Technical University of Košice

Faculty of Mining, Ecology, Process Control and Geotechnologies

Design and Implementation of Modern Methods for Modeling and Control of Technological Objects and Processes

Dissertation Thesis

2021 **Michal Takáč**

Technical University of Košice

Faculty of Mining, Ecology, Process Control and Geotechnologies

Design and Implementation of Modern Methods for Modeling and Control of Technological Objects and Processes

Dissertation Thesis

Study Programme:	Cybernetics
Field of study:	Process Control
Department:	Institute of Process Control and Informatization (URIVP)
Supervisor:	prof. Ing. Ivo Petráš, DrSc.
Consultant(s):	

Košice 2021

Michal Takáč

Abstract

Text abstraktu v svetovom jazyku je potrebný pre integráciu do medzinárodných informačných systémov. Ak nie je možné cudzojazyčnú verziu abstraktu umiestniť na jednej strane so slovenským abstraktom, je potrebné umiestniť ju na samostatnú stranu (cudzojazyčný abstrakt nemožno deliť a uvádzať na dvoch strabách).

Keywords

Steelmaking, Mathematical modeling, Visualization, Virtual Reality

Abstrakt

Abstrakt je povinnou súčasťou každej práce. Je výstižnou charakteristikou obsahu dokumentu. Nevyjadruje hodnotiace stanovisko autora. Má byť taký informatívny, ako to povoľuje podstata práce. Text abstraktu sa píše ako jeden odstavec. Abstrakt neobsahuje odkazy na samotný text práce. Mal by mať rozsah 250 až 500 slov. Pri štylizácii sa používajú celé vety, slovesá v činnom rode a tretej osobe. Používa sa odborná terminológia, menej zvyčajné termíny, skratky a symboly sa pri prvom výskyte v texte definujú.

Kľúčové slová

Oceliarstvo, Matematické modelovanie, Vizualizácia, Virtuálna realita

Assign Thesis

Namiesto tejto strany vložte naskenované zadanie úlohy. Odporúčame skenovať s rozlíšením 200 až 300 dpi, čierno-bielo! V jednej vytlačenej ZP musí byť vložený originál zadávacieho listu!

Declaration

I hereby declare that this thesis is my own work and effort. Where other sources of information have been used, they have been acknowledged.

Košice, April 1, 2021 . . . .

Signature

Acknowledgement

I would like to express my sincere thanks to my supervisor Prof. Ing. Ivo Petráš, DrSc., the main Supervisor, for his constant, and constructive guidance throughout the study. To all other who gave a hand, I say thank you very much.

Preface

Predhovor (*Preface*) je povinnou náležitosťou záverečnej práce, pozri (**?**). V predhovore autor uvedie základné charakteristiky svojej záverečnej práce a okolnosti jej vzniku. Vysvetlí dôvody, ktoré ho viedli k voľbe témy, cieľ a účel práce a stručne informuje o hlavných metódach, ktoré pri spracovaní záverečnej práce použil.

Contents

List of Figures

List of Tables

List of Terms

Dizertácia je rozsiahla vedecká rozprava, v ktorej sa na základe vedeckého výskumu a s použitím (využitím) bohatého dokladového materiálu ako i vedeckých metód rieši zložitý odborný problém.

Font je súbor, obsahujúci predpisy na zobrazenie textu v danom písme, napr. na tlačiarni. To čo vidíme je písmo; font je súbor a nevidíme ho.

Kritika je odborne vyhrotený, prísny pohľad na hodnotenú vec. Medzi recenziou a kritikou je taký pomer ako medzi diskusiou a polemikou. Pri kritike treba prísnosť chápať v tom zmysle, že sa v nej okrem iného navrhuje, ako hodnotené dielo skvalitniť.

Meter (m) je vzdialenosť, ktorú svetlo vo vákuu prejde za časový interval 1/299 792 458 sekundy.

Písmom rozumieme vlastný vzhľad znakov.

Problém termín používaný vo všeobecnom zmysle vo vzťahu k akejkoľvek duševnej aktivite, ktorá má nejaký rozoznateľný cieľ. Samotný cieľ nemusí byť v dohľadne. Problémy možno charakterizovať tromi rozmermi – oblasťou, obtiažnosťou a veľkosťou.

Proces je postupnonosť či rad časovo usporiadaných udalostí tak, že každá predchádzajúca udalosť sa zúčastňuje na determinácii nasledujúcej udalosti.

1 Introduction

The objective of process control is to keep key process-operating parameters within narrow bounds of the reference value or setpoint. Controllers are used to automate a human function in an effort to control a variable. A basic controller can keep an individual loop on an even point, so long as there is not too much disruption. Complex processes like ones in metallurgy might employ dozens or even hundreds of such controllers, but keeping an eye on the big picture was, until not so long ago, a human process.

Although a device was used to automate a human function in an effort to control a variable, there was no sense of what the process was doing overall. A basic controller could keep an individual loop on an even keel, more or less, so long as there was not too much disruption. Complex processes might employ dozens or even hundreds of such controllers, each with its performance displayed on a panel board, but keeping an eye on the big picture was still a human process.

When distributed control system (DCS) platforms were introduced in the 1970s, they simplified the mechanics of the panel board, but did not do much to improve its capabilities. Big-picture analysis was still largely a human responsibility. Sure, getting beyond the technical constraints of pneumatic field devices with their troublesome compressed air tubing made it easier to install more instruments and actuators, but the basic control concepts did not really change. Any movement to advanced process control (APC) and other forms of control optimization were still in their infancy. Process automation capable of supporting APC had to encompass many technologies and techniques. It was characterized by incorporating many more input data points into algorithms and orchestrating more complex sequences.

The transition to process automation and advanced process control (APC) was empowered by being able to create an all-encompassing platform capable of coordinating more than single loops or small cascade groups. One major advantage of newer platforms is the ability to optimize a process to suit the owner's specific economic goals based on any number of desired outcomes. The process automation system can operate the plant to minimize energy consumption, maximize output, and deliver specific product quality attributes.

Implementing such systems is challenging. During the initial design phase of a control system upgrade or a new installation, it is far too easy to focus just on process fundamentals, and never get beyond considering desired steady-state conditions. Automation system upgrades and new installations can therefore miss opportunities to engage with process and automation technology experts capable of uncovering better ways of doing things. Many capabilities of modern process automation systems are still underutilized in most process plants. Far fewer companies use APC as effectively as they could, even though basic APC technologies have been around for decades.

Modeling, together with simulation, are important parts of engineering. Integrating them into engineer's toolbelt yields many benefits. Before actual testing in physical reality, one can do many different virtual tests by simulating the modeled phenomena with lower overhead of financial requirements. There's also close to zero risk of hurting anybody (most notable risk being ill-treatment of electricity). Still, coming up with and refining actual mathematical or physical model, to the point of it being accurate enough for the job at hand, takes a lot of time. Also, actual simulation part can take hours or days to converge. However, it can be argued that, in some industrial applications, broad parameter space mapping is sometimes more valuable than higher order analysis of fewer design points (Harwood et al., n.d.).

With the continuation of expanding computational power available for engineers that can tap into it for their simulations, the time for simulation to converge decreases accordingly. it becomes clear enough that ...

We'll discuss the path towards interactive simulation in section 3.1.

