Conversion of Iterative Balance Models to Directly Calculating Explicit Models for Real-time Process Optimization and Scheduling

Tomas Björkqvist[1]*, Olli Suominen[1], Matti Vilkko[1], Mikko Korpi[2]

[1] Department of Automation Science and Engineering, Tampere University of Technology, Tampere, Finland; * *tomas.bjorkqvist@tut.fi*

[2] Research Center Pori, Outotec (Finland) Oy, Pori, Finland

Simulation Notes Europe SNE 26(4), 2016, 257 - 266
DOI: 10.11128/sne.26.tn.10357
Received: November 10, 2016
Accepted: December 5, 2016 (Special Issue Review)

Abstract. Optimal utilization of complex processes involves real-time operational optimization and scheduling, especially in cases where the production line consists of both continuous and batch operated unit processes. This kind of real-time optimization requires process models which can be computed significantly faster than real-time. Iterative balance calculation is typically far too slow for these cases. This paper presents a method for converting an iterative balance model to a directly calculating model suitable for on-line process optimization. The approach is demonstrated with the first unit process in the copper smelting line, the flash smelting furnace (FSF). The method consisted of formulating an equation group based on the constrained FSF HSC-Sim model and solving the unknown parameters and static states with use of a symbolic calculation software. The solution was implemented as a function whose calculation time fulfilled the requirements for scheduling use.

Introduction

The general digitalization of society and advances in computational power have brought on a pronounced digitalization wave in process industry. Utilization of the advantages of digitalization can improve efficiency and the ability to stay competitive in increasing global competition in many conventional industrial processes. The design of industrial processes is often based on long term empirical and theoretical knowledge which has been incorporated into thoroughly built mathematical models.

These models often include iterative balance calculations to fulfill empirical and physical process constraints. These models are well suited for steady state process design and often used when offering, planning and constructing new process lines, however, they are often computationally too cumbersome for use in real-time solutions demanding short execution time.

Optimal utilization of processes should ideally include real-time operational optimization and scheduling where results can be presented to operators and/or process control quickly. Due to the time requirements and computational complexity of the optimization schemes, the underlying process models must be capable of producing results significantly faster than real-time. Thus, models requiring iterative calculations are typically too cumbersome to incorporate into the optimization. The high demand on execution time can often be compensated by lowering demands on model precision for the real-time operational optimization. Examples of demanding real-time optimization utilized in process design can be found in [1,2,3,4].

Good examples of thoroughly built steady state models can be found in metallurgy. Most metallurgical processes are old and have large societal impact which has allowed extensive development work to model process behavior over many decades. These processes comprise complex physical and chemical reactions and modelling has been both theoretical and empirical. To fulfill the basic requirement of mass and energy conservation and empirical observations iterative calculation is often employed. Lately also dynamic models purely based on fundamental physical laws have been successfully derived, e.g. [5] for the melting process in electric arc furnace.

These types of models are built on differential equations and needs careful parameter determination. They are normally solved with integration algorithms which, depending on model complexity, can be too slow for real-time operational optimization and scheduling.

The incentive for this study is the need for operational optimization of a copper smelting line. Optimal operation of a copper smelting line is challenging for the operators as the operation is divided into many complex individual sub processes. Plant wide optimization is required to maximize production and resource efficiency. Additionally, more challenging ores have to be used to retain economic competitiveness worldwide which increases the need for process optimization. Improved operation of copper smelting can provide improved utilization of different input materials and recyclants. Copper smelters present a challenging optimization problem where the harsh environment can prevent obtaining mineral and operational information, data is highly uncertain or measurements may be severely delayed. A full scale optimization of the complete process line will include a considerable amount of variables and require the consideration of large time horizons. Further, many of the underlying models are nonlinear. Thus, sub processes and the related models should be relatively lightweight in terms of their computational requirements. In principle, the development of optimization for a copper smelting line operation consists of modelling of unit processes and designing of optimization / scheduling for the combined unit process models.

Static input-output process models can be derived with use of mass and energy balances supplemented with sometimes uncertain process reaction knowledge completed with empirical knowledge. In principle, this empirical knowledge can be written as constraints in equation form. These equations can be completed with mass and energy balances to form a complete equation group determining process reactions. By solving the equation group, the unknown parameters and thereby the static process state can be solved under the given constraints. In practice this approach is challenging as the equations are often complex and manual solutions may be error prone and exceptionally time consuming.

Development of aids for this challenge started in the beginning of the 1970s under the scientific area of symbolic computation. Software programs for manual computation are called computer algebra systems (CAS) and are at present highly developed and even implemented in hand held calculators.

