

THE BIOREMEDIATION OF ACID MINE DRAINAGE UTILISING INDIGENOUS SOUTH AFRICAN GRASS AS THE ORGANIC CARBON SOURCE FOR DISSIMILATORY SULFATE REDUCTION



Natalie Greenway

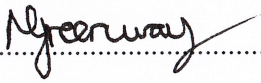
360352

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Johannesburg, 2019

DECLARATION

I declare that this dissertation is my own unaided work. It is being submitted to the Degree of Master of Science in Engineering to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other University.



(Signature of Candidate)

.....28.....day of.....08.....year.....2019.....

ABSTRACT

Acid Mine Drainage (AMD) is an acidic sulfate-rich wastewater that contains a high concentration of heavy metals. Due to historic and ongoing mining activities around the Witwatersrand, large quantities of AMD is generated in this area which seeps into surface and ground water. AMD has negative environmental, health and economic impacts and requires treatment. Passive treatment is a cost-effective option for the remediation of AMD and involves the process of dissimilatory sulfate reduction (DSR) using sulfate reducing bacteria (SRB) and an organic carbon food source. Indigenous South African grass was identified as an inexpensive and suitable organic carbon source for the bioremediation of AMD.

The research presented in this thesis identified a one-stage leach bed reactor (LBR), packed with indigenous South African grass, as a suitable and cost-effective reactor design and configuration for the treatment of AMD. A laboratory-scale LBR was constructed and an experimental analysis determined that an optimal ratio of indigenous South African grass to a synthetic AMD leachate was 6 g of grass / L of synthetic AMD. At higher solid/liquid ratios the system had higher sulfate removal efficiencies.

The experiment was conducted using a synthetic AMD leachate and an AMD leachate in order to evaluate the performance of the designed process. The bioreactor containing a synthetic AMD leachate had an average sulfate removal efficiency of 59% and a dissolved iron removal efficiency of 65%. A black coating of iron (II) sulphide formed along the sides of the LBR and on the grass. The bioreactor containing an AMD leachate had an average removal efficiency of 28% and 31% of sulfates and dissolved irons respectively. The pH remained acidic throughout the course of the experiment. AMD has a complex chemistry, including a high concentration of various heavy metals, and has a high mineral and total acidity. A required pre-treatment step of alkaline dosing may be required. An investigation should be made into using a sequentially fed LBR coupled with an upflow anaerobic sludge bed (UASB) in order to increase sulfate and dissolved iron removal efficiencies.

A model of the one-stage LBR was developed and tested against the experimental results of synthetic AMD leachate and the AMD leachate. This research used the Anaerobic Digestion Model No.1 (ADM1) as the basis for the model and upgraded it to include the process of DSR and iron transformations. The model was able to qualitatively predict the behaviour of the system (10% < mean average percent error <30%) but there were discrepancies in predicting the dissolved iron concentrations. Further work should be conducted in upgrading the model to account for transformations and the weak acid-base chemistry of various heavy metals present in AMD.

ACKNOWLEDGEMENTS

I would like to thank my supervisors Professor Lizelle van Dyk and Professor Craig Sheridan for their guidance and support throughout my postgraduate studies.

I would like to thank my peers, including Nimat Fakir and Ebrahim Lambat, for their help and assistance in the laboratory.

I would like to thank Ulf Jeppeson, Krist V. Gernaey, Xavier Flores-Alsina and Kimberly Solon for their guidance on modelling and implementing the Anaerobic Digestion Model No.1.

I would like to thank Maria Heyneke for encouraging me to challenge myself and achieve more.

I would like to thank the three musketeers for their constant love.

I would like to thank William Kane for always reminding me that I am braver than I believe, stronger than I seem and smarter than I think.

I would like to thank my beautiful mother, Colette Greenway, for all the love and support you have given me. You have been my pillar of strength. I am so proud to call you my mother.

CONTENTS

Declaration.....	ii
Abstract.....	iii
Acknowledgements.....	iv
List of Figures	vii
List of Tables	x
List of Symbols	xii
Nomenclature.....	xiv
1 Introduction.....	15
1.1 Problem Statement.....	16
1.2 Research Objectives.....	16
1.3 Dissertation Layout.....	17
2 Literature Review.....	19
2.1 The generation of acid mine drainage (AMD)	19
2.2 The impact of acid mine drainage (AMD)	20
2.3 The treatment of acid mine drainage (AMD)	24
2.4 The selection of Indigenous South African grass as the organic carbon source for the bioremediation of acid mine drainage (AMD)	27
2.5 Process selection and design configuration	28
2.6 Anaerobic digestion model No.1 (ADM1)	30
3 Experimental Methodology	36
3.1 Experimental and apparatus set-up.....	36
3.2 Operational scheme	39
3.3 Materials and methods.....	39
3.4 Analytical methods	41
3.5 Modelling methods	45

4	Results and Discussion	54
4.1	Optimal Solid/Liquid Ratio for Anaerobic Digestion and Increased Sulfate Removal Efficiencies	54
4.2	Model Results	72
4.3	Model Testing	74
5	Conclusion and recommendations	84
6	References	87
	Appendix A The Upgraded Anaerobic Digestion Model No. 1 (ADM1)	95
	Appendix B The Chemical Composition of Acid Mine Drainage	105
	Appendix C Mathematical modelling in MATLAB® and Simulink®	106
	Appendix D Raw Experimental Data	212
	Appendix E Publications	219

LIST OF FIGURES

Figure 2.1 'Yellow boy' (Liefferink, 2009).....	21
Figure 2.2 A dam polluted with AMD in the Kurgersdorp Game Reserve (Liefferink, 2009).....	22
Figure 2.3 A graphical representation of the extracellular and biochemical process, and the relevant products of the ADM1 (Batstone et al., 2002)	31
Figure 2.4 Determination of substrate composition by extended Weender analysis (Koch et al., 2010)	34
Figure 3.1 Reactor set-up of a one-stage LBR.....	36
Figure 3.2 Laboratory scale leach bed reactor (LBR).....	37
Figure 3.3 Top view and cross-sectional view of the sieve	37
Figure 3.4 Spray nozzle head.....	38
Figure 3.5 Final laboratory-scale reactor configuration	38
Figure 3.6 The anaerobic model of the extended ADM1 to include sulfate reduction adapted from (Batstone et al., 2002) and (Barrera et al., 2013).....	47
Figure 4.1 pH vs time for the solid/liquid ratios 1.5, 6 and 30	55
Figure 4.2 Sulfate concentration for the solid/liquid ratios 1.5, 6 and 30.....	57
Figure 4.3 The percentage average sulfate removed for solid/liquid ratios of 1.5, 6 and 30.....	57
Figure 4.4 The relationship between the removal of Sulfate and the production of COD	58
Figure 4.5 Soluble COD produced per gram of grass for the solid to liquid ratio of 1.5, 6 and 30.....	59
Figure 4.6 SEM analyses of grass, before and after the experiment was conducted, for the solid/liquid ratios of 1.5 at a magnification of 100x (a-b), 500x (c-d), 1000x (e-f), 200x (g-h) and 5000x (i-j).	61

Figure 4.7 SEM analyses of grass, before and after the experiment was conducted, for the solid/liquid ratios of 6 at a magnification of 100x (a-b), 500x (c-d), 1000x (e-f), 200x (g-h) and 5000x (i-j).	62
Figure 4.8 Averaged pH vs time for AMD and Synthetic AMD.....	63
Figure 4.9 Averaged concentration of COD produced for AMD and Synthetic AMD	65
Figure 4.10 Averaged sugar concentration for AMD and Synthetic AMD	66
Figure 4.11 Averaged concentration of sulfate for AMD and Synthetic AMD.....	66
Figure 4.12 Percentage of average sulfate removed	67
Figure 4.13 Averaged dissolved sulfide concentration for AMD and Synthetic AMD	68
Figure 4.14 Percentage average dissolved sulfide produced	68
Figure 4.15 Averaged dissolved iron concentration for AMD and Synthetic AMD	69
Figure 4.16 Averaged total iron concentration for AMD and Synthetic AMD	70
Figure 4.17 Percentage of average iron removed	70
Figure 4.18 A visual comparison of the production of iron (II) sulfide between the reactor containing and AMD leachate (left) and the reactor containing synthetic AMD (right).....	71
Figure 4.19 Soluble COD synthetic AMD experimental concentration vs simulated concentration.....	75
Figure 4.20 Soluble COD AMD experimental concentration vs simulated concentration.....	76
Figure 4.21 Sugar experimental concentration produced by the synthetic AMD vs simulated concentration	77
Figure 4.22 Sugar experimental concentration produced by the AMD vs simulated concentration.....	77
Figure 4.23 Simulated pH vs synthetic AMD pH.....	78
Figure 4.24 Simulated pH vs AMD pH	78

Figure 4.25 Sulfate experimental concentration produced by the synthetic AMD vs simulated concentration	79
Figure 4.26 Sulfate experimental concentration produced by the AMD vs simulated concentration.....	80
Figure 4.27 Sulfide experimental concentration produced by the synthetic AMD vs simulated concentration	81
Figure 4.28 Sulfide experimental concentration produced by the AMD vs simulated concentration.....	81
Figure 4.29 Dissolved iron experimental concentration produced by the synthetic AMD vs simulated concentration	82
Figure 4.30 Dissolved iron experimental concentration produced by the AMD vs simulated concentration	82
Figure 6.1 Simulink Model Page 1	107
Figure 6.2 Simulink Model Page 2 (Leach Bed)	108

