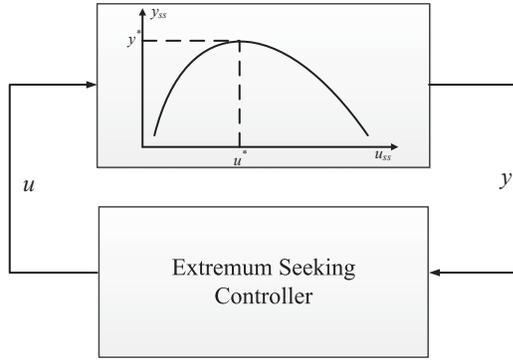


Fig. 1. Principle of Extremum-Seeking Control

system ( $x$ ). In most cases, this requires using an observer, which adds more complexity to the controller. In this work we will adopt the perturbation based approach.

The idea behind the perturbation based approach is to add a small signal (sometimes called dither signal) to a base input ( $u_0$ ), and then measure the change in the output ( $y$ ) with respect to this signal. Based on this change, it will be decided to increase or decrease the input signal ( $u_0$ ). The Gradient based methods, inspired by the steepest descent optimization method, are based on of this approach, and rely on using the gradient of the output with respect to the input in order to decide how to change the base variable  $u_0$ . The ESC controller will consequently be divided into two parts, the Gradient estimator, to find the rate of change of  $y$  with respect to  $u$  (i.e.  $\partial y/\partial u$ ), and an integrator to accumulate the values of the gradient to give the base variable  $u_0$ . A basic gradient estimator is implemented by using a sinusoidal wave as a perturbation signal, and a High-Pass filters (HPF) and a Low-Pass filters (LPF), as shown in the Fig. 2. After the pioneer work of Krstić and the proof of closed-loop stability of this approach [9], this kind of controllers have become popular.

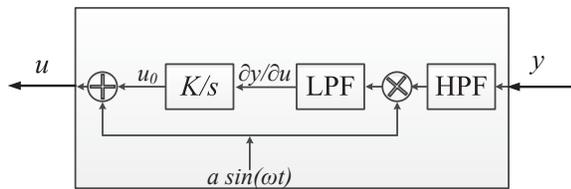


Fig. 2. Gradient Based ESC, using Filters to find the gradient

The design of such a controller, involves the selection of the frequency and the amplitude of the perturbation signal, the corner frequencies of the HPF and LPF, and the gain of the integrator ( $K$ ). The crucial variable is the perturbation frequency which should be selected slower than the dynamics of the plant, i.e. so that the output of the plant will track in the input.

Another approach for gradient estimation is to use an Extended Kalman Filter (EKF) [10]. The Method was sug-

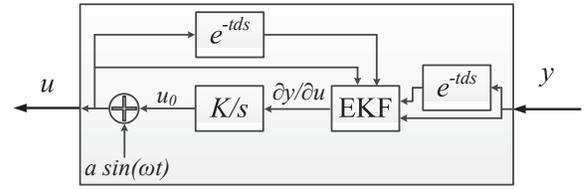


Fig. 3. Gradient Based ESC, using Extended Kalman Filter to find the gradient

gested originally for static systems, but can also be applied to dynamic systems. The idea behind using the EKF as a gradient estimator is to approximate the output of the system ( $y$ ) by a tangent at the point of operation. i.e.

$$y = y_0 + ku \quad (2)$$

where  $k$  is the slope of the tangent. The EKF is then employed to estimate two state variables,  $x_1 = k$  and  $x_2 = y_0$ . Knowing that the EKF will be implemented in discrete time, two samples are required to have observability [10]. The discrete state space system will be described by

$$x(t_{k+1}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t_k) + w_k \quad (3)$$

$$\begin{bmatrix} y(t_k) \\ y(t_{k-n}) \end{bmatrix} = \begin{bmatrix} u(t_k) & 1 \\ u(t_{k-n}) & 1 \end{bmatrix} x(t_k) + v_k \quad (4)$$

where  $n$  is the time interval between the two samples which is usually selected to equal a quarter or three quarters of the cycle time of the perturbation signal (i.e.  $\pi/(2\omega)$  or  $3\pi/(2\omega)$ ). The difficulties in designing this type of controllers may be less than for the previous type, but the selection of  $\omega$  remains crucial.