2 Basic Oxygen Furnace

The oxygen converter process (LD/BOF and its derivatives) is a refinement step in the production of steel from ore. The main purpose of the process is to remove excess carbon from the pig iron produced in the blast furnace. The main feature of the process is to add oxygen through a top lance, in order to remove unwanted impurities through oxidation. It is the main method of carbon and low alloy steelmaking, annual production approaching to 60% of total crude steel production Jalkanen (2006). BOF is a widely preferred and effective steelmaking method due to its high productivity and considerably low production cost Wang et al. (2010). LD/BOF process consists of these subprocesses:

1. Charging slag
2. Charging hot metal
3. Oxygen blowing and addition of slagging and alloying additives 4. Measurement of steel temperature and composition
4. Tapping of steel
5. Tapping of slag

The first commercial operation of steelmaking with oxygen top blowing in the converter was in the early 1950s at Linz and Donawitz (Austria). This manner of steelmaking became known as Linz-Donawitz or LD process. For many years now, most of the steel has been made by top oxygen blowing for which different names are given. For example, in European steel plants the process is still called LD; in the UK, BOS (basic oxygen steelmaking); in the Far East and America, BOF (basic oxygen furnace), with the exception of U.S. Steel where it is called basic oxygen process Turkdogan (1996).

In modern steel mills about 300 tons of steel are produced within a 30-40 minute cycle. Various additives are added during the process to adapt the steel quality and slag formation. The converter furnace is inclined during charging and tapping. The converter has a vertical position during oxygen blowing. The changes in the position of the converter

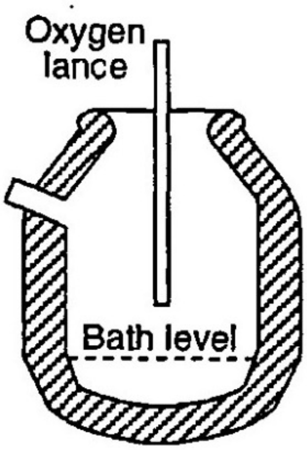


Fig. 2 – 1: Production of steel in the converter with top-blown oxygen (LD/BOF) (Turkdogan, 1996).

during the individual elementary processes are shown in the figure **??**.

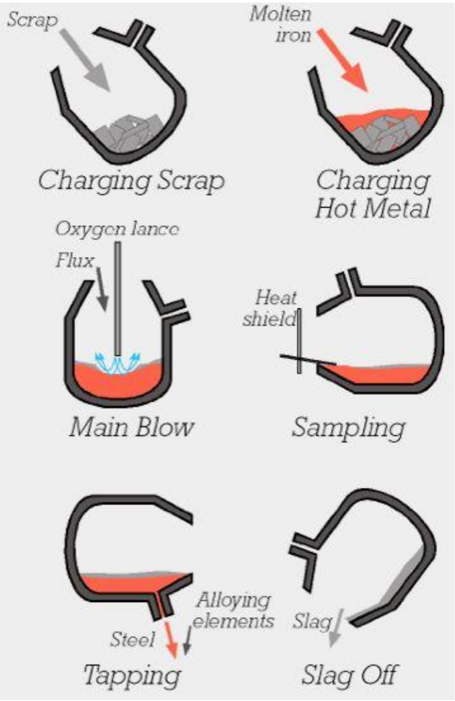


Fig. 2 – 2: Representation of elementary processes in LD converter.

Depending on local operating conditions, availability of scrap, blast furnace iron and the ex- tent of hot metal pretreatment, the metallic charge (LD/BOF, Q-BOP) is 75 to 95% pig iron and the remainder is steel scrap. The types of scrap used are usually those produced in a steel mill: sheet scrap, damaged molds, bimetallic cans etc. (Turkdogan, 1996).

Oxygen is blown at a high speed (up to twice the speed of sound) on the surface of the metal bath in the converter and so-called hot area is formed in the region where the oxygen stream hits the surface. The oxidation products dissolve in the slag, with the exception of carbon monoxide, which passes through the slag layer and forms the major component of the converted gas. The oxidation intensity of the individual elements depends on their chemical affinity for oxygen. Carbon oxidation is one of the most important processes. Carbon is oxidized in the metal during the steelmaking process by the influence of oxygen, in particular on CO and partly on CO_2, depending on the reactions

$$\mathrm{C} + \frac{1}{2}\mathrm{O_2} \longrightarrow 2\,\mathrm{CO} \tag{2.1}$$

$$\mathrm{C} + \mathrm{O_2} \longrightarrow \mathrm{CO_2} \tag{2.2}$$

Manganese is oxidized to MnO

$$\mathrm{Mn} + \frac{1}{2}\mathrm{O_2} \longrightarrow \mathrm{MnO}, \tag{2.3}$$

Phosphorus is undesirable in steel and oxidizes to P_2O_5

$$2\,\mathrm{P} + \frac{5}{2}\mathrm{O_2} \longrightarrow \mathrm{P_2O_5}. \tag{2.4}$$

Sulphur is a harmful element and passes into the slag in the form of CaS based on the reaction of CaO

$$\mathrm{CaO} + \mathrm{MnS} \longrightarrow \mathrm{CaS} + \mathrm{MnO} \tag{2.5}$$

whereby MnS is formed by reaction

$$\mathrm{Mn} + \mathrm{S} \longrightarrow \mathrm{MnS} \tag{2.6}$$

and sulphur also goes out in the form of gas as SO_2

$$\mathrm{S} + \mathrm{O_2} \longrightarrow \mathrm{SO_2}. \tag{2.7}$$

Silicon has a high affinity for oxygen, so it is easily oxidized to form SiO_2

$$\mathrm{Si} + \mathrm{O_2} \longrightarrow \mathrm{SiO_2}. \tag{2.8}$$

In the initial stages of blowing, most of the silicon oxidizes to form a slag of low basicity - the composition of the metal and slag changes TODO: DOPLNIT ZDROJ!!!.

An intense oxygen flow induces fluid flows in the iron bath, forcing the highly oxidized metal and the molten oxidation products from the iron bath surface to penetrate into the bath, where they react with the “fresh” hot metal with a high content of impurities and therefore loss of iron in the form of FeO and Fe_2O_3should also be considered:

$$\mathrm{Fe} + \frac{1}{2}\mathrm{O_2} \longrightarrow \mathrm{FeO} \tag{2.9}$$

$$2\,\mathrm{Fe} + \frac{3}{2}\mathrm{O_2} \longrightarrow \mathrm{Fe_2O_3}. \tag{2.10}$$

This oxygen stream and gas bubbles generated in the bath bring portions of the iron melt to the slag. The heat generated by the highly exothermic oxidation reactions is consumed by heating and melting the feed materials, heating the iron bath, slag and carbon oxides that are formed during the oxidation of the carbon and are partially lost to the environment during the blowing process.

The resulting SiO_2 passes into the slag as $2\,CaO \cdot SiO_2$ according to the equation

$$SiO_2 + 2\,CaO \longrightarrow 2\,CaO \cdot SiO_2 \tag{2.11}$$

and additionally, P_2O_5 passes into the slag as $3\,CaO \cdot P_2O_5$ according to the equation Laciak et al. (2017)

$$P_2O_5 + 3\,CaO = 3\,CaO \cdot P_2O_5 \tag{2.12}$$

Circulation in the iron bath caused by the flow of oxygen, rising gas bubbles and purging of inert gas through the lower tubes in converters with combined blowing type transports minor iron melt components (C, Si, Ti, Mn, P, V, etc.) to the upper bath layers (Jalkanen, 2006).