LIST OF TABLES

Table 2.1 Effects of Sulfate on Aesthetics and Human Health (Department of Water Affairs and Forestry., 1996).....	23
Table 2.2 Effects of iron on aesthetics, human health and household distribution systems (Department of Water Affairs and Forestry., 1996).....	24
Table 2.3 The Theoretical Oxygen Demand (ThOD) for the different organic fractions found in grass.....	34
Table 3.1 Solid/Liquid Ratios	40
Table 3.2 Composition of the synthetic AMD solution and initial pH for solid/liquid ratios of 1.5 g/L, 6 g/L and 30 g/L	40
Table 3.3 COD concentrations of prepared standards	42
Table 3.4 Sulfate concentrations of prepared standards	42
Table 3.5 Sulfide concentrations of prepared standards	44
Table 3.6 State variables as obtained from the IWA	53
Table 4.1 Percentage of Cations present in grass using a solid/liquid ratio of 6 before and after experiment was conducted.....	56
Table 4.2 Heavy metal concentrations in AMD (ICP analysis).....	64
Table 4.3 Composition of indigenous South African grass determined by the extended Weender analysis.....	72
Table 4.4 Summary of parameters used to describe the characterisation of Indigenous South African grass.....	73
Table 4.5 Initial starting conditions of AMD and Synthetic AMD	74
Table 6.1 Biochemical rate coefficients ($v_{i,j}$) for soluble components	96
Table 6.2 Biochemical rate coefficients ($v_{i,j}$) for particulate components.....	97
Table 6.3 Biochemical rate coefficients ($v_{i,j}$) for S model extension components	98
Table 6.4 Biochemical rate coefficients ($v_{i,j}$) for Fe model extension components.....	99
Table 6.5 Stoichiometric matrix for precipitation process.....	99

Table 6.6 Stoichiometric matrix for liquid-gas transfer process.....	100
Table 6.7 Process kinetic rate equations (ρ_j)	101
Table 6.8 Parameter values for the modified model	103
Table 6.9 The Chemical Composition of Acid Mine Drainage	105
Table 6.10 pH (ratio 1.5).....	212
Table 6.11 pH (ratio 6).....	212
Table 6.12 pH (ratio 30).....	213
Table 6.13 Sulfate Concentration (ratio 1.5)	213
Table 6.14 Sulfate Concentration (ratio 6)	213
Table 6.15 Sulfate Concentration (ratio 30)	214
Table 6.16 COD Concentration (ratio 1.5)	214
Table 6.17 COD Concentration (ratio 6)	215
Table 6.18 COD Concentration (ratio 30)	215
Table 6.19 pH AMD and syn AMD.....	216
Table 6.20 COD Concentration AMD and syn AMD	216
Table 6.21 Sulfate Concentration AMD and syn AMD	216
Table 6.22 Sulfide Concentration AMD and syn AMD	217
Table 6.23 Dissolved Iron Concentration AMD and syn AMD	217
Table 6.24 Total Iron Concentration AMD and syn AMD.....	217
Table 6.25 Sugar Concentration AMD and syn AMD	218

LIST OF SYMBOLS

Symbol	Description
$\partial_{i/j}$	Activity of species <i>i</i> or component <i>j</i>
C	Concentration
D	Biodegradable fraction
f_i	Stoichiometric <i>f</i> -factor
G _{ch4}	Methane gas
G _{co2}	Carbon dioxide gas
G _{h2}	Hydrogen gas
G _{h2s}	Hydrogen sulphide gas
ΔH°	Enthalpy change
I	Inhibition factor
$v_{i,j}$	Biochemical rate coefficients
K _{A/B,i}	Process rate coefficients
k _{cryst,i}	Mineral precipitation rate coefficient
K _H	Henry's constant
K _i	Equilibrium Constant
K _{I,j}	50% Inhibitory concentration of component <i>j</i>
K _{m,j}	Maximum specific uptake of component <i>j</i>
K _{SP,i}	Solubility factor of mineral <i>i</i>
m _i	Mass of component <i>i</i>
ρ	Density
P _{T,i}	Liquid / gas transfer kinetic rate
ρ _j	Process rate
R	The gas constant
r _{cryst}	Mineral precipitation rate
S _{aa}	Amino acids
S _{ac}	Acetate
S _{bu}	Butyrate
S _{ch4}	Methane
S _{h2}	Hydrogen
S _{fa}	Long fatty chain acids
S _I	Soluble inerts
S _{Fe2}	Iron (II)
S _{Fe3}	Iron (III)
S _{IC}	Inorganic carbon
S _{IN}	Inorganic nitrogen
S _{IS}	Hydrogen sulfide
S _{pro}	Propionate
S ⁰	Sulfur

S_{SO_4}	Sulfate
S_{su}	Sugars
S_{va}	Valerate
V	Volume
X_{aa}	Amino acid degraders
X_{ac}	Acetate degraders
X_{aSRB}	Acetate-degrading sulfate reducing bacteria
X_{bSRB}	Butyrate-degrading sulfate reducing bacteria
X_c	Composite material
X_{c4}	Valerate and butyrate degraders
X_{ch}	Carbohydrates
X_{fa}	Long fatty chain acid degraders
X_{h_2}	Hydrogen degraders
X_{li}	Lipids
X_{hSRB}	Hydrogenotrophic-degrading sulfate reducing bacteria
X_{pr}	Proteins
X_{pro}	Propionate degraders
X_{su}	Sugar degraders
X_I	Inerts
γ	Activity coefficient

NOMENCLATURE

Term / abbreviation	Definition
AAS	Atomic Absorption Spectroscopy
AD	Anaerobic Digestion
ADF	Acid Detergent Fibre
ADL	Acid Detergent Lignin
ADM1	Anaerobic Digestion Model No.1
ALD	Anoxic Limestone Drains
AMD	Acid Mine Drainage
DMPD	N.N.dimethyly-p-phenylene diamine dihydrochloride
DSR	Dissimilatory Sulfate Reduction
COD	Chemical Oxygen Demand
CSTR	Continuous Stirred Tank Reactor
DWAF	The Department of Water Affairs and Forestry
ICP	Inductively Coupled Plasma
IWA	International Water Association
LBR	Leach Bed Reactor
LFCA	Long Fatty Chain Acids
LT	Leachate Tank
MAPE	Mean Absolute Percent Error
NDF	Neutral Detergent Fibre
NfE	N-free Extracts
ODE	Ordinary Differential Equation
PFR	Plug Flow Reactor
PRB	Permeable Reactive Barriers
RF	Raw Fibre
RL	Raw Lipids
RP	Raw Protein
SEM	Scanning Electron Microscope
SRB	Sulfate Reducing Bacteria
SRT	Solid Retention Time
ThOD	Theoretical Oxygen Demand
TS	Total Solids
UASB	Upflow Anaerobic Sludge Bed
VFA	Volatile Fatty Acids
VS	Volatile Solids
WWTP	Wastewater Treatment Plant
XRF	X-ray Fluorescence

1 INTRODUCTION

Mining occurs across the globe and is tied to the development of mankind. Mining activities have resulted in the formation of acidic wastewater enriched with sulfate and heavy metals (Peppas et al., 2000; Potgieter-Vermaak et al., 2006; Sheridan et al., 2013). This wastewater, known as acid mine drainage (AMD), is formed when mineral-sulfides, such as pyrite, are exposed to water and oxygen (Simate & Ndlovu, 2014). There are several health and environmental risks associated with AMD, which necessitates the treatment thereof (Greenway et al., 2017).

In the past, active treatment, which involves the addition of a neutralising chemical reagent, was commonly implemented to remediate AMD (Name & Sheridan, 2014), but this can be costly, requires constant monitoring and produces a bulky sludge which must be disposed of (Tichy et al., 1998). Passive treatment offers several advantages over active treatment, including the recovery of metals in the form of sulfides (Kaksonen & Puhakka, 2007). Active treatment does require smaller areas of land compared to passive biological systems, however, by optimising the bioprocesses, the land required for passive treatment can be reduced (Johnson & Hallberg, 2005).

Currently, research is being conducted on passive AMD bioremediation technologies which have the potential to be cost-effective and viable as long-term treatment solutions. During passive treatment the biological process, anaerobic dissimilatory sulfate reduction (DSR), is used to neutralize the pH, reduce sulfate concentration levels and lower heavy metal concentrations in AMD (Ramla & Sheridan, 2015). It is more difficult to accurately model and predict the discharge concentrations when implementing bioreactors. A comprehensive understanding of the multiple biological processes during passive treatment is required.

In previous research, Ramla & Sheridan (2015) showed that amelioration of AMD could be achieved using indigenous South African grass as an organic carbon source for anaerobic dissimilatory sulfate reduction (DSR) with 99% and 80% iron and sulfate removal efficiencies respectively. The pH of the treated AMD also increased from 3 to 8.5. The research presented in this thesis identified a suitable process configuration and conducted experiments to determine the treatment efficiency of this configuration.

A model was developed in order to fully understand the bioremediation of AMD using indigenous South African grass and the selected reactor design. The Anaerobic Digestion Model No.1 (ADM1) was developed by the International Water Association (IWA) to predict anaerobic processes. The model has been applied successfully to numerous wastewater treatment plants (WWTP) (Flores-Alsina et al., 2016) and is able to predict complex biochemical and physiochemical process of anaerobic digestion (Batstone et al., 2002). The basis of the ADM1 showed potential to predict the bioremediation of AMD and this thesis aimed to update the ADM1 and use the model to predict the bioremediation of AMD under anaerobic conditions using an organic substrate. Further research, conducted in this thesis, upgraded the ADM1 to include dissimilatory sulfate reduction (DSR) and iron transformations and forms the basis of the work presented in this study.

1.1 Problem Statement

There exists a need to develop cost-effective AMD treatment technologies for active and closed mines that require little maintenance. Treatment of AMD using indigenous grasses in a passive biological process offers a potential solution to this problem. Ramla & Sheridan (2015) have shown proof of concept of this process, however further process development is required to determine the technological feasibility of such a process and whether such a process is viable for industry.

1.2 Research Objectives

This research aimed to develop a semi-passive treatment system for the bioremediation of AMD by examining various bioreactor designs and configurations using indigenous South African grass for the anaerobic process of dissimilatory sulfate reduction (DSR). In addition, the potential to extend the existing Anaerobic Digestion Model No.1 (ADM1) to include prediction of dissimilatory sulfate reduction (DSR) and iron transformations is explored. The specific objectives of this study are to:

- a) Design a process by identifying suitable bioreactor types and reactor configurations;
- b) Construct a laboratory scale bioprocess based on the most suitable process design;

- c) Conduct an experimental study to determine the most suitable operating conditions including the ratio of indigenous South African grass to the volume of AMD in the system;
- d) Conduct an experimental study with a laboratory scale bioprocess to determine its efficiency in the treatment of the AMD and synthetic AMD; and
- e) Develop a process model which can predict the bioremediation of AMD and compare the process model data to the experimental data to determine the reliability of the model.