## II. CONE CRUSHERS MODELING

The cone crusher, as shown in Fig. 4, consist of a concave bowl, providing a space called the Chamber, for the material to be crushed by a large concave metal piece, called Mantle. The mantle is rotating in an eccentric path to deliver several strokes to the material passing the chamber, which provides the required breakage until it reaches the size that allows it to pass through the controlled small opening in the end (down) of the chamber.

Cone crusher modeling has been subject of much research effort [5], [12]. First principles models may assume one stage of operations or several stages (multi-zone) or even be continuous involving a partial integro-differential equation [11]. Others use an empirical approach based on mixing Linear and Non-Linear blocks. In this work we use a simplified version of a model from [13], where the Crusher is divided into zones as shown in Fig. 5.

In this model, the material size distributions are represented as vectors and the operations as matrix multiplications, which is also known as matrix analysis approach [14]. Accordingly, the material are partitioned into size classes  $\infty > D_1 > D_2 \dots > D_m > 0$ , and the mass (or the mass flow) size distribution are vectors on the form  $M =$

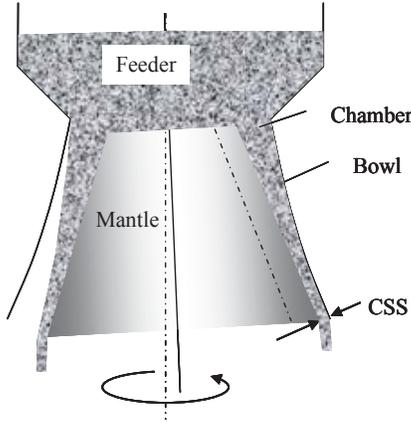


Fig. 4. Cross Section of Cone Crusher [11]

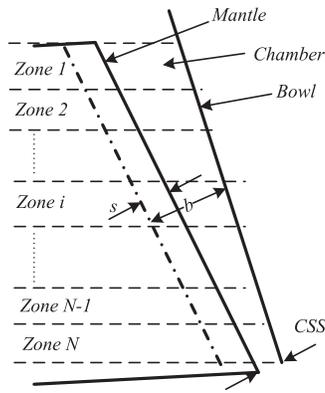


Fig. 5. Cross section showing the zones, CSS, stroke ( $s$ ), and bed height ( $b$ )

$[M_1 \ M_2 \ \dots \ M_m]$ , where  $M_j$  represents a mass (or a mass flow) of particles in the size interval  $[D_{j-1}, D_j]$ .

The Block diagram of the operation in Zone  $i$  is shown in Fig. 6. Here  $U_i$  is a vector representing the input to the zone, i.e. the mass size distribution per cycle (stroke) accepted in the  $i$ th zone. Similarly  $Z_i$  represents the output mass size distribution per cycle of Zone  $i$ , and is at the same time input to Zone  $i+1$  (i.e.  $U_{i+1} = Z_i$ ).  $X_i$  represents the mass size distribution inside the  $i$ th zone while  $F_i$  represents the material that is retained in the zone (i.e. did not pass down), after applying the actions of selection and breakage on it. The input of the first zone ( $U_1$ ) is the material fed into the crusher, and the Output of the last zone ( $Z_N$ ) is the output of the crusher.

The Breakage Matrix  $B_i$  of Zone  $i$  is a lower triangular matrix,

$$B_i = \begin{bmatrix} b_i^{1,1} & 0 & \dots & 0 \\ b_i^{2,1} & b_i^{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ b_i^{m,1} & b_i^{m,2} & \dots & b_i^{m,m} \end{bmatrix} \quad (5)$$

where the entry  $b_i^{k,j}$  represents the mass proportion of particles from size class  $k$  that ends up in size class  $j$ , after

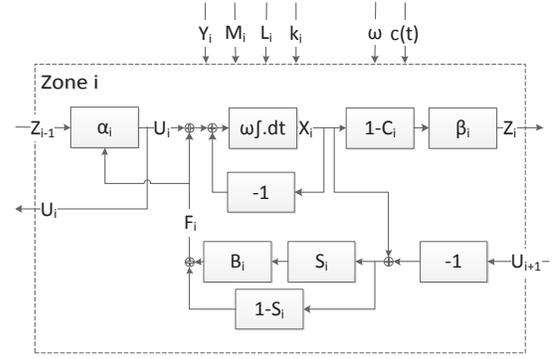


Fig. 6. Dynamic Model of Zone  $i$  [13]

breakage in Zone  $i$  [14]. The Selection Matrix  $S$  is a diagonal matrix,

$$S_i = \text{diag} \left( [s_i^1, s_i^2, \dots, s_i^m] \right) \quad (6)$$

where each entry  $s_i^j$  represents the probability of breakage of a particle in the  $j$ th size class in Zone  $i$ .