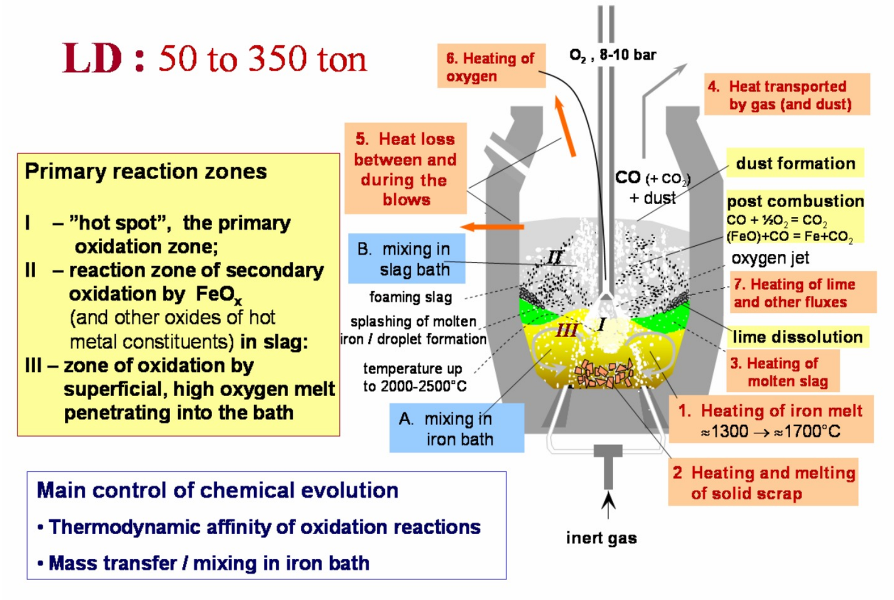

LD : 50 to 350 ton
6. Heating of oxygen
O₂ , 8-10 bar
4. Heat transported by gas (and dust)
5. Heat loss between and during the blows
dust formation
CO (+ CO₂) + dust
post combustion
CO + ½O₂ = CO₂
(FeO)+CO = Fe+CO₂
oxygen jet
Primary reaction zones
I – "hot spot", the primary oxidation zone;
II – reaction zone of secondary oxidation by FeOₓ (and other oxides of hot metal constituents) in slag:
III – zone of oxidation by superficial, high oxygen melt penetrating into the bath
B. mixing in slag bath
foaming slag
splashing of molten iron / droplet formation
temperature up to 2000-2500°C
II
I
III
7. Heating of lime and other fluxes
lime dissolution
3. Heating of molten slag
1. Heating of iron melt ≈1300 → ≈1700°C
A. mixing in iron bath
2 Heating and melting of solid scrap
Main control of chemical evolution
• Thermodynamic affinity of oxidation reactions
• Mass transfer / mixing in iron bath
inert gas

Fig. 2 – 3: Chemical and thermal processes in LD converter (Jalkanen, 2006).

In order to monitor and control the process, different measuring systems can be employed to give feedback to either the operator or directly to existing system for automated control. These measurements can be either direct or indirect as well as with or without time delays. The real-time information on the process state can be obtained from a number of sensors, e.g. off-gas flow rate from venturi; off-gas temperature with temperature sensors; indication of changes in slag level from sound level measurement with microphone, accelerometer on lance or vessel or lance elongation sensor; pressure difference with ambient pressure with hood pressure sensor, etc.

All these signals can, in principle, be used for closed-loop control. Changes in lance height are easier to measure as well as set and therefore preferable to use in a closed- loop control system. Oxygen gas flow rate should not be changed more than approximately ±5% since the nozzle is designed for a specific flow rate (Widlund et al., 1998).

TODO: DOPLNIT SCHEMU RIADIACEHO SYSTEMU

Different operators apply different rules for deciding when to stop the blowing of oxygen. These operator induced irregularities make it difficult to achieve production disturbances. The converter process is only one link in a chain of unit operations from raw materials to final product. Therefore, production disturbances can change process conditions. For example, poor blast furnace performance might lead to insufficient hot metal supply or even production disruptions. Under these circumstances, more metal scrap and fuel should be used instead of hot metal to uphold production volumes. Logically, this implies changes in blowing practice (Widlund et al., 1998).

3 Computer-aided Mathematical Modeling and Numerical Simulation

The motivation for using computer simulations to investigate metallurgical processes is two-fold. First, it enables design changes to be tested before building a prototype, which naturally leads to a lower total design cost. Second, it makes it possible to investigate phenomena that cannot easily be measured or observed in the process. Even a seemingly simple operation such as the continuous measurement of the temperature during the decarburization process is difficult due to the very high temperatures in the process and generally harsh conditions prevailing in the steel plants (Ersson and Tilliander; 2018).

In metallurgy, simulating linear and non-linear processes that we encounter in steelmaking by creating mathematical models is of great importance. Since first attempts to use mathematical techniques for the simulation and optimization of large scale metallurgical operations (Ray et al.; 1973), various numerical methods were implemented as algorithms and used to simulate phenomena in steelmaking processes. One class of such methods is Monte Carlo, which is useful for simulating systems with many coupled degrees of freedom such as fluids.

Modern fluid mechanics problems would be impossible to solve without use of Computational Fluid Dynamics (CFD), since the scope of analytical solutions to fundamental equations of fluid mechanics is very limited and, once a more difficult geometry is encountered, we usually have to choose a given numerical method for obtaining a solution. CFD encompasses a wide spectrum of numerical methods used for solving complex three-dimensional (3D) and time-dependent flow problems (Rapp; 2017). Since early pioneering work in the metallurgical field done by Szekely et al. (1977), the cost of performing computer simulations has decreased over the last few decades, while the available processing power has increased. Most of the processors and processing units that are currently developed and produced have several cores that can execute instructions in parallel. Thus, the processing power available to a CFD software also depends on the capability of the software to execute in parallel. A study by Ersson and Tilliander (2018) of the last two decades of metallurgical CFD simulations reveals huge improvements on the type of phenomena that can be explored and we will see this trend is continuing thanks to improvements in both the available processing power and the avail- able algorithms. Therefore CFD found its way into numerous studies in steelmaking, where these methods proved useful in demonstrating the hidden and significant proper- ties. However, its use in the steel industry may not be as integrated as in the aero and automotive industries, in which the development of new designs is of key importance. The major difference between aero and metallurgical industries is that the metallurgical industries almost always deal with multiphase systems at elevated temperatures and that the motivation of modeling is mainly process optimization. With a continuing development in multiphase models as well as in reacting flow modeling, the continued usefulness of CFD in metallurgy remains clear.

In LD/BOF process, different chemical reactions among oxygen, slag, and molten iron in oxygen converter, in combination with vigorous stirring process to promote slagging, dephosphorization, decarbonization, heating of molten steel, and homogenization of steel composition and temperature, determines the final properties of steel. The objective of the oxygen converter is to refine molten iron to crude steel through oxidization to achieve a specified temperature and chemical composition at the end blow. Failure to do this leads to the need to reblow. The impact of oxygen jet into molten bath strongly affects the bath and promotes the three-phase flow among gas, slag, and molten steel in the bath. With the move from old rule-based systems to a model-based, real-time closed-loop control of lance movement and oxygen flow, significant drawbacks were eliminated. There have been efforts in developing accurate and efficient numerical models within CFD field to solve the jets flow in the oxygen converter. Peng and Han (1996) established the conditions of optimum nozzles of performance by deriving the system of mathematical equations to simulate the steady, quasi-one-dimensional supersonic flow through a single De Laval nozzle. Tago and Higuchi (2003) analyzed single-nozzle and multi-nozzle lances with the help of two-dimensional simulation based on fluid dynamics and found that higher ambient temperature leads to the lower density and the higher velocity of the gas jets, but does little affect the dynamic pressure. They report that CFD proved useful method to predict the effect of the inclination angle and the number of the nozzles on the jet behavior in the top blown processes. Wang, Yuan, Matsuura, Zhao, Cheng and Tsukihashi (2010) developed a three-dimensional mathematical model to simulate the compressible jets flow from the top-blown lance, taking into consideration variations of fluid density, viscosity, high temperature and Mach number. They demonstrated that

k–ω turbulence model is superior to the widely used k–ε turbulence model to calculate turbulent conditions within multiple jets.