1.3 Dissertation Layout

The thesis is presented as five chapters each of which are described here:

Chapter 1: This chapter gives a brief background into AMD, it describes the problem statement and lays out the research objectives of this dissertation.

Chapter 2: The literature review provides a comprehensive description of theory into the formation of AMD, the impacts of AMD and passive treatment of AMD. This chapter investigates possible bioreactor types and reactor configurations for treating AMD using indigenous South African grass. It also investigates the potential of using the Anaerobic Digestion Model No.1 (ADM1) as a basis for developing a model to predict the bioremediation of AMD. An examination is made into how indigenous South African grass was characterised for the ADM1.

Chapter 3: This chapter outlines the experimental and apparatus set-up of the laboratory-scale reactor and the operational scheme. It describes the analytical methods conducted during the experiments. This chapter includes methods used for model development and testing.

Chapter 4: The findings of the experiments are presented in this chapter. An investigation was made into the suitable solid/liquid ratios for the described process. A comparison of the results of the process using AMD and synthetic AMD is made. This chapter includes the upgrade of the ADM1 to account for sulfate reduction and iron transformations. It also provides an insight into the reactor design equation used in the model development.

A direct and cross validation of the developed model and a comparison of the modelling results to experimental results for both AMD and synthetic AMD is given.

Chapter 5: In this chapter a summary of the findings from this research is presented and an overview of recommendations is given.

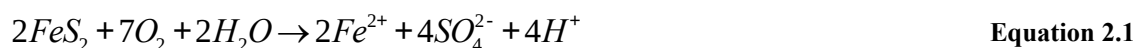
2 LITERATURE REVIEW

2.1 The generation of acid mine drainage (AMD)

One of the main contributors to water pollution in countries that have historic or existing mining industries is acid mine drainage (AMD). South Africa is rich in mineral resources and as a result the mining industry is one of the main economic driving forces (Potgieter-Vermaak et al., 2006; Ramla & Sheridan, 2015). However, mining activities can cause serious environmental problems when sulfide-based polymetallic ores are exposed to oxygen, water and microorganisms which may create a net acidic effluent from existing and abandoned mines (Ramla & Sheridan, 2015).

AMD is currently a major problem in the Witwatersrand basin, which has been subjected to mining activities, giving rise to the pollution of fresh water sources such as rivers, streams and ground water (McCarthy, 2011). The sulfate-rich and acidic liquid can impair human health and the biodiversity of surrounding ecosystems (Akciil & Koldas, 2006). Characteristics of AMD are high concentrations of sulfate, heavy metals and other toxic substances, as well as a low pH (Peppas et al., 2000; Potgieter-Vermaak et al., 2006; Sheridan et al., 2013). The most common mineral sulfide-based ore is pyrite (FeS_2) although pyrrhotite (FeS) and chalcocite (Cu_2S) can also contribute to the increased concentrations of heavy metals and sulfate in mine wastewater. AMD is formed through three main pathways: the oxidation of pyrite by oxygen and ferric iron, the oxidation of ferrous iron and ferric iron hydrolysis and precipitation (Simate & Ndlovu, 2014; Name & Sheridan, 2014).

The oxidation of pyrite leads to the formation of ferrous ions and protons, increasing the net acidity as shown in Equation 2.1.



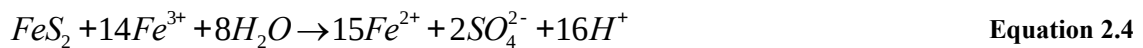
The formation of sulfate is a direct indication that there is oxidization of a sulfide-based mineral (Simate & Ndlovu, 2014). Ferrous iron is further oxidized into ferric iron in the following pathway:



Ferric iron precipitates to $Fe(OH)_3$ or jarosite ($H_3OFe_3(SO_4)_2(OH)_6$) at a pH range of 2.3-3.5 (Akciil & Koldas, 2006; Simate & Ndlovu, 2014).



The additional ferric iron left in the system oxidizes pyrite decreasing the pH and increasing the sulfate concentration.



Naturally occurring bacteria such as the acidophilic sulfide oxidizing species *Acidithiobacillus ferrooxidans*, increase the rate of AMD formation. These chemoautotrophs obtain their energy from the oxidation of ferrous iron (Banks et al., 1997).

AMD generation takes place from various sources such as; underground mine tunnels and shafts, open pits, mill tailings, waste rock dumps and ore stockpiles which can be active sources of AMD for decades after mining activities have ceased (Peppas et al., 2000; Potgieter-Vermaak et al., 2006; Ramla & Sheridan, 2015; Simate & Ndlovu, 2014). Active mines have low water tables due to artificial pumping. However, once a mine is closed, pumping ceases and groundwater decants into abandoned mines (Johnson & Hallberg, 2005; Potgieter-Vermaak et al., 2006). This acidic, metal enriched water passes through adits and enters catchment areas causing environmental damage (Potgieter-Vermaak et al., 2006).

2.2 The impact of acid mine drainage (AMD)

South Africa's water security is currently threatened due to drought and extreme weather conditions and AMD poses a further challenge to the country's water shortage (Cobbing, 2008). Large volumes of AMD are generated around the Witwatersrand basin and the effluent from mines seeps into the clean surface and groundwater resources (Akciil & Koldas, 2006). AMD has negative social, environmental and economic impacts.

AMD has a damaging effect on the economy. The corrosive nature of AMD can be detrimental to infrastructure (Taylor, 2006; Johnson & Hallberg, 2005). Dissolved hydrogen sulphide is highly reactive under anaerobic conditions. The high concentrations of sulfate in AMD results in the production of H_2S which is highly toxic and corrosive (Mambo, 2011). Scaling in pipes and degradation concrete structures results when dolomite rock is exposed to AMD (Madzivire, 2015). AMD can have a negative impact on agriculture as it adjusts the biochemistry of soil; plants are not able to grow due to the lack of essential nutrients required by plants. The elevated iron concentrations, found in AMD, prohibits plants to absorb manganese which is required for plant growth (Taylor et al., 2005).

The term 'yellow boy' has been given to streams and rivers polluted by AMD due to the orange characteristic of AMD as shown in Figure 2.1. Iron (III) hydroxide causes water to turn orange in colour (Lieberink, 2009).



Figure 2.1 'Yellow boy' (Lieberink, 2009)

Figure 2.2 depicts two hippopotami swimming in a dam in the Krugersdorp Game Reserve which was polluted by AMD (Lieberink, 2009). Water pollution, caused by

AMD, contains concentrations of toxic heavy metals and tends to be radioactive in the Witwatersrand basin, since the ore-body is uraniferous (Tutu et al., 2008).



Figure 2.2 A dam polluted with AMD in the Kurgersdorp Game Reserve (Lieverink, 2009)

The high iron concentration, present in AMD, builds up in animal's organs and fish gills (Jennings et al., 2008) which results to a loss of wildlife and livestock. Microbes and organisms present in water bodies, ingest iron found in AMD (Jennings et al., 2008) and are unable to obtain the nutrients required to survive. This leads to a decrease in the food source necessary to sustain aquatic life. Jennings et al. (2008) reported that iron (III) hydroxide forms a layer of coating along riverbeds and reduces the penetration of light through the water. All these factors negatively impact the biodiversity of the surrounding environment.

AMD imposes health risks when consumed by humans. AMD adjusts the chemistry of water which is needed to sustain human life. High concentrations of heavy metals, sulfates and a low pH can have detrimental impacts on human health and lead to various diseases including cancer. High concentrations of sulfate can have a laxative effect when consumed and can lead to diarrhoea. The Department of Water Affairs and Forestry

(DWAF) reported the effects of high sulfate concentrations in the South African Water Quality Guidelines published in 1996, as shown in Table 2.1.

Table 2.1 Effects of Sulfate on Aesthetics and Human Health (Department of Water Affairs and Forestry., 1996)

Sulfate Range (mg/L as SO₄⁻)	Effects
Target Water Quality Range 0-200	<i>No health or aesthetic effects are experienced</i>
200 - 400	Tendency to develop diarrhea in sensitive and some non-adapted individuals. Slight taste noticeable.
400 - 600	Diarrhoea in most non-adapted individuals. Definite salty or bitter taste.
600 – 1 000	Diarrhoea in most individuals. Use-adaption does not occur. Pronounced salty or bitter taste.
> 1 000	Diarrhoea in all individuals. User-adaption does not occur. Very strong salty and bitter taste.

The presence of mercury and lead in AMD impacts the central nervous system when consumed (Taylor, 2006). Taylor (2006) reported diseases associated with cancer can be brought on by high concentrations of arsenic and uranium found in AMD. Coetzee & Winde (2006) reported a presence of radioactive material found along the Witwatersrand Basin which poses a risk of cancer and impacts the environment. Cobbing (2008) has conducted research on the effects that consumption of water containing low iron concentrations by pregnant woman and the findings show that this can impact fetal development and can cause mental disorders. Table 2.2 reports the findings from the DWAF on the effects of iron on human health. Low pH ranges increase the solubility of toxic heavy metals which impacts human health.

**Table 2.2 Effects of iron on aesthetics, human health and household distribution systems
(Department of Water Affairs and Forestry., 1996)**

Iron Range (mg/L)	Effects
Target Water Quality Range 0-0.1	<i>No taste, other aesthetic or health effects associated with consumption and use</i>
0.1 – 0.3	Very slight effects on taste and marginal other aesthetic effects. Deposits in plumbing with associated problems may begin to occur. No health effects; the water is generally well tolerated.
0.3 – 1.0	Adverse aesthetic effects (taste) gradually increases as do possible problems with plumbing. No health effects.
1.0 – 10	Pronounce aesthetic effects (taste) along with problems with plumbing. Slight health effects expected in young children and sensitive adults.
10 - 30	Severe aesthetic effects (taste) and effects on plumbing (slimy coatings). Slight iron overloads in some individuals. Chronic health effects in young children and sensitive individuals in the range of 10 – 20 mg/L, and occasional acute effects toward the upper end of this range.
30 - 100	As above Long-term health effects gradually increase.
100 - 300	As above Chronic health effects. Acute toxicity may begin to appear.
300 – 3 000	As above Chronic and acute health effects. Accidental iron poisoning from water is rare.
3 000 – 30 000	As above Lethal toxicity occurs.