Finally, the Classification Matrix  $C$  is a diagonal matrix ,

$$C_i = \text{diag} \left( [c_i^1, c_i^2, \dots, c_i^m] \right) \quad (7)$$

where each entry  $c_i^j$  represents the fraction (proportion) of material from the  $j$ th size class in Zone  $i$  that is prevented from moving to the next zone.

The zone capacity factor ( $\alpha_i \leq 1$ ) is a linear action on all the material  $Z_i$  that is qualified to pass to the next zone and determines the percentage of material to flow into the zone in order not to exceed its maximum capacity ( $M_i$ ).

The Speed-Zone Length factor  $\beta_i = \frac{g\eta^2}{2\omega^2 L_i}$  represents the fraction of material that passes down out of the material that is classified to pass down because the length of the zone ( $L_i$ ) is too large for all material to have time to pass. The material has  $\eta T$  seconds to fall down, where  $T = \frac{1}{\omega}$  is the cycle time in seconds, and  $\eta$  is a factor determining the effective nominal time for the material to fall down which is assumed to average all three modes of transportation down (falling, sliding, or squeezing) [4].

In conclusion the adopted model is governed by the following set of equations:

$$Z_i = \beta_i(1 - C_i)X_i \quad (8a)$$

$$U_i = \alpha_i Z_{i-1} \quad (8b)$$

$$\frac{dX_i}{dt} = \omega \left( -X_i + U_i + F_i \right) \quad (8c)$$

$$F_i = (B_i S_i + (I - S_i)) (X_i - U_{i+1}) \quad (8d)$$

$$\alpha_i = \min \left( \frac{M_i - \mathbf{1}_m F_i}{\mathbf{1}_m Z_{i-1}}, 1 \right) \quad (8e)$$

$$\beta_i = \frac{g \eta^2}{2 \omega^2 L_i} \quad (8f)$$

For the last zone, there is no successive zone, so  $U_{N+1}$  will be replaced by  $Z_N$ , and for the first zone, there is no previous zone, so  $Z_{i-1}$  will be replaced by the size

distribution vector of the inflow to the crusher.

### A. Modeling Assumptions

In this section we will present the assumptions used for the simulations in the next section. The Breakage Function is from King [15]

$$B(x,y) = K \left(\frac{x}{y}\right)^{n_1} + (1-K) \left(\frac{x}{y}\right)^{n_2} \quad (9)$$

and the parameters were selected as  $n_1 = 10$ ,  $n_2 = 8$ , and  $K = 0.1$ , knowing that larger values for  $n_1$  and  $n_2$  corresponds to harder materials. From this the elements of the  $B$  matrix are given as  $b_{i,j} = B(D_{i-1}, D_j) - B(D_i, D_j)$ .

The classification function is also from [15] and is, as suggested in [11], applied at each level in the crusher. Thus

$$C(t,y,D_j) = \begin{cases} c_0(t,y,D_j), & d_1 \leq D_j < d_2 \\ 0, & D_j < d_1 \\ 1, & D_j \geq d_2 \end{cases} \quad (10)$$

where

$$c_0(t,y,D_j) = 1 - \left( \frac{D_j - d_2(t,y)}{d_1(t,y) - d_2(t,y)} \right)^n \quad (11)$$

$$d_1(t,y) = \alpha_1 l(c(t),y) \quad (12)$$

$$d_2(t,y) = \alpha_2 l(c(t),y) + d^* \quad (13)$$

$$l(c(t),y) = l_0(y) + c(t) \quad (14)$$

and where  $l_0(y)$  is the crusher chamber profile when CSS equals zero. We select a simple linear profile  $l_0(y) = (1 - y/Y) * 0.1m$ . Here  $Y$  is the length of the crusher chamber, and was selected to equal  $1.2m$ , and  $c(t)$  is the CSS. The other parameters were chosen to be  $\alpha_1 = 0.6$ ,  $\alpha_2 = 2$ ,  $n = 2$ , and  $d^* = 0$ .