These systems include Mathematica [6] and Maple [7], the latter has been implemented in Matlab [8] as the Symbolic Math Toolbox. In later Matlab versions, the toolbox is based on the MuPAD symbolic engine originally developed at the University of Paderborn. Matlab offers a convenient way of shifting from symbolic calculus to numerical computation.

Utilization of symbolic computation for solving unknown variables of restricted mass balance equations seems to be a rare approach or rarely reported. A similar method was used in [9,10] in the same research group but the authors have not found similar work by others. Symbolic computation is, however, commonly utilized when forming first principle models [11,12,13]. Its use is especially convenient for model design with e.g. Lagrangian mechanics [14].

For optimization of the operation of the copper smelting line computationally lightweight models of all unit processes are required. This paper presents a method for converting an iterative balance model to a directly calculated model suitable for process operation optimization. The method is demonstrated with the first unit process in the copper smelting line, the flash smelting furnace (FSF).

1 Examples of Industrial Process Optiomization and Scheduling

Process optimization in general can be viewed as requiring predictive models capable of evaluating the evolution of the process under different process variables and operational schemes. In many cases linear models or finite response models are used to facilitate the fast calculation of predictions. Optimization determines the variables which minimize or maximize some objective function while fulfilling process constraints. The simplest objective is often the maximization of throughput. More advanced objectives may include considerations of energy use or different quality variables. When more exact predictions are required or linearization is not applicable for some other reason, nonlinear process models are used. Solutions will then require complicated optimization algorithms for the determination of optimal process variables and operation. In general, these algorithms require iterative calculation to find optimal values.

Scheduling problems determine which process units are used when and for which process tasks. Almost all scheduling problems consider batch processes and thus require integer variables introducing additional complexity to the optimization problem. Further complexity is introduced when the processes, such as copper production, include the combination of batch and continuous sub processes. Scheduling problem formulations can be roughly divided into discrete time or continuous time problems. In either implementation the number of variables is often in the hundreds and even multiple thousands of constraints are required. The most common method is to define the scheduling problem as a mixed integer linear program (MILP) which can typically be solved in seconds. In addition to production rates and task timings logistical concerns related to transfer of materials and maintenance are often incorporated into the formulation.

Scheduling has been in common use in many industrial applications, especially related to chemical processes, for decades. Some implementations of industrial scheduling and optimization include the scheduling of a pulp and paper machine reported in [15]. Here an optimal production schedule was defined and energy production and prices were considered in the objective. Steel production has often been considered in scheduling problems. One implementation was reported in [16] where the production of different product recipes was considered and the problem solved with decomposition. The optimization and scheduling of copper production has seldom been reported. One implementation was introduced in [17] where throughput is maximized while also enabling the consideration of different maintenance tasks. More recently, [18] reported an implementation of a greedy algorithm to plan the production of a copper plant. The authors introduced a nonlinear optimization of copper production in [19] where a simulation based approach was used.

Different optimization algorithms are available in many commercial products. MILP solvers are included in most computational software. In [17], the problem included 750 variables, of which 84 were binary integer variables, and 984 constraints with a solution time of under one second. MILP solvers roughly work by relaxing the integer constraints, find an optimal solution and if this does not fulfill the integer constraints perform a branch-and-bounding of the problem and find new optimal solutions for the new problems. Nonlinear solvers also require iterative search methods.

For example, Matlab includes the interior point and sequential quadratic programming algorithms for use with constrained nonlinear problems. In [19], an iterative simulation was used to predict the evolution of the process. Solutions were produced in about 60 seconds. Required iteration amounts are in the hundreds or thousands. This illustrates the need for lightweight models to enable real-time optimization.

2 Copper Production Line

Copper smelting plants convert the input materials, concentrates, which consist of mainly copper and iron sulphides, to almost pure copper through multiple oxidization stages. This begins from the mixing of a suitable concentrate mix with a copper content of 20-30 % which, after drying, is fed to the FSF. The mix reacts with the oxygen-enriched air feed and separates to matte (~60-65 % Cu) and slag. Silica flux is added to the FSF feed during operation to achieve suitable conditions for separation of matte and slag. The oxidization reactions generate heat though in some cases additional heating may be required.

Matte and slag are removed intermittently from the FSF, matte is moved to the converters, and slag is processed further in the slag treatment plant. After treatment, both FSF and converter slag can be recycled back to the FSF. Pierce-Smith converters (PSC) use a submerged feed of oxygen enriched air. Converters are operated in batches where first, in multiple slag-making stages, FSF matte is added between air blows. Here, most of the iron compounds will react and move to slag. Second, after removal of slag, in one longer copper-making stage the remaining sulphur is removed from copper compounds. Temperature is controlled with the addition of recycled material, e.g. scrap metal. The ensuing blister copper (~99 % Cu) is moved to anode furnaces where oxygen is removed from the blister copper and the copper is cast to anodes and finally transported to refinery for electrolytic purification to cathode copper. Figure 1 shows a full copper production line including both smelting and refining. A detailed description of the smelting process can be found for example in [20].