2.3 The treatment of acid mine drainage (AMD)

Inhibiting AMD formation is the preferred form of control however, this is not always pragmatic and migration control has to be implemented. Migration control can be in the form of active or passive treatment. Both methods of treatment aim to generate alkalinity and promote the precipitation of heavy metals. Active treatment implements an alkaline chemical reagent by mechanical addition in order to increase pH (Gaikwad & Gupta, 2008; Ochieng et al., 2010). Lime, ammonia and caustic soda (Name & Sheridan, 2014) are a few examples of alkaline chemicals utilised in active treatment, however the addition of alkaline chemicals leads to high operating costs. Active treatment requires

continuous monitoring, maintenance and skilled manpower of which availability is limited in South African rural areas. Actively treated AMD often results in sludge rich in iron and other heavy metals, which is problematic to dispose. Passive treatment aims to reduce human intervention by implementing biological and chemical reactions that are naturally occurring (Name & Sheridan, 2014; Ochieng et al., 2010). An advantage of passive treatment is that systems are self-operating; therefore, requiring infrequent monitoring and low maintenance costs. Although passive treatment systems may require a large surface area, mining sites in South Africa often have ample space to accommodate them.

2.3.1 Active Treatment

Active treatment is commonly used in industry (Coulton et al., 2003b). Active treatment involves the addition of an alkaline reagent in order to raise the pH of AMD (Jennings et al., 2008). Examples of neutralising agents that are used during active treatment include lime, limestone, sodium carbonate, calcium carbonate, sodium hydroxide, hydroxide, magnesium oxide and ammonia (Johnson & Hallberg, 2005). During active treatment, the oxidation of the ferrous iron is accelerated (Akciil & Koldas, 2006), resulting in the precipitation of iron and other heavy metals as carbonates and hydroxides. This precipitation results in the production a black sludge containing iron and other heavy metals (Johnson & Hallberg, 2005). Active treatment uses the following chemical and physical process for the treatment of AMD (Taylor et al., 2005):

- pH control and precipitation;
- Electrochemical concentration;
- Biological mediation / redox control (sulfate reduction);
- Crystallisation;
- Ion exchange / absorption or adsorption; and
- Flocculation and filtration.

Active treatment requires a small area of land, is a proven technology and is an effective and fast treatment solution (Burgers, 2002). Active treatment allows for the treatment of large volumes of AMD and can cope with variations in temperature, acidity and metal concentrations (Burgers, 2002). However, one of the biggest drawbacks of active

treatment is that it requires high capital and operating costs (Johnson & Hallberg, 2005). Regular maintenance is necessary (Johnson & Hallberg, 2005) for active treatment which requires skilled man power. Active treatment not only utilises expensive chemical reagents, it also requires the disposal of the black sludge that is produced (Johnson & Hallberg, 2005). Passive treatment solutions are becoming more desirable as they are inexpensive and require less monitoring and maintenance.

2.3.2 Passive treatment of acid mine drainage (AMD)

An alternative to active treatment is using anoxic limestone drains (ALDs) (Kleinmann et al., 1998). ALDs increases the alkalinity of AMD and prevents the oxidation of Fe^{2+} . AMD is passed through a bed of limestone (Johnson & Hallberg, 2005) in the absence of oxygen. ALDs have limitations as they are not able to treat all types of AMD, particularly AMD containing high concentrations of iron and aluminium, as the build-up of these metal hydroxides cause ALDs to fail (Johnson & Hallberg, 2005).

Anaerobic wetlands are low-cost alternatives for the passive treatment of AMD (Kaksonen & Puhakka, 2007). Wetlands are reconstructed ecosystems that utilises numerous physical, chemical and biological processes for the treatment of AMD (Johnson, 2000). Wetlands are often used in conjunction with ALDs (Kaksonen & Puhakka, 2007). Wetlands require lower costs and less maintenance, however, it's performance can vary, and it requires large areas of land (Johnson & Hallberg, 2005). The performance of wetlands are dependent on the seasons which means they are harder to monitor and control (Kaksonen & Puhakka, 2007).

Alternative to wetland are permeable reactive barriers (PRBs) which utilises reactive materials that are contained in a pit and allows for the flow through of water (Johnson & Hallberg, 2005). PRBs incorporate SRBs to reduce sulfates by using organic sources contained in the PRBs and alkalinity is self-generated (Richardson & Nicklow, 2002). Limestone and gravelly may be added to increase alkalinity and permeability of the PBRs (Richardson & Nicklow, 2002). Drawbacks include clogging due to metal precipitation on the barriers and the reduction of organic sources which leads to a decreased efficiency of PBRs (Kaksonen & Puhakka, 2007).

Another alternative is bioreactors which offer several advantages for the treatment of AMD such as the ability to control and predict the performance of the system, reduced concentrations of sulfates and they allow for the recovery of heavy metals (Johnson & Hallberg, 2005). Bioreactors also require smaller land areas than wetlands. Bioreactors need to be optimised and for optimal functioning require a comprehensive understanding of the multiple biological, chemical and physical reactions involved.

Bioremediation of AMD often involves the use of SO_4^{2-} and carbon to feed sulfate reducing bacteria (SRB) which results in the generation of alkalinity and precipitation heavy metals. SRB catalyses DSR under anaerobic conditions (Lindsay et al., 2011). SRB coupled with an organic carbon source, which serves as an electron donor, causes the reduction of sulfate to sulphide (Ramla & Sheridan, 2015).



The pH of the system is raised by the generation of bicarbonate. The production of hydrogen sulfide allows for the precipitation of heavy metals by the formation of insoluble metal sulfides.



The rate of DSR is dependent on the chemical oxygen demand (COD), which is related to the availability of oxidizable organic carbon (Greiben et al., 2009; Sheridan et al., 2013). Biological remediation has been successfully implemented at full-scale as anaerobic wetlands and bioreactors (Sheridan et al., 2013; Kadlech et al., 2000; Johnson & Hallberg, 2005).

2.4 The selection of Indigenous South African grass as the organic carbon source for the bioremediation of acid mine drainage (AMD)

Cost-effective and suitable sources of organic carbon for the bioremediation of AMD are lignocellulosic based biological waste products, in which the cellulose breaks down into monomeric sugars and oligomers (Ramla & Sheridan, 2015). As the organic source is metabolized organic acids are generated which are utilised by the SRB (Greiben et al.,

2009). Greben et al. (2009) conducted a study using grass cuttings as an organic carbon source in an anaerobic bioreactor of 20 L containing SRB. The dissimilatory sulfate reduction (DSR) rate was not limited as grass (*Pennisetum clandestinum*) was added to the experiment biweekly. Greben et al. (2009) demonstrated that Kikuyu grass was a cost-effective and suitable carbon source in the remediation of AMD as the efficiency of sulfate removal was 86%. The removal of metals was also observed as precipitation of metal sulfides. Kikuyu grass is not indigenous to South Africa therefore, Ramla & Sheridan (2015) conducted experiments using indigenous grass (*Hyparrhena hirta* and *Setaria Sphacelata*) as a carbon and energy source for the DSR process under anaerobic conditions. After 70 days iron and sulfate was removed at an efficiency of 99% and 80% respectively, while the pH increased from 3 to 8.5. These results showed that grass, indigenous to South Africa, could be utilised as a sustainable organic carbon source for the remediation of AMD (Ramla & Sheridan, 2015).

Grass consists mainly of lignin, hemicellulose and cellulose. Cellulose, which is embedded by lignin and hemicellulose, is a polymer of fermentable monomeric sugars. AMD is recognised as a source of sulfuric acid and can be utilised for the acidic pre-treatment of grass (Magowo et al., 2015), which disrupts the lignin and hemicellulose structures that acts as a protective shield around cellulose. The pre-treatment of grass exposes monomeric constituents of cellulose for enzymatic hydrolysis (Esteghlalian et al., 1997). Lignin is comprised of reactive groups that are able to bind to and remove heavy metals. The enzymatic digestibility of cellulose is also increased for cellulolytic enzymes during acidification of the grass using AMD due to the reduction in the cellulose crystallinity (Kootstra et al., 2009; Esteghlalian et al., 1997). During saccharification cellulase enzymes are utilised as a catalyst for the hydrolysis of cellulose to produce fermentable sugars including: glucose, xylose, arabinose, mannose and galactose (Sun & Cheng, 2002; Sun & Cheng, 2005). The production of fermentable sugars from indigenous South African grass is not considered in this study.

2.5 Process selection and design configuration

In literature various process designs and configurations have been used for the anaerobic digestion of organic waste. These designs can incorporate one-stage or two-stage digesters, wet or dry digesters, batch or continuous digesters and also include high-rate

digesters. The selection of reactor and process configuration must be chosen in accordance for the treatment of the selected organic waste. This section reviews possible process configurations (semi-passive) and aims to identify the one most suitable for the mono-digestion of indigenous South African grass.

Anaerobic digestion can be implemented in one-stage or two-stage digesters. One-stage digesters allow for all processes of anaerobic digestion to occur simultaneously in one vessel. One-stage digesters are common in industry as the process is associated with lower costs, fewer technical difficulties and are simple to design, operate and maintain (Vandevivere et al., 2003). Two-stage digestion separates the microbial phases, hydrolysis and acidogenesis in the first digester and acetogenesis and methanogenesis takes place in the second digester. Two-stage digestion allows for the optimisation of the process by ensuring each phases of anaerobic digestion is complete (Yu et al., 2002).. A one-stage digester is proposed for this study due to its lower costs, fewer required operating skills and lower maintenance requirements.

The mono-digestion of grass can be achieved using either wet digesters or dry batch digesters. Wet digesters are usually continuous stirred tank reactors (CSTR) operating in a single or multi-stage process, implementing a recirculated liquid digestate (Nizami & Murphy, 2010). Wet digesters require feedstock that contains less than 20% total solids and therefore grass bales would require a dilution pre-treatment. Grass that has not gone through a pre-maceration process tends to get caught in moving parts (Dieterich, 2008). Dieterich (2008) reported that grass silage in the system tended to float on the surface of the liquid and as a result mixing had to be implemented. Additional processes of maceration and mixing yielded higher costs and higher energy demand. The application of dry digesters has increased over the past decade (De Baere & Mattheeuws, 2000). No dilution of the grass is required in batch digesters, instead a leachate is re-circulated, and no mixing takes place. Dry batch digesters have a lower energy demand than wet digesters, as there is no need to initially dilute the feedstock and does not necessitate the need for pump slurries and mixing. The volatile solids (VS) concentration generated in a dry bath digester will have a concentration curve similar to that of a parabola curve, the concentration of the VS solids will reach a maximum peak and then decline (Nizami & Murphy, 2010). This can be overcome by implementing a series of batch reactors that are

sequentially fed in order to simulate a continuous process. As the process aims to be semi-passive, a dry digester is proposed.