The selection function is from [16], modified to take into account the effect of the compression ratio as suggested by Evertsson [5]. Thus

$$s(D_j,t,y) = s_0(\sigma(t,y)) D_j^a \quad (15)$$

where [17]

$$s_0(\sigma) = s_{ini} * (-3.086\sigma^2 + 3.5508\sigma - 0.0082) \quad (16)$$

The parameter  $s_{ini}$  was selected to equal 0.1 and  $\sigma(t,y)$  is the compression ratio  $\left(\frac{\text{stroke}}{\text{bedheight}} \text{ or } \frac{s}{b}\right)$  which depends on the CSS and the chamber profile as

$$\sigma(t,y) = \frac{k(y)}{k(y) + l_0(y) + c(t)} \quad (17)$$

The stroke  $k(y)$  was assumed to be independent of  $y$ , and chosen to equal  $10 \text{ mm}$ .

The number of size classes was selected to be 24 ( $m = 24$ ), with the largest size class equal to  $161.27 \text{ mm}$  and the ratio between consecutive classes taken to be  $(1/\sqrt[3]{2})$ . A truncated Rosin-Rammler distribution of the feed material was assumed. The Size Distribution function is then [15]

$$P(D) = \begin{cases} 1 - e^{-(\eta/\eta_{63})^\psi}, & D \leq D_0 \\ 1, & D > D_0 \end{cases} \quad (18)$$

where  $\eta = D/(D_0 - D)$  and  $\eta_{63} = D_{63}/(D_0 - D_{63})$ . The parameters  $D_{63}$  and  $\psi$  determine the shape of the size distribution of the fed materials and are in general time varying.  $\psi$  was selected to be constant and equal to 1.2. The Crusher was simulated using Matlab Simulink. The number of zones was selected to  $N = 10$ , with equal length  $L_i = \frac{Y}{N}$  and the mass capacity of each zone was selected to be linearly spaced from  $20 \text{ kg}$  in the first zone, and  $10 \text{ kg}$  in the last zone.  $\eta$  was selected to equal 0.5. CSS was assumed to vary between  $10$  and  $40 \text{ mm}$ , while the eccentric speed  $\omega$  was varied between  $3.5$  and  $20 \text{ rps}$ . A  $32 \text{ mm}$  sieve (screen) was simulated to filter the output of the crusher, the rejected large material was returned to the input of the crusher.

### III. APPLICATION OF ESC ON CONE CRUSHERS

The main point in ESC, as explained earlier, is that the steady-state relation between the input(s) and the objective index (output) should have a unique maximum (or minimum).

Simulations of the steady-state behavior of the model in section II is demonstrated in the form of a Crusher Performance Map (CPM) [3], which presents the steady state relation between the control variables (CSS and  $\omega$ ) and the Total and sub-sizes of the total production of the crusher as shown in Fig 7. The figure shows that the model satisfies this condition, both for the total throughput and for individual size classes.

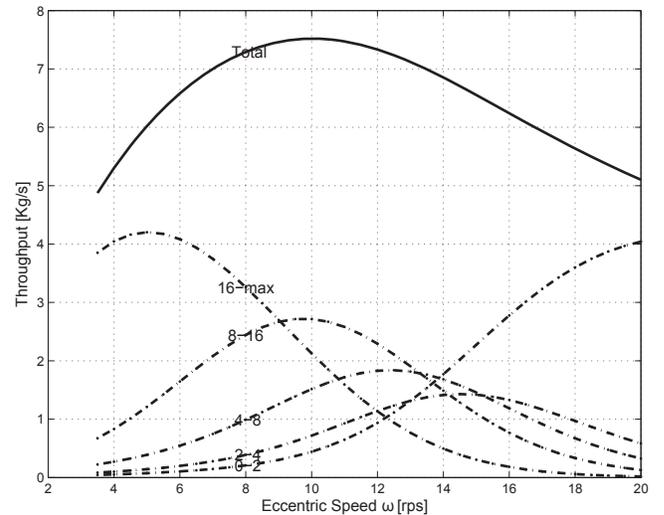


Fig. 7. CPM for Total and Sub-Classes Throughput versus Eccentric Speed [13]