The FSF matte copper content can be viewed as one of the main decision variables in smelting as the higher copper content in matte (matte grade) is, the higher the copper content in the slag both in FSF and PSC and less blowing time in PSC.

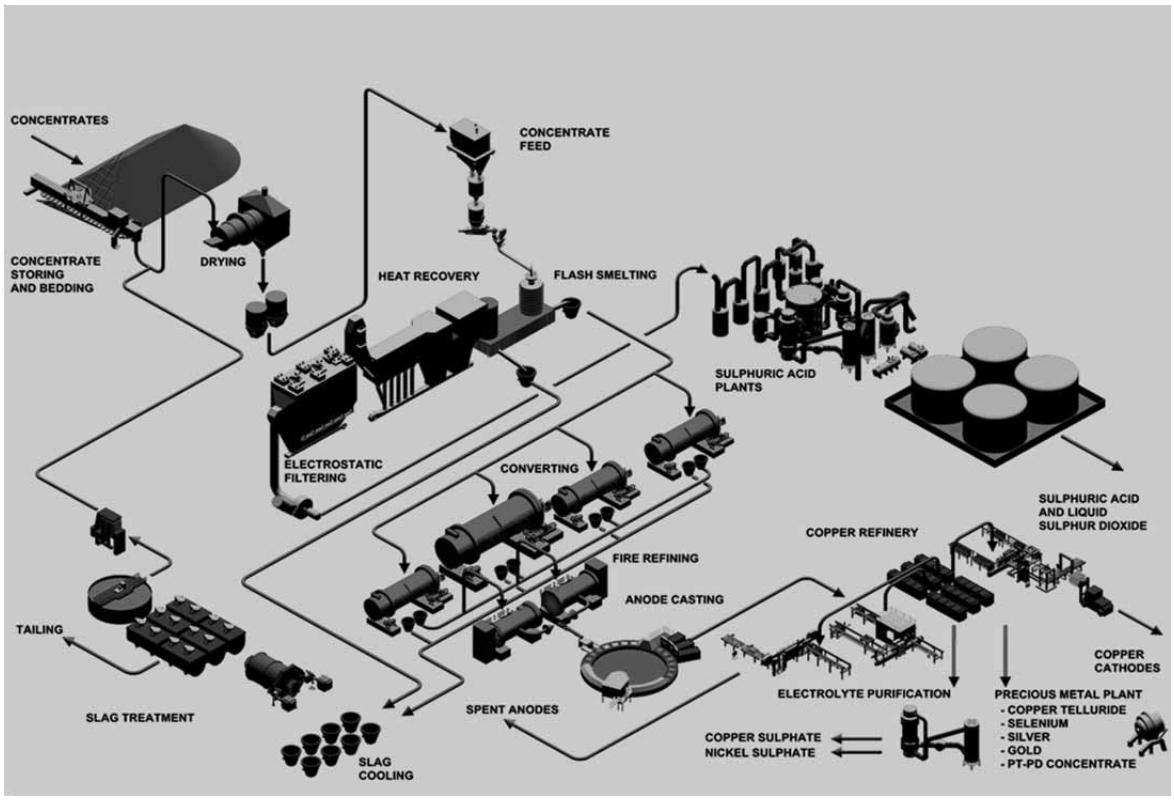

Figure 1: Flow sheet of copper process at Boliden Harjavalta [22].

Additionally, in FSF modelling matte grade is often used as a variable in distribution of other valuable metals, such as silver (Ag), cadmium (Cd), cobalt (Co), nickel (Ni), lead (Pb), tin (Sn), and zinc (Zn), to both matte and slag. Production bottlenecks include the production rate of FSF and the required converter tasks and availability of converters. Transportation of material from the FSF to converters is handled with cranes and may present limits for production rates. SO2 gases are produced in all production stages and the capacity of the gas treatment plant must be considered when determining the production rates and timings of different production tasks.

3 Model Conversion

The method for converting an iterative balance model to a directly calculating model is here demonstrated with a model of the flash smelting furnace, modelled in HSC-Sim [21]. HSC-Sim is a calculation module of HSC Chemistry software developed by Outotec. The name refers to the automatically utilized thermochemical database which contains enthalpy (H), entropy (S) and heat capacity (Cp) data for an extensive amount of chemical compounds.