CFD models have been also used in developing deeper understanding of the decarburization processes in steelmaking. However, these processes are highly complex with large variations in time and length, and therefore it makes the systems extremely demanding to simulate. Ersson and Tilliander (2018) reviewed latest research on the subject from 1998 until 2016 and found out that, even though several reports have been published discussing research about modeling parts of the decarburization processes numerically, no models have been presented that can handle the entire complexity of the processes. Many authors had simplified the system in existing models in order to achieve an understanding of particular phenomena rather than of the entire process.

Another important part of the oxygen steelmaking process is keeping the usual balance of 80% hot metal and 20% scrap during charging to regulate the temperature of steel in the vessel. To define the charge conditions and oxygen blowing requirements to achieve the temperature and chemical composition, mathematical and thermodynamic models have been developed (Laciak et al., 2017; **?**). Reactions that take place in LD process can vary significantly from heat to heat, while not many variables involved are not accurately known. Therefore, it is necessary to take into account the uncertainty affecting the whole process reactions. To correct the differences between the theoretical predictions of the process models and the real results, Bouhouche et al. (2012) introduced a random quantity term into their models and improved the prediction model with the use of Support Vector Regression and Monte Carlo Simulation methods in combination. Most of the control schemes rely on an accurate system model. However, as these systems become more complex, writing down the dynamics from the first principles is extremely challenging. In such cases, neural networks are used to approximate the dynamics directly using system data. In this context, neural networks can be thought of as a generalization of linear regression for non-linear dynamics. At the Institute of Control and Informatization of Production Processes at BERG Faculty (TUKE), team around Laciak et al. (2018)

built upon Bouhouche's work and started experimenting with machine learning in process control and its application in oxygen steelmaking, precisely in LD converter. They applied Support Vector Machines (SVM) and Support Vector Regression (SVR) to predict the final melt temperature and final carbon concentration based on dynamical data. Their work also focuses on developing innovative fractional-order mathematical models for indirect measurement of molten steel temperature and concentration of CO and CO2. The non-linear nature of these processes presents the opportunity to model them by using derivatives of non-integer order, which in their definition are based on the influence of past data on the present value of derivative.

3.1 Interactive Simulation

A real-time simulation platform is a manifestation of this concept, allowing run-time manipulation of geometrical and physical simulation variables. This enables users to rapidly and intuitively investigate different scenarios and design configurations. Ul- timately, a complete interactive simulation package would be capable of simulating a multi-physics 3D environment in real-time to an application-appropriate degree of accu- racy. However, significant inter-disciplinary research is required in reduced-order physical modelling, numerical methods, and software integration to realise a solution (Harwood et al., n.d.).

In order to achieve real-time flow simulation, numerical methods need to be selected carefully such that they can make full use of the capabilities of accelerated computing hardware. Our work focuses on the use of the lattice-Boltzmann Method (LBM):2 a CFD method ideally suited to acceleration on GPUs due to its spatial and temporal locality. GPU-LBM simulations have extremely high computational throughput compared with traditional CFD methods.

general purpose GPU (GPGPU) computing Despite the growing list of success stories, GPU software development adoption has had a slow rise. The slowness of the rise is attributable to the difficulty in programming GPUs (Malcolm et al., 2012).

4 Lattice Boltzmann Method

LB method has witnessed an astonishing growth in its methodology development and application over the past quarter of a century. It fills a vital gap between the macroscopic continuum approaches such as the Navier–Stokes solvers and the particle-based microscopic approaches such as molecular dynamics. Such a mesoscopic approach has found applications in almost all areas of energy and combustion Li et al. (2016).

Real-time fluid simulation, i.e. the ability to simulate a virtual system as fast as the real system would evolve, can benefit to many engineering application such as the optimisation of the ventilation system design in data centres or the simulation of pollutant transport in hospitals. And although real-time fluid simulation is an active field of research in computer graphics, these are generally focused on creating visually appealing animation rather than aiming for physical accuracy. The approach taken for this thesis is different as it starts from a physics based model, the lattice Boltzmann method, and takes advantage of the computational power of a graphics processing unit (GPU) to achieve real-time compute capability while maintaining good physical accuracy. Delbosc (n.d.).

turbulence modelling using the Smagorinsky model in LBM for the simu- lation of high Reynolds number flow and the coupling of two LBM simulations to simulate thermal flows under the Boussinesq approximation.

Interfaces between different phases and/or components are ubiquitous in multiphase flows and energy applications, such as rain dynamics, plant spraying, water boiling, and gas turbine blade cooling, to name but a few. A deeper understanding of the fundamental physics of such complex interfaces is of great importance in many natural and industrial processes. The dynamics of the interfaces is difficult to investigate because typical interfaces are extremely thin, complex in shape, and deforming at short time scales. In addition, the density ratio and Weber and Reynolds numbers involved in many practical multiphase flows, such as binary droplet collisions and melt-jet breakup, are usually very high, which further increases the complexity of the phenomena involved. Therefore, development of robust and accurate computational methods to capture the com- plex interfacial phenomena is crucial in the study of multiphase flows Fei et al. (2019).

During the last three decades, the mesoscopic lattice Boltzmann method (LBM), based on the kinetic theory, has become an increasingly important method for numerical simulations of multiphase flows, mainly on account of its meso-scale features, easy the numerical stability compared with the SRT-LBM. The corresponding non-orthogonal MRT-LBM has been extended to sim

implementation, and computational efficiency. existing multiphase LB models can be classified into four categories: the color-gradient model, the pseudopotential model, the it was shown by Li et al. that a non-orthogonal MRT-LBM free-energy model, and the mean-field model.

Among them, can retain the numerical accuracy while simplifying the implementation of its orthogonal counterpart. In parallel, the CLBM which can be viewed as a non-orthogonal MRT-LBM in the co-moving frame, has been shown to possess very good numerical stability for high Rayleigh number thermal flows,39 as well the pseudopotential model is considered in the present work due to its simplicity and computational efficiency. In this model, the interactions among populations of molecules are modeled by a density-dependent pseudopotential. Through interactions among the particles on the nearest-neighboring sites, phase separation and breakup and/or merging of phase interfaces can be achieved automatically. For further details about the multiphase LB models, interested readers are directed to some comprehensive review

4.1 State of The Art

To overcome difficulties of numerical instability in applying the LBM method, the multiple-relaxation-time (MRT) scheme is useful to stabilize the solution and to obtain satisfactory results because the MRT model allows the usage of an independently optimized relaxation-time for each physical process Suga et al. (2015).

5 Methodology

6 Modeling

6.1 Governing Equations

The volume of fluid (VOF) model was used in this simulation. By tracking the volume fraction of each control unit, the VOF model can solve a single momen- tum equation. Thus, it can be used to simulate the fluid flow of two phase or multiphase, and it is typically applied to track the steady-state or transient gas–liquid interface. Each phase in the model has its own volume fraction a. The sum of the volume fraction of each phase in an arbitrary calculation area is 1 Lv et al. (2013).