The set of equations (8) can be rewritten in order to eliminate non-state variables (i.e  $F_i$ ,  $U_i$ , and  $Z_i$ )

$$\frac{dX_i}{dt} = \omega \left( -X_i + \alpha_i \beta_{i-1} (1 - C_{i-1}) X_{i-1} + (B_i S_i + (I - S_i)) (X_i - \alpha_{i+1} \beta_i (1 - C_i) X_i) \right)$$

$$\alpha_i = \min \left( \frac{M_i}{\mathbf{1}_m \beta_{i-1} (1 - C_{i-1}) X_{i-1}} - \frac{\mathbf{1}_m (B_i S_i + (I - S_i)) (X_i - \alpha_{i+1} \beta_i (1 - C_i) X_i)}{\mathbf{1}_m \beta_{i-1} (1 - C_{i-1}) X_{i-1}}, 1 \right)$$

$$\beta_i = \frac{g \eta^2}{2 \omega^2 L_i}$$

$$y = \mathbf{1}_m X_N$$

In general, the classification matrix ( $C$ ), and the selection matrix ( $S$ ) will depend on CSS, which require the following more general equations to describe the behavior of the crusher:

$$\frac{dX}{dt} = f(X, \omega, CSS) \quad (19a)$$

$$y = h(X) \quad (19b)$$

In our case, CSS is considered to be controlled manually, and is regarded as a disturbance like other parameters that govern the operation of the crusher, input material size distribution, Type of input material (ex. hardness), Moisture, Crusher chamber profile, etc.

#### IV. SIMULATION OF ESC OF TOTAL PRODUCTION USING ECCENTRIC SPEED

In the simulations the crusher was operated in four modes, each with duration of 500 seconds. The parameters of each mode are shown Table I:

TABLE I  
MODES OF OPERATION

Mode	CSS [mm]	$D_{63}$ [mm]	Soft/Hard Materials
1	30	80	Soft
2	15	80	Soft
3	15	100	Hard
4	15	60	Hard

Each mode has his own CPM as shown in Fig. 8 and a corresponding optimal  $\omega$  and maximal output. The size distribution of the input material was additively disturbed with Gaussian distributed noise with zero mean and variance equal to 15% of the value of each size class. The controller was disabled during the first 125 seconds, in order to let the crusher reach a steady state.

##### A. Standard ESC: Band-Pass filters

The sinusoidal perturbation was selected to have frequency 0.2  $rad/s$ , and amplitude 0.4  $rps$ . The gain of the integrator ( $K$ ) is equal to 30. The LPF was limited with a saturation level, in order to prevent a fast change in the eccentric speed which will have no influence on the stability of the system.

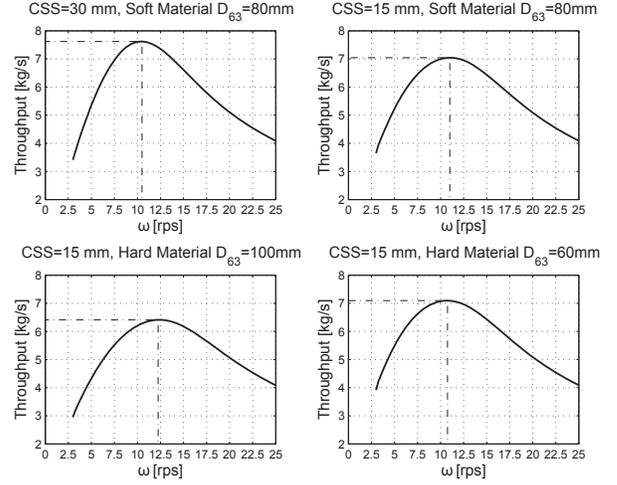


Fig. 8. CPM's for the four modes in Table I

The saturation level was set to  $\pm 5 * 10^{-3}$ . The filters were selected to be of second order with transfer functions:

$$G_{HPF} = \frac{s^2}{(0.08s + 1)^2} \quad (20)$$

$$G_{LPF} = \frac{0.04 * 0.08}{(s + 0.04)(s + 0.08)} \quad (21)$$

Fig. 9 shows the behavior of the controller in the form of a contour plot of the capacity as a function of  $\omega$  and time as the four modes in Table I are traversed. The capacity is normalized so that the same red color represents the highest production in each mode, and the light blue represents the lowest. Over this chart the eccentric speed was plotted to show how the controller brings the production of the crusher to its maximum. In the lower part, the dotted line presents the maximum throughput, and the thick line shows the actual throughput achieved by the ESC controller.