The HSC-Sim module enables application of HSC Chemistry to a whole process made up of process units and streams. The HSC-Sim module consists of a graphical flowsheet and spreadsheet type process unit models. The custom-made variable list enables creation of different types of process models in chemistry, metallurgy, mineralogy, economics, etc. Each process unit is actually one Excel file. In the Distribution units the compounds are divided into elements and calculation is done with element distribution coefficients. Based on process knowledge some coefficients are defined as fixed. Coefficients for assisting elements in compound formation are calculated based on molar need and supply and called float. Surplus elements are divided with coefficients called rest. Units can be used together or separately and the calculations can be Excel- or DLL-based.

HSC Sim pyro models are mathematical process models based on mass- and energy balances and empirical knowledge controlling the equilibrium state. These models are successfully used in strategic planning of metal processing. The drawback of these models is the iterative calculation needed for reaching the equilibrium state. This iterative calculation is too slow for use in online process optimization of the whole smelter line.

3.1 Legacy model

The flash smelting furnace process has been modelled in HSC-Sim as a static division process with empirical knowledge controlling parts of the division coefficients. The implementation is a spreadsheet-like division calculation with iterative calculation to fulfill constraints derived from empirical and physical knowledge.

The model consists of three main spreadsheets; Input, Distributions and Output, each containing between 146 and 424 rows and 68 columns. The Input sheet is sparsely filled with element mass flows and describes how input compounds in different streams are broken up to elements according to chemical molar consistency. The Distributions sheet is sparsely filled with distribution coefficients dividing element mass flow into compounds for different output streams partly according to chemical reactions. The Output sheet is filled with corresponding element mass flows that build up the output compounds in different output streams. In addition to the three main spread sheets, a Controls sheet includes 27 empirical process observations that must be fulfilled in the stationary state.

In principle, the distribution from input compounds to output compound is built up around how the main elements copper (Cu) and iron (Fe) is distributed between compounds in the output streams. The chemical reactions require assisting element as oxygen (O) and silicon (Si) which are brought in as floating elements. Sulphur (S) is partly handled as a main element and partly as an assisting element. As a result, the model consists of some fixed distribution coefficients, many coefficients which are iteratively adjusted to fulfill the empirical observations and numerous coefficients calculated as float according to corresponding chemical reactions or as rest for surplus elements. The model is thus a system of four spread-sheets with a large number of interconnected cells. An iterative routine is used to solve the distribution coefficients and thereby the element and compound streams in the stationary state.

The calculation is very useful for off-line strategic planning of metal processing. The calculation is, however, too slow for real-time process optimization.

3.2 Method for derivation of fast calculating model

In general, the objective for the study was to find a method for converting iterative output controlled balance models to directly calculating models suitable for process scheduling. The basic idea was to form a symbolic equation group based on the flash smelting furnace HSC-Sim model and to solve this group analytically with symbolic computation to achieve causal outputs as direct functions of inputs. The solution is possible due to empirical knowledge included in the Controls sheet of the FSF HSC-Sim model.

Thus, the task was to write a fully parametrized equation group based on the FSF HSC-Sim model where the equations are based on the equations of empirical knowledge in the Controls sheet. The model is in this analytic approach simplified. The input elements include only the main elements; copper (Cu), iron (Fe), nitrogen (N), oxygen (O), sulphur (S), silicon (Si) and other content (Ot). The distribution of the elements between the output streams, which are settler gas, settler fume, settler dust, slag and matte, is fully in line with the FSF HSC-Sim model. The eight equations determining empirical knowledge regarding the main elements was chosen as base for the equations. To enable an analytic solution with the symbolic software the equation group has to be exactly determined.

The equation group formulation starts with defining all basic variables as symbolic variables. This example included 7 element mass flows, 23 distribution coefficients for element distribution to output streams and 41 distribution coefficients for element distribution into compounds in the different output streams. The main formulation work is to define the relationship between these variables with emphasis on the formulation of the float and rest variables. Here, this part required about 75 definitions. After these definitions, the output compounds can be formulated. Afterwards, the final equations based on the empirical knowledge in the Controls sheets can be written. To ease the derivation of the analytic solution of the software the nonlinearities in the empirical knowledge were linearized. The same variables as the manipulated variables in the iterative solution of HSC-Sim model were chosen as variables for the calculation to solve. They were; distribution coefficient for Fe to matte, distribution coefficient for Fe in slag to FeS, distribution coefficient for Cu to slag, distribution coefficient for Fe in matte to Fe3O4, Ot to matte, Si input stream, O input stream and distribution coefficient for Fe in slag to Fe3O4.