Continuous digesters provide a constant stream of feedstock into the system through the use of plug flow reactors (PFRs), CSTRs and upflow anaerobic sludge beds/blankets (UASB) (Nizami & Murphy, 2010). Feedstock with a high solid content being fed into continuous digesters would require dilution through slurring and pulping. Grass bales have a high concentration of solids and a less energy extensive approach would be to implore batch digesters. Batch digesters are initially fed with the feedstock and, under anaerobic conditions, was allowed to completely degrade under a certain period of time (Nizami & Murphy, 2010). The leach bed reactor (LBR) is essentially a batch reactor that holds the volatile solids while a leachate passes through the leach bed and is recirculated back to the leach bed. Batch digesters can be implemented in a sequential configuration and partnered with a continuous digester (such as an UASB) in order to simulate a continuous configuration.

High rate digesters retain biomass and increases the solid retention time (SRT) by attaching the biomass onto carriers forming granules of biomass. Upflow anaerobic sludge beds (UASB) are widely used due to its characteristics of low retention times (0.5 – 12 days) (Vandevivere et al., 2003) and the ability to take in a solid feed concentration of 4% to 15% (Barnett et al., 1978; Marchaim, 1992). UASBs are associated with higher costs and require more maintenance than LBRs. Consequently, it is proposed to use a leach bed reactor (LBR) in the process as this batch digester does not require any pre-processing of the grass and it recirculates the leachate.

The potential use of a one-stage LBR for the bioremediation of AMD using indigenous South African grass was therefore investigated and the results are presented in this thesis. A model was developed, on the basis of the Anaerobic Digestion Model No.1 (ADM1), in order to fully understand the bioremediation of AMD and the design configuration.

2.6 Anaerobic digestion model No.1 (ADM1)

Anaerobic digestion is the breakdown of a substrate by microbes in anoxic conditions. It has received increasing attention over the past decade due to the process' ability to

generate energy. The application of anaerobic digestion in the waste treatment sector has led to the formulation of many anaerobic models, however the discrepant and specific nature of these models hinder its use in industry (Batstone et al., 2002). The International Water Association (IWA) assigned a task force to establish a generalized model for anaerobic digestion. The anaerobic digestion model No.1 (ADM1) aimed to combine and update existing models for the goal of achieving a model to predict anaerobic digestion (Greenway et al., 2017). The ADM1 model is able to predict the disintegration of the composite materials (X_c) into carbohydrates (X_{ch}), lipids (X_{li}), protein (X_{pr}) and inerts (X_i). The ADM1 predicts the enzymatic hydrolysis of the particulate constituents into its soluble monomers of monosaccharides (S_{su}), long fatty chain acids (S_{fa}) and amino acids (S_{aa}) as shown in Figure 2.3. Disintegration and hydrolysis are extracellular processes (Batstone et al., 2002; Greenway et al., 2017). DSR takes place in anoxic conditions, utilising an organic substrate as a carbon source for SRB. As SRB use the soluble organic compounds produced during anaerobic digestion, in the presence of no oxygen, the ADM1 was selected as a suitable model and was upgraded to include the process of DSR.

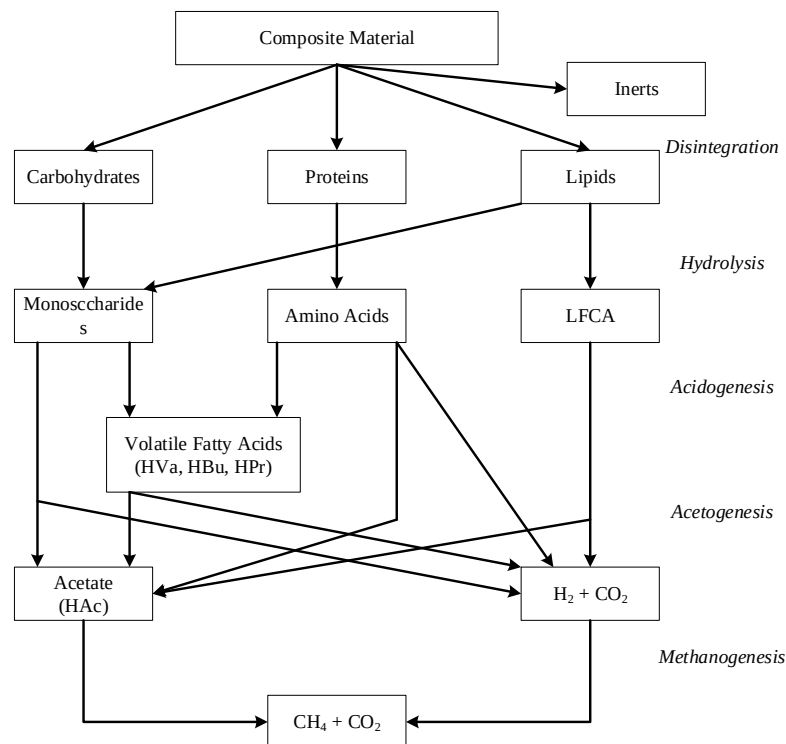


Figure 2.3 A graphical representation of the extracellular and biochemical process, and the relevant products of the ADM1 (Batstone et al., 2002)

The ADM1 model predicts three biochemical processes. The first is the conversion of soluble monomers to acetate (S_{ac}), butyrate (S_{bu}), valerate (S_{va}) and propionate (S_{pro}). Further production of acetate (S_{ac}) is predicted through the biochemical process of acetogenesis (Batstone et al., 2002). Lastly, methane production is included in the model through conversion of acetate and hydrogen by methanogenic bacteria. The stoichiometry matrix for these extracellular and biochemical processes are given in Appendix A for soluble and particulate compounds (Batstone et al., 2002). The matrices include the process rates (ρ_j), rate coefficients ($v_{i,j}$) and inhibition factors (I) for all the relevant processes. For more information please refer to Appendix A. The ADM1 model takes into account physiochemical processes including gas-liquid transfer (Batstone et al., 2002).

The increased popularity of the ADM1 has led to its implementation to a variety of organic matter in Wastewater Treatment Plants (WWTP). The heterogeneous nature of organic waste needs a systematic approach to characterize substrate influent and requires to be implemented as the fractioning of input state variables, which is closely linked to hydrolysis in AD and related to the degradability of the composite material (X_c). An investigation was made into determining the fractioning of Indigenous South African grass (X_c) into its particulate constituents including carbohydrates (X_{ch}), protein (X_{pr}), lignin (X_{li}) and inerts (X_I).

2.6.1 Characterisation of Indigenous South African Grass for the ADM1

In the ADM1, the composite material (organic matter) disintegrates into carbohydrates, lipids, proteins, and inerts (Batstone et al., 2002). The fractioning of the composite material (X_c) into its following particulate constituents was assumed to fractionate as lumped variables (Batstone et al., 2002). The fractioning of the composite material has previously been neglected in the ADM1 (Koch et al., 2010). Rather closer attention has been paid to the kinetic parameters of hydrolysis, as this is the rate limiting step in anaerobic digestion (AD) and an important factor in predicting the potential production of biogas (Koch et al., 2010). The ADM1 was derived on the basis of chemical oxygen demand (COD). The total input COD is comprised of two main parts, the unbiodegradable fraction and the biodegradable fraction. The unbiodegradable fraction is comprised of inerts. The biodegradable fraction is comprised of both particulates (X_{ch} ,

X_{pr} , and X_{li}) and soluble compounds (S_{su} , S_{aa} , S_{fa} , S_{va} , S_{bu} , S_{pro} and S_{ac}), which varies for different organic wastes.

Biomass is composed of three main components: water, organics and inerts. Inerts are comprised of inorganics and ash and will not degrade during anaerobic digestion (AD). The ADM1 model distinguishes the biodegradable fraction (D) and states that $D=1$ for organic compounds that are completely biodegrade. However, it is rarely the case that a biomass will fully degrade, and the fraction of biodegradable compounds is denoted as $(1-D)$ (Koch et al., 2010). Unlike the commonly implemented organic matter in an AD process, indigenous South African grass is a particulate composite material and determination of the biodegradable and unbiodegradable fractions will focus on obtaining the composition of the grass. The next step in distinguishing the biodegradable fraction of an organic compound is to determine the amount of total solids (TS) and volatile solids (VS) using methods described by (Lübken et al., 2011).

The production of biogas (CH_4 and H_2S) is dependent on the portion the lignocellulosic material disintegrates into carbohydrates (X_{ch}), lipids (X_{li}), protein (X_{pr}) and inerts (X_I). These fractions can be determined by using the Fodder Analysis to find both the Total and Volatile Solids content and the nutrients content of the grass. The extended Weender Analysis shown in Figure 2.4 depicts the COD fractioning of the lignocellulosic material into its relative constituents by determining the Raw Protein (RP), Raw Fibre (RF) and N-free extracts (NfE). The composition of the carbohydrates is further broken down to starch ($RF + NfE - NDF$), cellulose ($ADF - ADL$), hemicellulose ($NDF - ADF$) and lignin (ADL). Water and ash have been excluded from the ADM1 model as they do not have COD.

The theoretical oxygen demand (ThOD) of an organic compound can be determined by methods outlined by Gerike (1984) and shown in Equation 2.7 and Equation 2.8, assuming nitrogen is converted into ammonia.