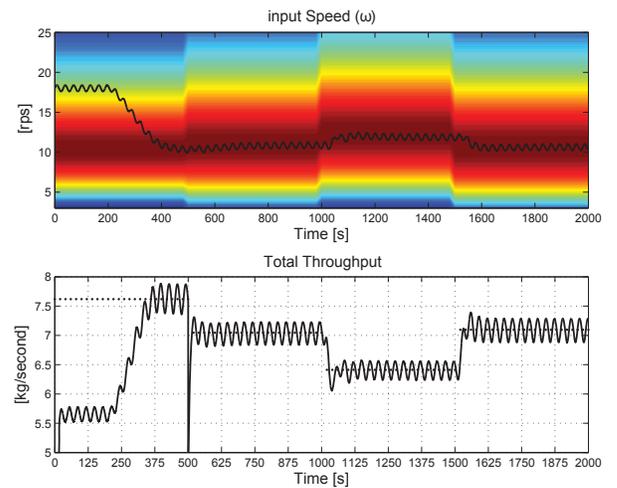


Fig. 9. Step Response of ESC with Band Filters

It is evident that the ESC controller works as expected and drives the crusher into its maximum production within ap-

proximately 100 seconds after an abrupt change of some operating variable.

### B. EKF Based ESC

For the EKF a constant step time ( $\Delta t$ ) of 0.05 second was chosen. Similar to [10], the EKF parameters were selected as:

$$Q = \Delta \bar{t} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (22)$$

where  $\Delta \bar{t} = \Delta t/s$ , and

$$R = 0.05 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (23)$$

where  $Q$  and  $R$  are the covariance matrix of the noise signals  $w_k$  and  $v_k$  respectively.

The sinusoidal perturbation was selected to have a frequency of 0.1 *rad/s*, and amplitude of 0.5 *rps*. The gain of the integrator ( $K$ ) was chosen to be 0.1. The input to the integrator was limited to  $\pm 1$  (see above). The time delay was selected to equal one quarter of the cycle time of the perturbation frequency.

Note that the perturbation frequency was selected to be low, because the algorithm was designed to work on a static map, and the slow dynamics of the process will affect the gradient estimation.

Most of the time the EKF gives a good estimate of the gradient, but in some cases, particularly when a disturbance changes abruptly, the EKF will not be able to estimate the value of the gradient, which will have a negative influence on the controller performance. To mitigate this problem, we suggest to neglect these values of the estimated gradient, thus keeping the value of  $u_0$  i.e. setting the input to the integrator to zero when there are no trusted estimates of the gradient.

If the absolute difference between the value of the estimated output ( $\hat{y}$ ) and the actual value of the output ( $y$ ) is large this means that the EKF is not able to keep tracking, and the estimated gradient will be neglected. In addition to this comparison, a small on-delay will be added in order to give the EKF enough time to return to track, before reusing the estimated gradient.

Fig 10 shows the ESC controller after adding this feature. The maximum permissible error between the estimated and the actual value was selected to be 0.1 *kg/second*, and the on-delay was 5 seconds.

Similar to Fig. 9, the upper part in Fig. 11 shows the response of the controller, and the lower part shows the corresponding output of the crusher. The black curves shows

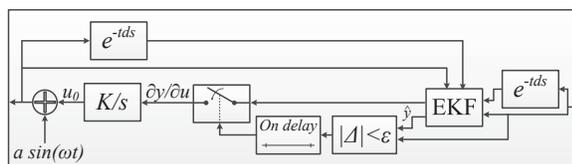


Fig. 10. Modified EKF-based algorithm

the response of the EKF based ESC with the modification described above, while the light curves shows the corresponding signals without the suggested modification. It is clear that without the modification there are large deviations from optimum caused by erroneous estimates of the gradient, and the proposed modification to the EKF algorithm mitigates the problem. The EKF method has been tuned to approximately

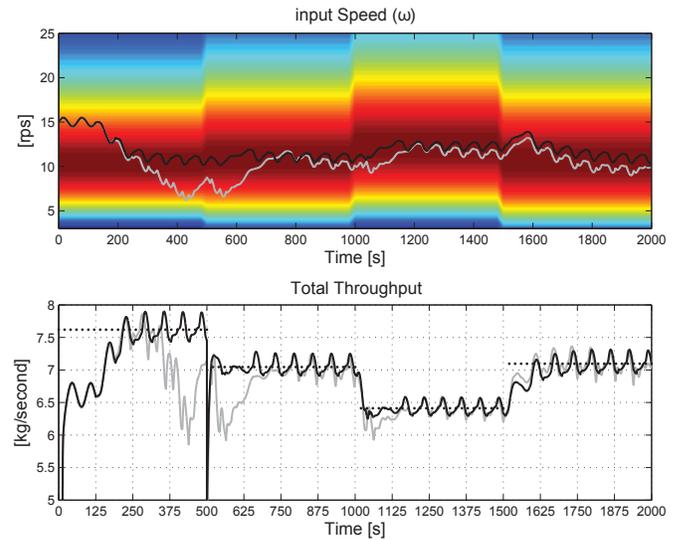


Fig. 11. Step Response of ESC with EKF, the black curves showing the response of the modified controller, while the light curves shows the response of the original controller