This study utilizes the Symbolic Math Toolbox in the Matlab software. With the relationships concerning use of oxygen still undefined, the solver managed to achieve a fully symbolical solution in around five minutes with a laptop.

When oxygen is taken into account, the solver has been forced to settle for a numeric approximation, which still includes all the variables in an appropriate manner. The length of the analytic solutions is over 25000 characters. The solutions are at this stage provided with the values of the fixed variables. The last task of the program is to produce usable functions of the long analytic solutions.

4 Model Validation and Discussion

Model validation is performed to ensure usability of the model in real-time process optimization and scheduling. As copper content in matte is a good measure of the process state, the validation is performed at varying matte copper percentage.

4.1 Similarity to legacy model

Figure 2 shows a comparison between the analytical direct solution results, with the blue line, and iteratively calculated HSC-Sim results, red line, as function of matte copper percentage.

The cause for the differences is the fact that the analytically solved model is a simplified model of the process including only the main elements. E.g. both silicon and oxygen is consumed by other minor compounds which are not included in the model. The difference is mainly a shift of magnitude which can easily be compensated by a term proportional to the total concentrate flow. With this compensation the analytically solved model is adequate for the on-line utilization.

4.2 Calculation time

As the optimization and scheduling algorithm calls the model hundreds of times per second the calculation time has to be short. A test function call from Matlab showed that the execution time is only some milliseconds for calls of two to eight variables, which is sufficient for the on-line utilization. The calculation time for the iterative solution of the HSC-Sim model is a few seconds.

5 Model Utilization

The directly calculating model of the flash smelting furnace process will be utilized in scheduling of a copper production line to optimize production and costs. When solving the equation group, the solvable variables can be freely chosen. There are two evident ways of model formulation that can be utilized.

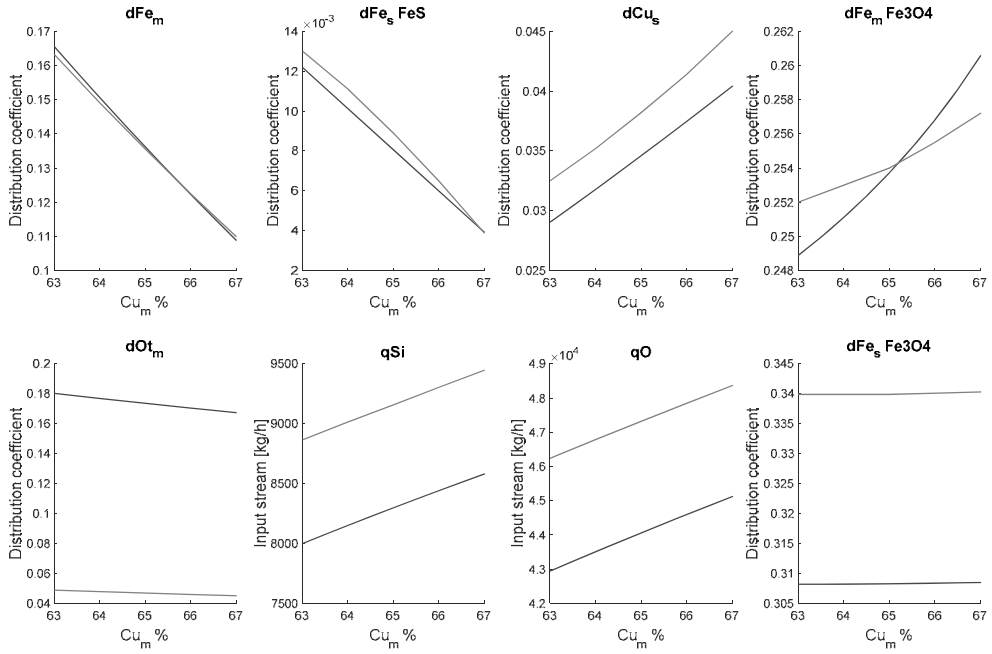

Figure 2: A comparison between analytical solution results with blue line and iteratively calculated HSC-Sim results with red line.

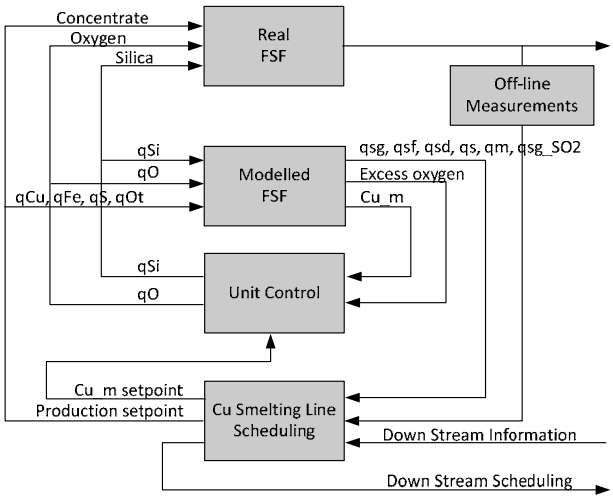

Figure 3: Direct input output model utilized in scheduling.