6.2 Turbulence Modeling

6.3 Physical Model

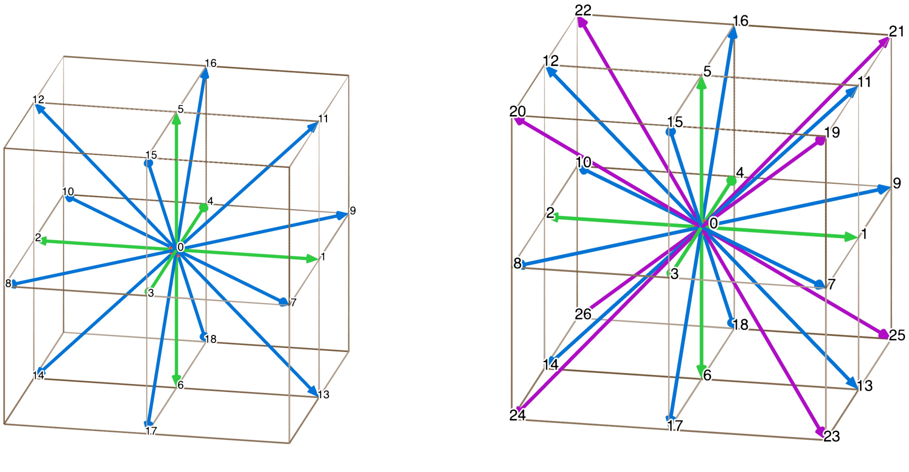
For the convenience of grid generation, the geometry of the converter in steelmaking was simplified to be cylindrical. To substitute for the converter, the cylinder with a radius of 2.0 m was used, in which the calculated depth of molten pool was 1.2 m when the mass of molten steel was 100 t. In consideration of the geometric symmetry of the physical model with a four-hole oxygen lance, one eighth of the model as shown in Figure1 was researched.

7 Lattice Boltzmann Method

The LBE is obtained by discretizing the velocity space of the Boltzmann equation into a finite number of discrete velocities e_α $\{\alpha = 0,1,...,26\}$. With a proper set of discrete velocities, the LBE recovers the continuum Navier–Stokes equations by the Chapman–Enskog expansion. (It is also true for the MRT schemes.) Although many schemes to discretize the velocity space have been proposed, for three-dimensional (3-D) flows, the present study focuses on so called the three- dimensional twenty-seven (D3Q27) discrete velocity model which is illustrated in Fig. 1. Table 1 lists the sound speed c_S , the discrete velocity e_α and the weight parameter w_α in the model with $c = \delta x/\delta t$ where δx and δt are the lattice spacing and the time step, respectively. The MRT LBM transforms the distribution function in the velocity space to the moment space by the transformation matrix M . Since the moments of the distribution function correspond directly to flow quantities, the moment representation allows us to perform the relaxation processes with different relaxation-times according to different time-scales of various physical processes. The evolution equation for the particle distribution function f is thus written as

$$|f(x_i+e_\alpha\delta t, t+\delta t)\rangle - |f(x_i,t)\rangle = -M^{-1}S[|m(x_i,t)\rangle - |m^{eq}(x_i,t)\rangle] + |F(x_i,t)\rangle. \quad (7.1)$$

where x_i is the position vector of node i, $\vec{S}$ is the collision matrix, m is the moment, F represents an external body force and the notation for the column vector (known as the ket vector) such as $|f\rangle$ represents



(a) D3Q19 (b) D3Q27

Fig. 7 – 1: Three-dimensional lattice node schemes.

$$\boldsymbol{m} = [m_0, m_1, ..., m_{26}]^T \quad (7.2)$$

For better numerical stability, Multiple Relaxation Time (MRT) scheme is used. It allows for more degrees of freedom and better tunability of relaxation parameters. Stability is a key property in any numerical scheme Succi (2001). It helps to protect against cumulative error build-up or other sources of inaccuracies.

Caution have to be taken when working within lattice's discrete world. For simulating the interface of blown oxygen with melted fluid slag, we're working with higher Mach speeds. The basic notion is that the lattice can only support signals with a finite propagation speed. Necessary criterion for stability is that physical information should not travel faster than fastest speed supported by the lattice Succi (2001).

We can calculate the error to $\varepsilon(Ma^3)$ in space and proportional to $\varepsilon(Ma \cdot dt)$ in time, where $Ma = \frac{u}{c_s}$ is the Mach number of the system.

$p = c_s^2\rho$ is the pressure, $c_s = \frac{c}{\sqrt{3}}$ is the speed of sound, and the kinematic and viscosity ν is related to the relaxation time rates for the second-order moments by $\nu = \left(\frac{1}{s_\nu - 0.5}\right) c_s^2 \Delta t$ and $\xi = \frac{2}{3}\left(\frac{1}{s_b - 0.5}\right)$ respectively.

Transformation matrix $\boldsymbol{M}$ can be obtained from Eqs. 7.2.

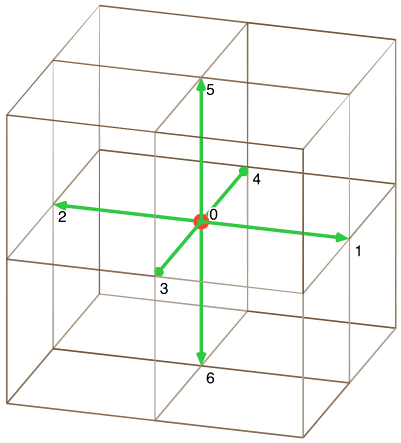


Fig. 7 – 2: D3Q6 lattice node schemes

8 Simulation Software

9 GPU Computing

GPU-based parallel computing reduces the time for some computing tasks by orders of magnitude, so big computations (like training a machine learning system on a large data set) that took weeks can now happen in hours, and moderate-sized computations (like producing a 3D contour plot) that took minutes can happen interactively in real time Storti and Yurtoglu (2015). These gains can be achieved at very reasonable costs in terms of both developer effort and the hardware required.

With the power of GPU computing, it's now possible to speed-up scientific computations in a way that what took minutes or hours can now be done in seconds or even milliseconds. Interesting case of engineering problem for which GPU computing is showing tremendous potential is solving differential equations while changing the initial or boundary conditions in real-time.

9.1 CUDA and OpenCL

CUDA is a hardware/software platform for parallel computing created and supported by NVIDIA Corporation to promote access to high-performance parallel computing. The hardware aspect of CUDA involves graphics cards equipped with one or more CUDA-enabled graphics processing units (GPUs). The NVIDIA CUDA Toolkit software provides a development environment based on the C/C++ family of programming languages.

Storti and Yurtoglu (2015)

The GPU-based approach to massively parallel computing used by CUDA is also the core technology used in many of the world's fastest and most energy- efficient supercomputers. The key criterion has evolved from FLOPS (floating point operations per second) to FLOPS/watt (i.e., the ratio of computing productivity to energy consumption), and GPU-based parallelism competes well in terms of FLOPS/watt (Storti and Yurtoglu, 2015).

(Karimi, n.d.)

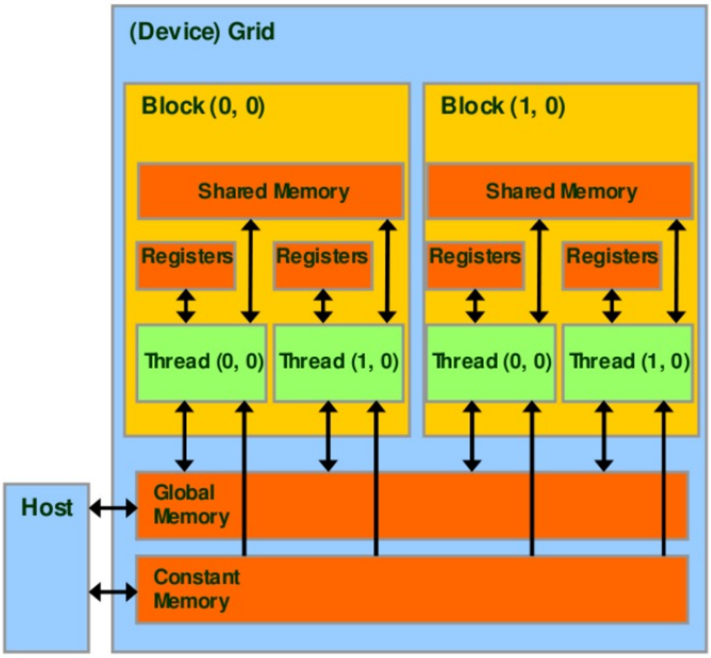


Fig. 9 – 1: CUDA memory model

... created a library to simplify programming for the GPU’s.