the same convergence rate as the BPF method, but achieves this with a lower perturbation frequency. This means that the rate of change in the actuated signal will be lower, which may prolong the life of the actuators.

### V. CONCLUSION AND FUTURE WORK

Two ESC algorithms have been tested on a dynamic model of a cone crusher and the results suggest that both of them are viable for on-line optimization of throughput.

The EKF based algorithm has advantages in a slightly simpler design and appears to require a lower perturbation frequency, but suffers from problems due to deteriorated gradient estimates in some situations. However, it is shown that this problem can be mitigated by a modification of the EKF algorithm.

The work will be extended into multi-variable optimization by using the CSS in addition to  $\omega$  as a control variable. Also, it can be applied to the maximization of individual size classes.

### REFERENCES

- [1] A. Lynch, P. Bush, and A. I. S. B. Unit, *Mineral crushing and grinding circuits: their simulation, optimisation, design, and control*. Elsevier Scientific Publishing Company, 1977.
- [2] E. Hulthén and C. Evertsson, "On-line optimization of crushing stage using speed regulation on cone crusher," in *Proceedings of XXIV International Mineral Processing Congress*, vol. 2, 2008, pp. 2396–2402.
- [3] C. Evertsson, "Output prediction of cone crushers," *Minerals engineering*, vol. 11, no. 3, pp. 215–231, 1998.

- [4] —, “Modelling of flow in cone crushers,” *Minerals engineering*, vol. 12, no. 12, pp. 1479–1499, 1999.
- [5] —, “Cone crusher performance,” Ph.D. dissertation, Chalmers University of Technology, 2000.
- [6] E. Lee and C. Evertsson, “A comparative study between cone crushers and theoretically optimal crushing sequences,” *Minerals Engineering*, vol. 24, no. 3, pp. 188–194, 2011.
- [7] M. Krstić and H. Wang, “Stability of extremum seeking feedback for general nonlinear dynamic systems,” *Automatica*, vol. 36, no. 4, pp. 595–601, 2000.
- [8] C. Zhang and R. Ordóñez, *Extremum-Seeking Control and Applications: A Numerical Optimization-Based Approach*. Springer, 2011.
- [9] M. Krstić, “Performance improvement and limitations in extremum seeking control,” *Systems & Control Letters*, vol. 39, no. 5, pp. 313–326, 2000.
- [10] G. Gelbert, J. Moeck, C. Paschereit, and R. King, “Advanced algorithms for gradient estimation in one-and two-parameter extremum seeking controllers,” *Journal of Process Control*, 2012.
- [11] A. Johansson, “Modeling and simulation of cone crushers,” in *Proceedings of IFACMMM 2009 Workshop on Automation in Mining, Mineral and Metal Industry*, 2009, pp. 13–18.
- [12] P. Itävu, “Dynamic modeling of a rock crushing process,” Master’s thesis, Tampere University of Technology, 2009.
- [13] K. Atta, A. Johansson, and T. Gustafsson, “Control oriented modeling of flow and size distribution in cone crushers,” *Submitted to Mineral Engineering*, 2012.
- [14] S. Broadbent and T. Callcott, “A matrix analysis of processes involving particle assemblies,” *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 249, no. 960, pp. 99–123, 1956.
- [15] R. King, *Modeling and simulation of mineral processing systems*. Elsevier, 2001.
- [16] P. Kapur, “The energy-size reduction relationships in comminution of solids,” *Chemical Engineering Science*, vol. 26, no. 1, pp. 11 – 16, 1971. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0009250971860761>
- [17] L. Zhao, L. Chen, D. Zhang, and Y. Li, “Modelling and simulation of particle breakage in jaw crushers,” *Applied Mechanics and Materials*, vol. 130, pp. 1148–1151, 2012.