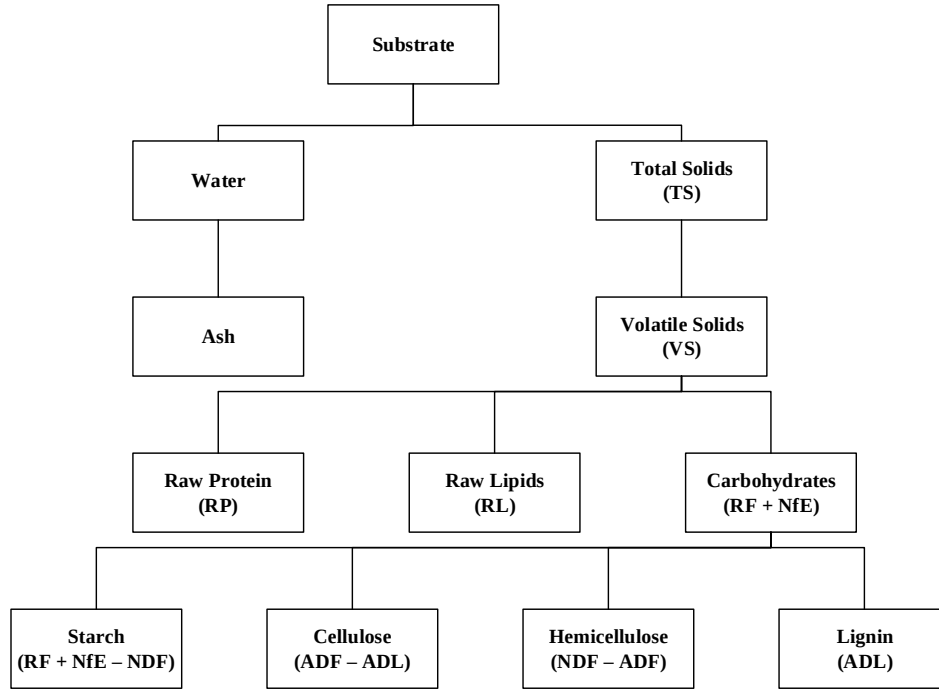
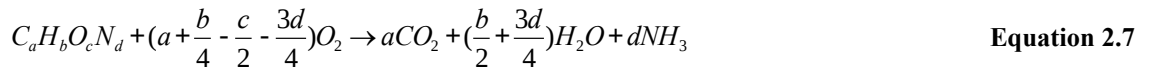


Figure 2.4 Determination of substrate composition by extended Weender analysis (Koch et al., 2010)



$$ThOD = \frac{16(2a + 0.5(b - 3d) - c)}{12a + b + 16c + 14d} \left[\frac{g_{O_2}}{g_{C_a H_b O_c N_d}} \right] \quad \text{Equation 2.8}$$

Koch et al. (2010) proposed that the ThOD for the composite material (X_c) can be determined by calculating the fraction of ThOD for each organic substance (X_{ch} , X_{li} , X_{pr} , X_l) as shown in Equation 2.9:

$$X_c = r_{substrate} \times TS \times ((RF + NfE - ADL) \times ThOD_{ch}) + (RP \times ThOD_{pr}) + (RL \times ThOD_{li}) + (ADL \times ThOD_l) \left[\frac{kgCOD}{m^3} \right] \quad \text{Equation 2.9}$$

Table 2.3 The Theoretical Oxygen Demand (ThOD) for the different organic fractions found in grass

Organic fraction	Elemental formula	Theoretical Oxygen Demand (ThOD) [Kg O ₂ / Kg TS]
Protein (X_{pr})	C ₅ H ₇ O ₂ N	1.42
Lipid (X_{li})	C ₅₇ H ₁₀₄ O ₆	2.90
Starch, cellulose, hemicellulose (X_{ch})	(C ₆ H ₁₀ O ₅) _n	1.19
Lignin (X_l)	C _{10.92} H _{14.24} O _{5.76}	1.56

The stoichiometric f-factors (organic fraction), of which the composite material is broken down into, are represented by Equation 2.10 – Equation 2.13 (Koch et al., 2010). The biodegradable fraction carbohydrates are given in Equation 2.12 represents the fraction of degradable cellulose and hemicellulose and starch ((NDF-ADL)·d). The percentage of degradable cellulose and hemicellulose is denoted by ‘d’ and calculated by methods described by Koch et al. (2010).

$$f_{pr_Xc} = \frac{RP}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad \text{Equation 2.10}$$

$$f_{li_Xc} = \frac{RL}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad \text{Equation 2.11}$$

$$f_{ch_Xc} = \frac{(RF + NfE - NDF) + (NDF - ADL) \times d}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad \text{Equation 2.12}$$

$$f_{l_Xc} = \frac{ADL + (NDF - ADL) \times (1 - d)}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad \text{Equation 2.13}$$

The fraction of inerts is the sum of the fraction of lignin and the non-biodegradable portion of cellulose and hemi-cellulose (1-d). The composition of the relative components is divided by the VS in order to ensure the sum of the organic fraction equals 1 (Greenway et al., 2017).

The ADM1 model is extensive but does not include the biogeochemical cycles of iron and sulfur, which are found in wastewater (Flores-Alsina et al., 2016).

3 EXPERIMENTAL METHODOLOGY

3.1 Experimental and apparatus set-up

A one-stage leach bed reactor (LBR) combined with a leachate tank (LT) was used for the bioremediation of AMD. The leachate (AMD/synthetic AMD) was stored in the LT and trickled over the indigenous South African grass in the LBR. The leachate percolated through the grass and through a sieve plate, which was used to retain solids in the LBR. The leachate flowed back to the LT from the LBR by gravitational force. Figure 3.1 demonstrates the reactor step-up.

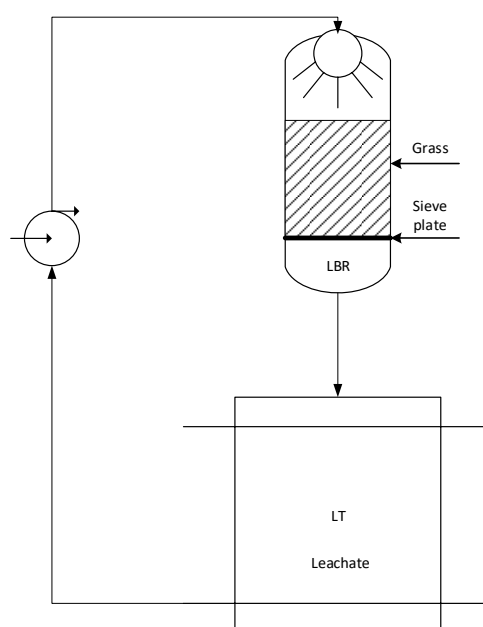


Figure 3.1 Reactor set-up of a one-stage LBR

A laboratory-scale reactor was built according to Figure 3.1 in order to passively treat the AMD utilising indigenous South African grass. The LBR was constructed using a 2 L plastic graduated measuring cylinder. A sampling port was created by inserting a tap at the bottom of the LBR. The LBR is shown in Figure 3.2. A plastic bucket with a lid was used to hold the leachate. A hole was drilled on the top of a plastic bucket in order to allow for piping to enter the bucket and the bucket was sealed to maintain anaerobic conditions. The picture below shows the constructed laboratory-scale reactor.



Figure 3.2 Laboratory scale leach bed reactor (LBR)

A sieve plate was constructed using multiple layers of materials with different pore sizes in order to stop solids flowing into the LT. The outer layers were constructed using stainless steel and had holes, with a diameter of 60 mm, drilled into it. The next layer contained in the sieve was a hard 1.19 mm polyethylene mesh followed by layers of filter cloth. The middle of the sieve contained a plastic dishwashing scouring pad. Figure 3.3 demonstrates the construction of the final sieve plate.



Figure 3.3 Top view and cross-sectional view of the sieve

The leachate was circulated from the LT to the LBR using a Watson Marlow Bredel 620Du pump (manufactured in the United Kingdom). A hose pipe garden sprinkler head was connected to the pump and inserted on top of the LBR. The sprinkler head was

mounted onto a plastic Perspex cast acrylic block slab and cylinder (as shown in Figure 3.4) in order to seal the LBR and prevent oxygen from entering the system.



Figure 3.4 Spray nozzle head

The final reactor configuration is shown in Figure 3.5. Two reactors were used in order to run experiments simultaneously.



Figure 3.5 Final laboratory-scale reactor configuration

3.2 Operational scheme

Initially the LT contained AMD or synthetic AMD which was trickled over the indigenous South African grass and became a leachate which contained soluble organic compounds. The soluble organic compounds included; monosaccharides (S_{su}), long fatty chain acids (S_{fa}), amino acids (S_{aa}), valerate (S_{va}), butyrate (S_{bu}), propionate (S_{pro}) and acetate (S_{ac}), which were utilised as an organic carbon food source for the bacteria in the system. The objective was for the process to be as passive as possible and hence a decision was taken not to inoculate to minimize the skills and maintenance requirements of the process and instead promote the growth of the naturally occurring microorganisms on the grass. The reactor was operated at ambient temperature to align with the aim of creating a semi-passive process for the bioremediation of AMD. The reactors were placed by a window that had natural sunlight for the majority of the day. The leachate was sprayed over the grass at a flow rate of 10 mL min^{-1} . The process was operated over a course of 30 days and the leachate was recirculated through the system. No materials were added or removed over the course of the experiment. Two sets of experiments were conducted. The first set aimed to find the optimal solid/liquid ratio to treat AMD. This experiment used synthetic AMD, which was made in the laboratory, and different loading rates. The next experiment focused on comparing the efficiency of the process using AMD and synthetic AMD.

3.3 Materials and methods

Three solid/liquid ratios were compared in order to find the optimal solid/liquid ratios for the bioremediation of grass. The first reactor contained 34 L of synthetic AMD and 50 g of grass. The starting solid/liquid ratio of 1.5 g/L was adapted from Thamsirioj et al. (2012), who focused their research on the two-stage digestion of grass for the production of biogas. The next solid/liquid ratio that was analysed was 17 L of synthetic AMD and 100 g of grass which gave a solid/liquid ratio of approximately 6 g/L. The last ratio (30 g/L) tested comprised of 5 L of synthetic AMD 150 g of grass as shown in Table 3.1. Table 3.2 demonstrates chemicals and quantities used to make up the synthetic AMD and the initial pH of all three solutions.