(Yalamanchili et al., 2015)

9.2 ArrayFire

ArrayFire is a high performance software library for parallel computing with an easy-to-use API. ArrayFire abstracts away much of the details of programming parallel architectures by providing a high-level container object, the Array, that represents data stored on a CPU, GPU, FPGA, or other type of accelerator. This abstraction permits developers to write massively parallel applications in a high-level language where they need not be concerned about low-level optimizations that are frequently required to achieve high throughput on most parallel architectures.

(Yalamanchili et al., 2015)

Going on from ArrayFire's version 3.2., the Unified backend was introduced. It allows developers to change between different backends CUDA, OpenCL and CPU

$$\begin{bmatrix} 1 & 2 & 3 \\ a & b & c \end{bmatrix} \tag{9.1}$$

9.2.1 Comment on GFOR

- TODO: mozno presunut do Appendixu?

ArrayFire provides the only GPU-enabled data-parallel loop: $gfor$. Use $gfor$ in your code in place of standard for-loops to batch process data parallel operations (Malcolm et al., 2012). You can think of gfor as performing auto-vectorization of your code, e.g. you write a gfor-loop that increments every element of a vector but behind the scenes ArrayFire rewrites it to operate on the entire vector in parallel (Yalamanchili et al., 2015). For example, the following for-loop calculates several matrix multiplications serially:

```
for (int i = 0; i < n; ++i) {
  A(i) = A(i) + 1;
}
```

Algorithm 1: Pseudo-code with imerative for loop

The same three matrix multiplications can be carried out in one pass instead of three by utilizing a gfor-loop,

```
A = constant(1,n,n);
gfor (seq i, n) {
  A(i) = A(i) + 1;
}
```

Algorithm 2: Pseudo-code with if-statement removed

Whereas the former loop serially computes each matrix multiplication, the latter loop computes all matrix multiplications in one pass. Similarly, gfor can be used for other such embarrassingly parallel codes in a straightforward fashion. A good way of thinking about gfor is to consider it as syntactic sugar: gfor serves as an iterative style of writing otherwise vectorized algorithms (Malcolm et al., 2012).

9.3 Optimizations for GPU Computing

9.3.1 Removing branch divergence

```
if (node_type == "fluid") {
  x = a;
} else {
  x = b;
}
```

Algorithm 3: Pseudo-code of if-statement in kernel

```
let is_fluid = node_type == "fluid";
x = a * is_fluid + b * (!is_fluid);
```

Algorithm 4: Pseudo-code with if-statement removed

```
use arrayfire as af;

fn main() {
  // dimensions
  let dims = af::dim4!(3,3,1,1);

  // 0 -> wall
  // 1 -> fluid
  // 2 -> gas
  let domain = af::Array::new(&[0,0,0,1,1,1,2,2,2], dims);

  // original array
  let a = af::randu::<f32>(dims);
  // part of the domain to be replaced
  let water = af::constant(1, dims);
  // new values
  let b = af::randu::<f32>(dims);

  let cond = af::eq(&water, &domain, false);

  let out = af::select(&a, &cond, &b);

  // Print the values
  af::af_print!("", &a);
  af::af_print!("", &b);
  af::af_print!("", &out);

  // Values from array ‘a‘
  // 0.6010     0.2126     0.2864
  // 0.0278     0.0655     0.3410
  // 0.9806     0.5497     0.7509

  // Values from array ‘b‘
  // 0.4105     0.3543     0.3636
  // 0.1583     0.6450     0.4165
  // 0.3712     0.9675     0.5814

  // Values from boolean array ‘cond‘
  // 0          1          0
  // 0          1          0
  // 0          1          0

  // Values from array ‘out‘
  // 0.4105     0.2126     0.3636
  // 0.1583     0.0655     0.4165
  // 0.3712     0.5497     0.5814
}
```

Algorithm 5: Example Rust code of removing branch divergence using ArrayFire

To avoid if-statements in kernel code for LBM algorithm, a “ghost layer” is added to the computational domain. In other words, for n_x, n_y and n_z as width, height and depth respectively, $n_x + 1$, $n_y + 1$ and $n_z + 1$ nodes are added to the computational domain’s grid. Ghost layer helps in a way that we can treat computations at the boundary position as the normal ones and not worry about running out of the index bounds Tran et al. (2017).

9.4 Programming Language

The Rust programming language helps you write faster, more reliable software. High-level ergonomics and low-level control are often at odds in programming language design; Rust challenges that conflict. Through balancing powerful technical capacity and a great developer experience, Rust gives you the option to control low-level details (such as memory usage) without all the hassle traditionally associated with such control. Klabnik (2018)

9.4.1 Why I Chose Rust Programming Language

Rust is ideal for many people for a variety of reasons.

When

Rust is for people who crave speed and stability in a language. By speed, we mean the speed of the programs that you can create with Rust and the speed at which Rust lets you write them. The Rust compiler's checks ensure stability through feature additions and refactoring. This is in contrast to the brittle legacy code in languages without these checks, which developers are often afraid to modify. By striving for zero-cost abstractions, higher-level features that compile to lower-level code as fast as code written manually, Rust endeavors to make safe code be fast code as well. Klabnik (2018)

The Rust language hopes to support many other users as well; those mentioned here are merely some of the biggest stakeholders. Overall, Rust's greatest ambition is to eliminate the trade-offs that programmers have accepted for decades by providing safety and productivity, speed and ergonomics. Give Rust a try and see if its choices work for you. Klabnik (2018)

It wasn't always so clear, but the Rust programming language is fundamentally about empowerment: no matter what kind of code you are writing now, Rust empowers you to reach farther, to program with confidence in a wider variety of domains than you did before.

Take, for example, "systems-level" work that deals with low-level details of memory management, data representation, and concurrency. Traditionally, this realm of programming is seen as arcane, accessible only to a select few who have devoted the necessary years learning to avoid its infamous pitfalls. And even those who practice it do so with caution, lest their code be open to exploits, crashes, or corruption.

Rust breaks down these barriers by eliminating the old pitfalls and providing a friendly, polished set of tools to help you along the way. Programmers who need to "dip down" into lower-level control can do so with Rust, without taking on the customary risk of crashes or security holes, and without having to learn the fine points of a fickle toolchain. Better yet, the language is designed to guide you naturally towards reliable code that is efficient in terms of speed and memory usage.

Programmers who are already working with low-level code can use Rust to raise their ambitions. For example, introducing parallelism in Rust is a relatively low-risk operation:

the compiler will catch the classical mistakes for you. And you can tackle more aggressive optimizations in your code with the confidence that you won't accidentally introduce crashes or vulnerabilities.

But Rust isn't limited to low-level systems programming. It's expressive and ergonomic enough to make CLI apps, web servers, and many other kinds of code quite pleasant to write — you'll find simple examples of both later in the book. Working with Rust allows you to build skills that transfer from one domain to another; you can learn Rust by writing a web app, then apply those same skills to target your Raspberry Pi.

This book fully embraces the potential of Rust to empower its users. It's a friendly and approachable text intended to help you level up not just your knowledge of Rust, but also your reach and confidence as a programmer in general. So dive in, get ready to learn—and welcome to the Rust community!

Klabnik (2018)

10 Visualization Software

Scientific visualization is the use of computer graphics to create visual images that aid in the understanding of complex numerical representations of scientific concepts or results. Computational fluid dynamics (CFD) based numerical simulations often output massive amounts of data. These simulations often contain high-dimensional data in a three- dimensional volume. The display of phenomena associated with this data may involve complex three-dimensional structures.