5.1 Direct input output

A natural solution would be to form a direct input output model to mimic the real smelting process. Figure 3 represents a scheduling structure that utilizes the input output model. As scheduling is a high level task whose interests are in production rate and oxidation level in first stage smelting, a lower level control structure has to deal with the unit control of the flash smelting furnace. This is shown as feedback control of the open loop model. In practice, this could be a sub optimization task for the scheduling routine.

5.2 Closed analytic solution

To enhance the direct scheduling interests, the required control variables can directly be chosen as solvable variables in the equation group. The static model allows us to utilize a closed analytic solution whose scheduling structure is clear and shown in Figure 4. This direct solution will not need the sub optimization. Feedback from the off-line measurements compensates for model inaccuracy.

5.3 Model based schedule calculation

To demonstrate the usage of the directly calculating model an example schedule is derived where special attention is paid to the calculation time.

A similar routine will be called at high frequency when the model is utilized in the real-time operational and scheduling optimization. The routine is called at the moment when the nonlinear optimization algorithm executes a new iterative schedule. The example is in line with the utilization of the closed analytic solution presented in Figure 4.

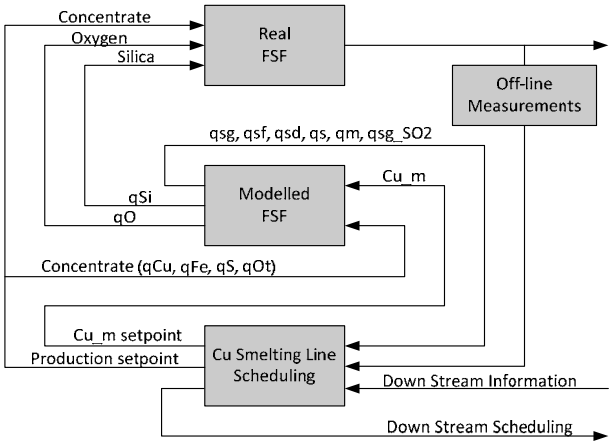

Figure 4: Closed analytic solution utilized in scheduling

The flash smelting furnace is here regarded as a static smelting process feeding parts of the formed compounds to matte. Matte volume in the bottom of the furnace is assumed to be fully mixed i.e. we have a static material distribution process followed by a fully mixed stock. The example comprises variable time moments which can be chosen by the optimization algorithm and where the algorithm can suggest changes in production rate and copper content of feed to matte. Additionally, the algorithm schedules tapping of matte for further delivery to Pierce-Smith converters. The example routine utilizes the closed analytic solution and simply track element flows to matte and respective stock situations starting from an initial state. As the stock is assumed to be fully mixed the copper percentage can be directly calculated as copper mass of total mass in storage. Figure 5 shows a plausible schedule of an optimization algorithm including two steps in concentrate feed rate, one step in copper content of feed to matte and one matte tapping.

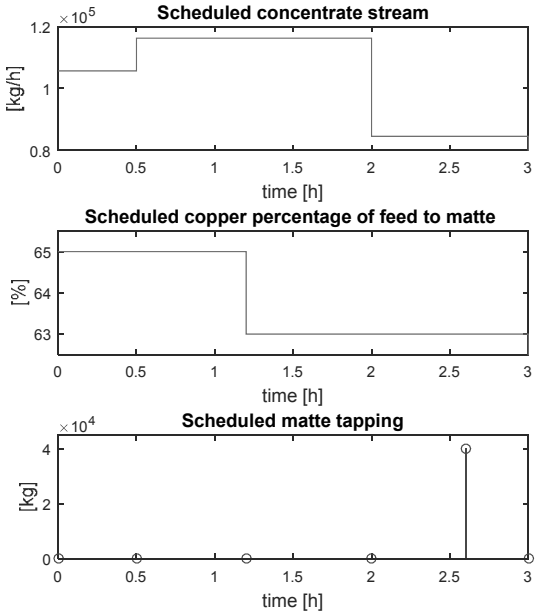

Figure 5: A plausible schedule of an optimization algorithm.