Table 3.1 Solid/Liquid Ratios

Experimental Run	Mass of grass (g)	Volume of liquid (L)	Ratio
1	50	34	1.5
2	100	17	6
3	150	5	30

Table 3.2 Composition of the synthetic AMD solution and initial pH for solid/liquid ratios of 1.5 g/L, 6 g/L and 30 g/L

Solid Liquid Ratio	FeSO ₄ ·7H ₂ O (g)	H ₂ SO ₄ (97%) (ml)	NaOH (50%) (ml)	Initial pH
1.5 g/L	80.37	42.93	0.75	2.83
6 g/L	40.18	21.46	0.37	2.77
30 g/L	11.82	6.31	0.11	3.23

The concentration of soluble COD, sulfates and pH was measured to determine the performance of the three solid/liquid ratios. A scanning electron microscope (SEM) analysis was also conducted on the grass to compare the destruction of the grass surface.

The indigenous South African grass obtained was from Braamfontein, Johannesburg South Africa, and was naturally occurring grass which had been cut and baled. The grass was a combination of *digitaria eriantha* and *eragrostis*. The grass was prepared by passing through a grass shredder three times to obtain an average length of 3 cm. The grass was weighed and then slightly washed using distilled water in order for it to be compressed and placed in the LBR.

The synthetic AMD composition described in Table 3.1 ratio 30 g/L, was selected for the comparison to the behaviour of AMD. AMD was obtained from an operational coal mine in Mpumalanga, South Africa. An Inductively Coupled Plasma (ICP) elemental analysis was conducted on the AMD, by the Chemistry Department at the University of the Witwatersrand, and the results are provided in Appendix B. The initial pH of the AMD was 2.4. A quantity of 5 L of AMD and 150 g of grass was used to conduct the experiment. The concentration of soluble COD, sulfates, dissolved sulphides, dissolved iron, sugars and pH was measured in order to compare the performance of the process for treating AMD compared to synthetic AMD. The experiment was conducted in duplicate.

3.4 Analytical methods

3.4.1 Preparation and storage of samples

Samples were taken from the sampling port at the bottom of the LBR and from the LT. The LT was closed in order to maintain an anaerobic system and prevent the loss of the leachate through evaporation. The samples (45 mL) were collected into 50 mL conical test tubes and were taken every 2-3 days for a period of 30 days. The samples were centrifuged and decanted in order to separate in particulates and the liquid. The liquid was then passed through a cellulose acetate membrane with a pore size of 0.45 μm . The samples were stored in a freezer to ensure that if any bacteria remained in the sample all microbial activity was minimised.

3.4.2 pH

The pH of all the sampled leachate was tested using a find out OHAUS ST3100 pH meter. The pH probe was stored in a solution of 3M KCl. Before each use, the meter was calibrated using a buffer solution of pH = 4, 7 and 9. Between the pH measurement of each sample, the pH probe was rinsed with distilled water to prevent cross contamination.

3.4.3 Testing the soluble COD concentration

COD was assayed using a Merck Spectroquant® solutions A+B according to the manufacture's specifications. Before samples were placed in the Merck Spectroquant® Pharo 300 UV/VIS, a blank sample containing distilled water was placed in the spectroquant at each measured wave length in order to allow for the Merck Spectroquant® Pharo 300 UV/VIS to be calibrated at the specified wavelength. A calibration curve was then determined by inserting samples with a known COD concentration (Table 3.3) into the Merck Spectroquant® Pharo 300 UV/VIS. The calibration curve was then used to determine the COD concentration at the measured absorbance.

Table 3.3 COD concentrations of prepared standards

Measuring COD Range (mg/L)	10-150	100-1500	500-10000
Blank (mg/L)	0	0	0
Standard 1 (mg/L)	20	200	1000
Standard 2 (mg/L)	75	750	5000
Standard 3 (mg/L)	150	1500	10000

3.4.4 Testing the sulfate concentration

A barium acetate assay was used in order to determine the sulfate concentrations of the samples. Glycerine (50 μ L) was mixed with 5 mL of sample (IS 3025: Part 24, 2003). A micro measurement of barium acetate was mixed with the solution and immediately shaken. The solution was then placed in the Merck Spectroquant® Pharo 300 UV/VIS and the absorbance were tested at a wavelength of 880 nm. Calibration curves were determined by preparing known sulfate concentration solutions as shown in Table 3.4.

Table 3.4 Sulfate concentrations of prepared standards

Measuring Sulfate range (mg/L)	0 – 1000	500 - 2000
Blank (mg/L)	0	0
Standard 1 (mg/L)	50	500
Standard 2 (mg/L)	100	1000
Standard 3 (mg/L)	500	1500
Standard 4 (mg/L)	1000	2000

The barium acetate assay was limited to sulfate concentrations below 2000 mg/L. In order to determine the sulfate concentrations of samples that contained a higher concentration of 2000 mg/L, the samples were diluted, and the final concentration was determined by:

$$C_1V_1 = C_2V_2$$

Equation 3.1

Where C_1 is the actual concentration of the sample and V_1 is the volume of the sample before dilution. C_2 is the sulfate concentration measured by the Merck Spectroquant® Pharo 300 UV/VIS and V_2 is the final volume after the sample has been diluted with distilled water.

3.4.5 Testing the dissolved sulfide concentration

The sulfide assay was performed using three solutions including, N,N-dimethyl-p-phenylene diamine dihydrochloride (DMPD) solution, ferric chloride solution and zinc acetate solution. The DMPD solution was prepared by mixing 2 g of $(CH_3)_2N.C_6H_4.NH_2.2HCl$ and 500 mL of 6 M HCl. The ferric chloride solution combined 8 g $FeCl_3$ and 500 mL of 6 M HCl. Finally, a solution of zinc acetate was prepared by combining 10.44 g of zinc acetate and 1000 mL of deionised water. Before the solutions were prepared, the crystals were rinsed with deionised water to remove any oxidation products that are present on the surface of the crystals. The ferric chloride solution is a catalyst for the assay. A soluble zinc sulphide precipitate is formed from the reaction of sulfide and zinc acetate. The precipitate is soluble in the acidic DMPD reagent forming a blue solution. The sample was prepared by pipetting 200 μ L of the zinc acetate in a clean test tube followed by 20 μ L of the sample. Deionized water was added to make up a volume of 5 mL. Next, 0.5 mL of DMPD solution was added, followed by 0.5 mL of the ferric chloride solution. The solution was placed into a vortex mixer and allowed to react for 5 minutes. The sample was placed into a Merck Spectroquant® Pharo 300 UV/VIS and measured at a wave length of 670 nm. A calibration curve was determined by using known dissolved sulfide concentrations shown in Table 3.5. The standards were made using a stock solution of $Na_2S.9H_2O$ and deionised water.

Table 3.5 Sulfide concentrations of prepared standards

Measuring Sulfide range (mg/L)	0 – 0.25	0.25 – 1
Blank (mg/L)	0	0
Standard 1 (mg/L)	0.05	0.25
Standard 2 (mg/L)	0.1	0.5
Standard 3 (mg/L)	0.15	0.75
Standard 4 (mg/L)	0.2	1
Standard 5 (mg/L)	0.25	-

3.4.6 Testing for sugar concentration

The sugar concentration was tested with an Agilent High Performance Liquid Chromatography (HPLC) using a Biorad Aminex fermentation monitoring column. A medium of 0.5 M H₂SO₄ was used at a minimum flowrate of 0.8 ml/min, while operating at 65 °C. A 80 µL sample injection volume was used to perform the assay.

3.4.7 Testing for dissolved Iron and total Iron concentration

Atomic absorption spectroscopy (AAS) was used for the determination of the iron concentration and the analysis was conducted using Agilent Technologies 200 Series AA. A mixture of gases, including air and acetylene was used and the detection wavelength was specified by the manufacture. An AAS was performed on the unfiltered samples to determine the total iron. The dissolved iron was determined by conducting an AAS on the filtered samples. Dilute nitric acid (0.45 mL) was added to 15mL of sample, which was stored in the refrigerator before testing.

3.4.8 Scanning Electron Microscopy (SEM)

A SEM analysis was conducted on the grass in order to determine the structural breakdown of the grass. An Oxford EDX detector, with nitrogen gas, on a Carl Zeiss Sigma field emission SEM was used for the analysis. The samples of grass were dried in an oven at 40 °C for four hours. Once the samples were dried, the grass was mounted onto stubs using electrically conductive carbon tape. The grass was then coated with

electrically conductive mixture of 60% gold and 40% palladium. The samples were analysed using the SEM to a magnification of 100 μm , 20 μm , 10 μm and 2 μm .

3.4.9 X-ray fluorescence (XRF) analysis

The elemental composition of the grass was determined by conducting an XRF analysis with a Philips PW2404 X-ray spectrometer using the technique of Norrish Fusion. X-ray photons of a certain wavelength are released when atoms have been excited by X-rays. Each atom will emit a characteristic wave length and the elemental composition of the grass can be determined by the XRF analyser. The grass was prepared by drying it in an oven at 80 °C for six hours. Once the grass was dried it was placed into a muffled furnace at 500 °C for 45 minutes. An XRF analysis was then conducted on the ash.

3.4.10 Methods used for the characterisation of Indigenous South African Grass using the Weender analysis

Nvirotek Laboratories, South Africa determined the composition of indigenous South African grass for the extended Weender analysis. Nvirotek Laboratories conducted a Near-Infrared Spectroscopy (NIRS) in order to determine the moisture, crude fibre, crude fat, crude protein, ash, neutral detergent fibre (NDF), acid detergent fibre (ADF) and the acid detergent lignin (ADL) content of the grass.

3.5 Modelling methods

3.5.1 Upgrading the ADM1 to include sulfate reduction

The ADM1 is a comprehensive detailed model able to predict the biochemical and physiochemical processes during AD. However, the model neglects to predict anaerobic digestion for sulfate-rich wastewater. Including sulfate reduction in the model posed difficulties due to competition by SRB for organic substrates and hydrogen (Batstone et al., 2002). The process of DSR involves the reduction of sulfate to sulfide by SRB coupled with an electron donor (Ramla & Sheridan, 2015). During AD, sulfate reducers compete with acidogenic bacteria, acetogenic bacteria and methanogens for volatile fatty acids (VFAs) and H_2 . Organisms use the VFAs produced during AD and as a result the different bacterial groups will compete for organic acids (Batstone, 2006).