10.1 Human-Computer Interface

10.1.1 SCADA

- ako sa scada systemy pouzivaju

- neexistuje prepojenie s virtualnou realitou

- future work co by sa dalo spravit

10.2 Visualization Tools

10.2.1 The Visualization Toolkit

10.2.2 ParaView

10.3 Virtual and Augmented Reality

Non-immersive interactive visualization systems implemented for the conventional desktop and mouse are effective for moderately complex problems. Immersive virtual environments, by comparison, lie at the other end of the spectrum and permit looking around an object by moving one's head position.

Therefore, a fun- damental difference between desktop-and-mouse virtual realia and immersive VR is that the latter is a true 3D representation that may be either viewer or object-centered while the first is exclusively viewer-centered. In other words, changes in the relative positions of a 2D object's components result from shifts in the viewer's perspective. The same may be true for objects viewed in a three dimensional environment, whether real or vir- tual. However, in such an environment, an object may also appear to change shape (e.g., through foreshortening), not due to an altered position of the viewer, but because the ob- ject itself has moved to a different position. Immersive virtual reality displays aid in the unambiguous display of these structures by providing a rich set of spatial and depth cues. Virtual reality interface concepts allow the rapid and intuitive exploration of the volume containing the data, enabling the phenomena at various places in the volume to be ex- plored, as well as provide simple control of the visualization environment through inter- faces integrated into the environment (Bryson; 1996).

Desktop-and-mouse interfaces for 3D visualizations make it difficult to specify positions in three dimensions and do not provide unambiguous display of 3D structure. Virtual reality interfaces attempt to provide the most anthropomorphic interfaces possible - that means they must be human-conforming and should be designed to allow the most natural, unambiguous way of scientific exploration. They must include two components: display and user control. Scientific visualization makes particular demands on virtual reality displays. The phenomena to be displayed in a scientific visualization application often involve delicate and detailed structure, requiring high-quality, high-resolution full-color displays. A wide field of view is often desirable, because it allows the researcher to view how detailed structures are related to larger, more global phenomena.

Historically, the early attempts at using head-mounted virtual reality technologies started with CRT-based Binocular Omni-Oriented Monitor (BOOM) created by Fakespace Systems Inc. BOOM was a stereoscopic display device with screens and optical system housed in a box that is attached to a multi-link arm. Head tracking was accomplished via sensors in the links of the arm that holds the box.

Advent of commodity-level VR hardware like HTC Vive or Oculus Touch has made this technology accessible for meaningful applications. These headset utilize lasers and photosensitive sensors (HTC Vive) or cameras (Oculus Touch) for head and hands tracking and provide six degrees of freedom (6DoF) for movement in virtual environment. By immersing the user into the simulation itself, virtual reality reveals the spatially complex structures in computational science in a way that makes them easy to understand and study. But beyond adding a 3D interface, virtual reality also means greater computational complexity (Bryson; 1996). The ability to provide real-time interaction can provide strong depth cues, either through allowing interactive rotations or through the use of head-tracked rendering. Applications and techniques are being developed to discern how immersive technology benefits visualization. The medical field provides an especially promising context for this development, as medical practitioners require a thorough understanding of specific 3D structures: human anatomy. Users may interact simultaneously with high resolution computed tomography (CT) scans and their corresponding, 3D anatomical structures.

Another frequently used type of immersive, interactive display technology nowadays is projection-screen-based Cave Automatic Virtual Environment (CAVE). These systems consists of 3 to 6 large displays positioned into a room-sized cube around the observer. The walls of a CAVE are typically made up of rear-projection screens, but recently the flat panel displays are commonly used. The floor can be a downward-projection screen, a bottom projected screen or a flat panel display. The projection systems are very high-resolution due to the near distance viewing which requires very small pixel sizes to retain the illusion of reality. The user wears 3D glasses inside the CAVE to see 3D graphics generated by the CAVE. People using the CAVE can see objects apparently floating in the air, and can walk around them, getting a proper view of what they would look like in reality. This is made possible by infrared cameras. Movement of the observer in the CAVE is tracked by the sensors typically attached to the 3D glasses and the video continually adjusts to retain the viewers perspective.

Many universities and engineering companies own and use CAVE systems. Researchers can use these systems to conduct their research topic in a more effective and accessible method. Engineers have found them useful in enhancing of a product development through prototyping and testing phases.

In field of mathematics, VR application named Cal (2019) is making serious progress. It is developed by a company Nanome, Inc. started by students from University of California San Diego. Team behind Calcflow is using VR to help students grasp the biggest ideas in vector calculus. Its features include visualizations of vector addition, cross product, parametrized functions, spherical coordinate mapping, double and surface integrals. Beside Calcflow, they are implementing a VR platform specialized for atomic, molecular and protein visualization, built for researchers and scientists (Nan; 2019).

One can say that virtual reality established itself in many disciplines of human activities, as a medium that allows easier perception of data or natural phenomena appearance. In fact, theme of this dissertation was influenced by my previous work with using virtual reality for mathematics education at the university TODO: CITOVAT MATHWORLDVR CLANOK!!!

10.3.1 Virtual Reality in Steelmaking

Substantial amount of work in applying 3D visualizations and virtual reality for solving technological issues and bringing new trends into steelmaking industry is currently happening at Center for Innovation through Visualization and Simulation (CIVS) at Purdue University Northwest (located in Indiana, USA). CIVS has been globally recognized for its integrated and application-driven approaches through state-of-the-art simulation and virtual reality visualization technologies for providing innovative solutions to solve various university research problems, industry issues, as well as education. More than 350 projects that have been completed at the center from its inception in 2014 until today provided substantial educational and economic impact, resulting in more than 40 million US dollars (more than 36.1 million e at the time of this writing) in savings for companies. In collaboration with other universities and companies from steelmaking industry, they focus on research regarding integration of virtual reality with simulation technologies and high performance computing; application of simulation and visualization technologies to industrial processes for process design trouble-shooting and optimization to address the issues of productivity, energy, environment, and quality; and last but not least, development of advanced learning environments in virtual reality for training and education. They launched novel, industry-led association of steel manufacturers and stakeholders called Steel Manufacturing Simulation and Visualization Consortium (SMSVC).

Interesting application of combining 3D CFD smulation and virtual reality for visual inspection is pulverized coal injection (PCI) and coke combustion model. Research efforts between the Canadian government (CANMET), CIVS and the American Iron and Steel Institute (AISI) were conducted and resulted in modeling of the blowpipe and tuyere of the blast furnace. Combination of aforementioned technologies turned out to be powerful and provided detailed information of flow streams that were previously very difficult to measure. The CFD model shown in Fig. 2 – 12 was used to simulate PCI with natural gas co-injection in the lance, blowpipe and tuyere TODO: DOPLNIT CITACIU!!!.

Another very interesting project conducted at CIVS involved development of comprehensive package of modules for simulating multiple processes in blast furnace. 3D CFD model shown in Fig. 2 – 13 has been developed by Zheng and Hu (2014) specifically for simulating the blast furnace hearth. The campaign life of a blast furnace is highly dependent on residual thickness of refractory lining in the hearth. The progress of hearth lining erosion is greatly affected by hot metal flow patterns and heat transfer in refractory under different operating conditions. CFD model incorporates both the hot metal flow and conjugate heat transfer through the refractories. They achieved consistency of results between measured and calculated refractory temperature profiles, as the model has been extensively validated using measurement data from industry blast furnace. The virtual reality (VR) visualization technology has been used to analyze the velocity and temperature distributions and wear patterns of different furnaces and operating conditions. This interactive 3D visualization is shown in Fig. 2 – 14. Based on the results, it was possible to predict the inner profile of hearth and provide guidance to protecting the blast furnace hearth.