When changing the concentrate feed rate, the internal element mix has been kept constant. Figure 6 shows the outcome of the example routine. First the routine has directly calculated the required silica and oxygen for the different static smelting process steps according to the closed analytic solution. Second the routine has calculated the stepwise changing element streams to matte and kept track of the total mass in matte and the copper amount in matte to be able to track the copper percentage in matte. The copper matte percentage is exact in the figure at the time moments. In reality the change between the time moments in copper matte percentage is similar to a first order step response due to the integrator effect of the matte volume in the bottom of the furnace. Due to the immediate tapping the total matte mass is not exact in the period before the tapping moment. This inexactness in plotted figures is not a problem as the optimization algorithm only need the values at the algorithm chosen moments. The necessity of including the matte volume in the schedule calculation is revealed when comparing the copper content of feed to matte to the actual copper matte percentage in the matte volume i.e. the copper matte percentage of the tapped copper delivered to the next unit process in the smelter line.

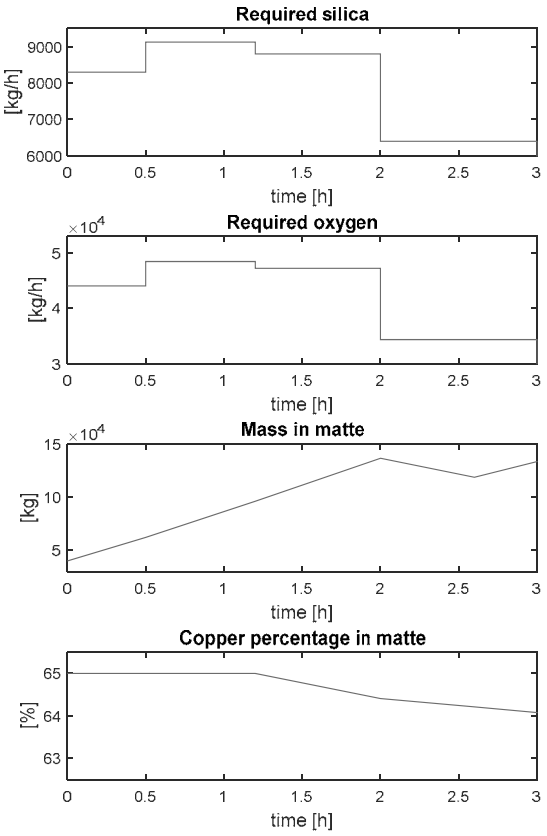

Figure 6: Outcome of example routine including both needed process feeds and matte state.

The calculation time for the example routine executed in Matlab is about 15 ms in a laptop computer. The routine included two function calls which prolonged the execution time but these function calls can be integrated to one when optimizing speed. The execution time is estimated to be short enough for real-time operational and scheduling optimization. FSF models with execution time in few seconds are earlier implemented successfully in controlling the FSF unit process [23], but the execution time demand changes significantly when the whole smelter line operation is to be optimized.

6 Conclusions

The objective of this study was to develop a method for converting iterative output controlled balance models to directly calculating models for process optimization and scheduling.

This method was used in the case of a flash smelting furnace, previously modelled in HSC-Sim. The fast calculating model is to be used in optimization of the total production line operation.

The method consisted of formulating an equation group based on the constrained FSF HSC-Sim model and solving the unknown parameters and static states with use of a symbolic calculation software. The study was successful even if it requires careful formulation work and the solution matched the solution of the original model. The equation group should be fully determined to enable a solution. The solution was implemented as a direct calculation function whose calculation time fulfilled the requirements for scheduling use.

The advantage with the approach is that even though the length of the generated functions disables model maintenance in function form, functions can easily be recalculated after updates in the HSC-Sim model are done. The modelling method has shown to be a powerful general way of converting complex iteratively solvable models to fast directly calculating models for utilization in real-time operational and scheduling optimization.

The presented demonstration model did not include an energy balance and thereby the amount of nitrogen (N) feed is kept constant even if the nitrogen feed is in practice the means to affect process temperature. The legacy model is built on the assumption that temperature is on normal level which enables a mass balance without temperature dependency. The energy balance will be included in future work.

This paper is an extended version of a paper presented at the 9th EUROSIM Congress on Modelling and Simulation [24].

Acknowledgment

This work was carried out in the SIMP research program coordinated by the Finnish Metals and Engineering Competence Cluster (FIMECC) Ltd. The support is gratefully acknowledged. The authors are also grateful for the process expert knowledge and good working possibilities provided by Petri Latostenmaa and Ville Naakka at Boliden Harjavalta.

References

[1] Harjunkoski I, Bauer R. Configurable Scheduling Solution using Flexible Heuristics, Proceedings of the 26th European Symposium on Computer Aided Process Engineering – ESCAPE 26 2016, pp. 2362-2366.