10.4 Visualizing Simulation in Virtual Reality

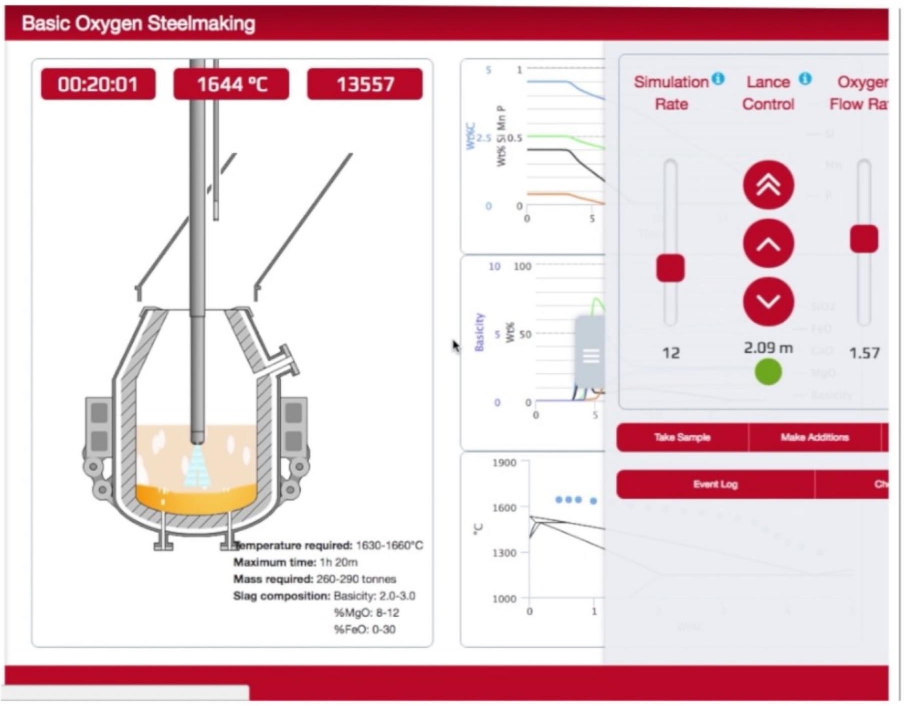
11 Results

12 Conclusions

13 Discussion

14 Appendix A: Simulation as an Educational Tool

In some simplified form, combinations of numerical simulations and visualizations of steel- making processes can be used as a educational tool in process control courses at technical universities. The aim of the online, web-based interactive simulation of basic oxygen steelmaking at steeluniversity.org shown in Fig. **??** is to introduce students to this process in a more fun and engaging way.



Temperature required: 1630-1660°C
Maximum time: 1h 20m
Mass required: 260-290 tonnes
Slag composition: Basicity: 2.0-3.0
%MgO: 8-12
%FeO: 0-30

Simulation Rate 12
Lance Control 2.09 m
Oxyger Flow Ra 1.57

Fig. 14 – 1: Interactive, educational simulation of basic oxygen steelmaking by steeluniversity.org.

Bibliography

Delbosc, N. (n.d.). Real-Time Simulation of Indoor Air Flow Using the Lattice Boltzmann Method on Graphics Processing Unit, p. 261.

Fei, L., Du, J., Luo, K. H., Succi, S., Lauricella, M., Montessori, A. and Wang, Q. (2019). Modeling realistic multiphase flows using a non-orthogonal multiple-relaxation-time lattice Boltzmann method, *Physics of Fluids* **31**(4): 042105.
URL: *http://aip.scitation.org/doi/10.1063/1.5087266*

Harwood, A. R. G., Wenisch, P. and Revell, A. J. (n.d.). A REAL-TIME MODELLING AND SIMULATION PLATFORM FOR VIRTUAL ENGINEERING DESIGN AND ANALYSIS, p. 11.

Jalkanen, H. (2006). Experiences in physicochemical modelling of oxygen converter process (BOF), *2006 TMS Fall Extraction and Processing Division: Sohn International Symposium* **2**: 541–554.

Karimi, K. (n.d.). A Performance Comparison of CUDA and OpenCL, p. 10.

Klabnik, S. (2018). *The Rust Programming Language*, No Starch Press.
URL: *https://www.xarg.org/ref/a/B071YKRV8Q/*

Laciak, M., Petráš, I., Flegner, P., Durdán, M. and Tréfa, G. (2017). Výskum nepriameho merania teploty a uhlíka v procese skujňovania, *Technical report*, Fakulta BERG Ústav riadenia a informatizácie výrobných procesov.

Li, Q., Luo, K., Kang, Q., He, Y., Chen, Q. and Liu, Q. (2016). Lattice Boltzmann methods for multiphase flow and phase-change heat transfer, *Progress in Energy and Combustion Science* **52**: 62–105.
URL: *https://linkinghub.elsevier.com/retrieve/pii/S0360128515300162*

Lv, M., Zhu, R., Guo, Y.-G. and Wang, Y.-W. (2013). Simulation of Flow Fluid in the BOF Steelmaking Process, *Metallurgical and Materials Transactions B* **44**(6): 1560–

1571.
URL: *http://link.springer.com/10.1007/s11663-013-9935-4*

Malcolm, J., Yalamanchili, P., McClanahan, C., Venugopalakrishnan, V., Patel, K. and Melonakos, J. (2012). ArrayFire: A GPU acceleration platform, *in* E. J. Kelmelis (ed.), *SPIE Defense, Security, and Sensing*, Baltimore, Maryland, USA, p. 84030A.

Storti, D. and Yurtoglu, M. (2015). *CUDA for Engineers: An Introduction to High-Performance Parallel Computing*, Addison-Wesley Professional.
URL: *http://cudaforengineers.com/*

Succi, S. (2001). *The lattice boltzmann equation: For fluid dynamics and beyond*, Numerical mathematics and scientific computation, Clarendon Press. tex.lccn: 2001036220.
URL: *https://books.google.hu/books?id=OC0Sj_xgnhAC*

Suga, K., Kuwata, Y., Takashima, K. and Chikasue, R. (2015). A D3Q27 multiple-relaxation-time lattice Boltzmann method for turbulent flows, *Computers & Mathematics with Applications* **69**(6): 518–529.
URL: *https://linkinghub.elsevier.com/retrieve/pii/S0898122115000346*

Tran, N.-P., Lee, M. and Hong, S. (2017). Performance Optimization of 3D Lattice Boltzmann Flow Solver on a GPU, *Scientific Programming* **2017**: 1–16.
URL: *https://www.hindawi.com/journals/sp/2017/1205892/*

Turkdogan, E. T. (1996). *Fundamentals of Steelmaking*, The Institute of Materials, London.

Wang, X., Han, M. and Wang, J. (2010). Applying input variables selection technique on input weighted support vector machine modeling for BOF endpoint prediction, *Engineering Applications of Artificial Intelligence* **23**(6): 1012–1018.
URL: *http://dx.doi.org/10.1016/j.engappai.2009.12.007*

Widlund, D., Medvedev, A. and Gyllenram, R. (1998). Towards Model-Based Closed-Loop Control of the Basic Oxygen Steelmaking Process, *IFAC Proceedings Volumes*

31(23): 69–74.
URL: *http://dx.doi.org/10.1016/S1474-6670(17)35858-5*

Yalamanchili, P., Arshad, U., Mohammed, Z., Garigipati, P., Entschev, P., Kloppenborg, B., Malcolm, J. and Melonakos, J. (2015). ArrayFire - A high performance software library for parallel computing with an easy-to-use API.
URL: *https://github.com/arrayfire/arrayfire*