[2] Touretzky C, Harjunkoski I, Baldea M. A Framework for Integrated Scheduling and Control using Discrete-Time Dynamic Process Models, Proceedings of the 26th European Symposium on Computer Aided Process Engineering – ESCAPE 26 2016, pp. 601-606.

[3] Pelusi D. PID and intelligent controllers for optimal timing performances of industrial actuators, (2012) International Journal of Simulation: Systems, Science and Technology, 13 (2), pp. 65-71.

[4] Pelusi D. Improving settling and rise times of controllers via intelligent algorithms (2012) Proceedings - 2012 14th International Conference on Modelling and Simulation, UKSim 2012, art. no. 6205447, pp. 187-192.

[5] Logar V. Modelling and Simulation of the Electric Arc Furnace Processes. Simulation Notes Europe SNE 26(2), 2016, 91-98. DOI: 10.11128/sne.26.tn.10335.

[6] https://www.wolfram.com/mathematica/, 23.6.2016

[7] http://www.maplesoft.com/solutions/education/, 23.6.2016

[8] www.mathworks.com/, 23.6.2016

[9] Korpela T, Björkqvist T, Majanne Y, Lautala P. (2014). Online Monitoring of Flue Gas Emissions in Power Plants having Multiple Fuels. IFAC-PapersOnLine, 19(1), 1355-1360. DOI: 10.3182/20140824-6-ZA-1003.01913

[10] Korpela T, Suominen O, Majanne Y, Laukkanen V, Lautala P. 2016. Robust Data Reconciliation of Combustion Variables in Multi-Fuel Fired Industrial Boilers. Control Engineering Practice. Vol. 55. P. 101–115. DOI: 10.1016/j.conengprac.2016.07.002.

[11] Belkhir W, Ratier N, Nguyen DD, Yang B, Lenczner M, Zamkotsian F, Cirstea H. Towards an automatic tool for multi-scale model derivation illustrated with a micro-mirror array, 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing 2015, pp. 47-54. DOI 10.1109/SYNASC.2015.17

[12] Lin J, Chen C. "Computer-aided-symbolic dynamic modeling for Stewart-platform manipulator," Robotica, vol. 27, pp. 331-341, 2009.

[13] Yakhno V, Altunkaynak M. "A polynomial approach to determine the time-dependent electric and magnetic fields in anisotropic materials by symbolic computations," COMPEL - the International Journal for Computation and Mathematics in Electrical and Electronic Engineering, vol. 35, pp. 1179-1202, 2016.

[14] Moosavian SAA, Papadopoulos E, "Explicit dynamics of space free-flyers with multiple manipulators via SPACEMAPLE," Adv. Rob., vol. 18, pp. 223-244, 2004.

[15] Säynevirta S, Luotojärvi M. Integrated paper production and energy planning. PulPaper 2004 conferences – Energy and carbon management (2004), Helsinki, Finland

[16] Harjunkoski I, Grossmann IE. (2001). A decomposition approach for the scheduling of a steel plant production. Computers & Chemical Engineering, 25(11), 1647-1660.

[17] Harjunkoski I, Fahl M, Borchers HW. Scheduling and Optimization of a Copper Production Process. Logistic optimization of chemical production processes (2008).

[18] Navarra A. Automated scheduling and scientific management of copper smelters. Mineral Processing and Extractive Metallurgy Vol. 125 , Iss. 1,2016.

[19] Suominen O, Mörsky V, Ritala R, Vilkko M. Framework for optimization and scheduling of a copper production plant. Proceedings of the 26th European Symposium on Computer Aided Chemical Engineering ESCAPE 26. Volume 38, 2016, Pages 1243–1248, DOI 10.1016/B978-0-444-63428-3.50212-5.

[20] Schlesinger ME, King MJ, Sole KC, Davenport WG. (2011). Extractive metallurgy of copper. Elsevier.

[21] http://www.outotec.com/en/Products--services/HSC-Chemistry/Calculation-modules/Sim--process-simulation/, 23.6.2016

[22] http://www.boliden.com/Documents/Press/Publications/Place%20broschures/boliden-harjavalta-en.pdf, 23.6.2016

[23] Björklund P, Ranki T, Miettinen E. Recent experiences with implementing dynamic process control and monitoring in the flash smelting process, Copper 2013 Conference, Chile. Volume III, Book 1: Nickolas Themelis symposium on pyrometallurgy and process engineering, 2014, Pages 153-164.

[24] Björkqvist T, Suominen O, Vilkko M, Korpi M. From Iterative Balance Models to Directly Calculating Explicit Models for Real-time Process Optimization and Scheduling. 2016 9th EUROSIM Congress on Modelling and Simulation, 184-189. DOI 10.1109/EUROSIM.2016.9