The H_2S produced during DSR is an inhibitory agent in the range of $0.05\text{--}0.43 \text{ kgS m}^{-3}$ and hinders biological process including acidogenesis, acetogenesis, methanogenesis and DSR (Rinzema & Lettinga, 1988). The effect of dissolved sulfide inhibition also needs to take into account the high solubility (0.1 M bar^{-1}) of H_2S (Speece, 1983). The acid-base association/dissociation of sulfide distributes into the components of H_2S , HS^- and S^{2-} . If there is a presence of heavy metals, a precipitate of metal sulphides will form at a pH range of $6.5 - 8$ as dissolved sulphides (S^{2-}) occur at this pH range (Rinzema & Lettinga, 1988). The formation of metal sulfides leads to a decrease of S^{2-} concentration in the system.

Batstone (2006) developed a simple model for wastewaters containing low sulfate concentrations assuming that the reduction of sulfate by SRB is achieved through utilizing available hydrogen. For systems that contain a ratio higher than 0.1 gS gCOD^{-1} SRB will compete for butyrate, propionate and acetate. The work presented in this thesis aims to combine previous conducted studies in order to provide a more comprehensive extension of the ADM1 to include sulfate reduction for sulfate-rich wastewater particularly focusing on AMD.

3.5.1.1 The stoichiometry and kinetics of sulfate reduction

Fedorovich et al. (2003) stated that acidogenic bacteria outcompete SRB for sugars, amino acids and long fatty chain acids (LFCA) due to the rapid-growth of the fermentative bacteria and can be neglected from the updated ADM1. SRB compete against acetogenic bacteria and methanogens for syntrophic substances (organic acids) and H_2 and have to be considered in the model. The levels of competition for organic substrates and H_2 are depicted in Figure 3.6.

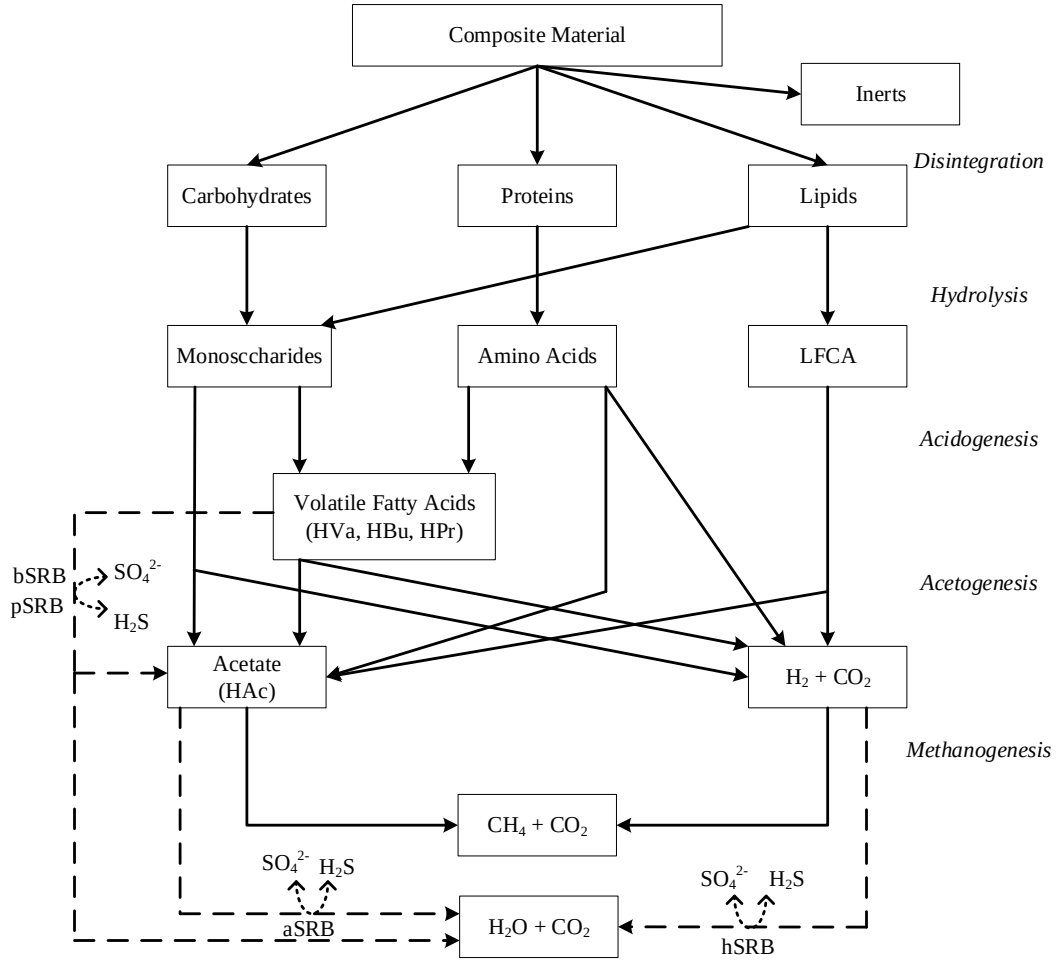
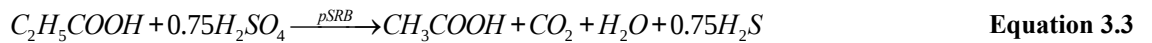
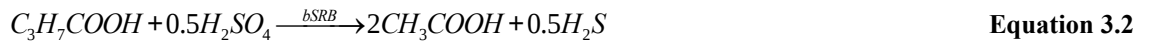


Figure 3.6 The anaerobic model of the extended ADM1 to include sulfate reduction adapted from (Batstone et al., 2002) and (Barrera et al., 2013)

The reactions described in Equation 3.2 – Equation 3.5 demonstrate the uptake of VFA's and H_2 by butyrate-degrading (X_{bSRB}), propionate-degrading (X_{pSRB}), acetate-degrading (X_{aSRB}) and hydrogenotrophic-degrading sulfate reducing bacteria (X_{hSRB}).



The kinetics of the DSR process takes into account the concentrations of both SO_4^{2-} (the electron acceptors) and the electron donors (VFAs and H_2) following Monod type kinetics. The last two terms of Equation 3.6 includes the inhibition function of both pH and sulfide inhibition.

$$r_{uptake,j} = k_{m,j} \times \frac{S_i}{K_{s,j} + S_i} \times \frac{S_{SO_4}}{K_{SO_4,j} + S_{SO_4}} \times X_j \times I_{pH} \times I_{sulfide} \quad \text{Equation 3.6}$$

Hydrogen sulfide can affect and breakdown cell membranes (Batstone et al., 2002) and therefore the inhibition of H₂S on all biological processes needs to be considered when extending the ADM1 to include sulfate reduction. Fedorovich et al. (2003) implemented Equation 3.7 as an approximation to determine sulfide inhibition, however this function can be instable and Equation 3.8 considered by (Poinapen & Ekama, 2010) provides a function which has a gradual approach to zero (Barrera et al., 2013).

$$I_{sulfide} = 1 - \frac{S_I}{K_I} \quad \text{Equation 3.7}$$

$$I_{sulfide} = \left[1 - \left(\frac{S_I}{0.60056 K_I} \right)^2 \right] \quad \text{Equation 3.8}$$

Both the overall function and kinetics of the cell can be affected by the pH. Batstone et al. (2002) recommended two functions for pH inhibition which consider both the upper and lower pH limits of the cell. The functions described in Equation 3.9 and Equation 3.10 should be implemented when both the upper and lower pH inhibitions should be taken into consideration and when just the lower inhibition applies respectively. The different parameters for upper pH and lower pH inhibitions for each process is giving in Appendix A.

$$I_{pH} = \frac{1 + 2 \cdot 10^{0.5(pH_{LL} - pH_{UL})}}{1 + 10^{(pH - pH_{UL})} + 10^{(pH_{LL} - pH)}} \quad \text{Equation 3.9}$$

$$I_{pH} = \exp \left[-3 \left(\frac{pH - pH_{UL}}{pH_{UL} - pH_{LL}} \right)^2 \right] \quad \left\| \begin{array}{l} pH < pH_{UL} \\ I_{pH} = 1 \end{array} \right. \quad \text{Equation 3.10}$$

The switch functions proposed by Batstone et al. (2002) results in discontinuities and numerical instabilities while implementing large ranges of time constants to a model (Rosén & Jeppsson, 2006). The pH based Hill function was chosen in order to determine the pH inhibition of the system as suggested by Rosén & Jeppsson (2006).

$$I_{pH} = \frac{pH^n}{pH^n + K_{pH}^n} \quad \text{Equation 3.11}$$

Where;

$$K_{pH} = \frac{pH_{LL} + pH_{UL}}{2} \quad \text{Equation 3.12}$$

The approximation of n can be determined implementing the upper pH and lower pH of each species as shown in Equation 3.13.

$$n_i = \frac{3.0}{pH_{UL,i} - pH_{LL,i}} \quad \text{Equation 3.13}$$

The decay rate ($\rho_{decay,j}$) of the SRB follows the same function as all other bacterial groups in the ADM1 and is described as the biomass concentration (X_i) and the specific endogenous mass loss rate ($k_{dec,i}$) (Batstone et al., 2002).

$$r_{decay,j} = k_{dec,j} \times X_i \quad \text{Equation 3.14}$$

3.5.1.2 Acid-base association/dissociation

The acid-base association/dissociation plays a vital role in determining the overall pH of the system in the ADM1. The pH of the system can be determined by performing a charge balance on the all the ionic species. Therefore, when upgrading the ADM1 to include sulfate reduction it is useful to determine the concentration of the dissociated species $[SO_4^{2-}]$, $[HS^-]$ and $[S^{2-}]$ and the effect these ionized compounds have on the charge balance. The sulfate acid/base system involving the associated/dissociated compounds $H_2SO_4^{2-}/HSO_4^-/SO_4^{2-}$ and the sulfide acid/base system involving $H_2S/HS^-/S^{2-}$ were included in extension of the ADM1.

Due to the fact that sulphuric acid is a strong acid it is assumed to completely dissociate and only HSO_4^- is present in the system; Equation 3.15 can be neglected from the model. The equilibrium value for the dissociation reaction of HSO_4^-/SO_4^{2-} is extremely low ($pK=1.99$). However due to the AMD exhibiting the characteristics of low pH (2-3), Equation 3.16 was included in the extension of the ADM1.



The dissociation of hydrogen sulfide can once again be assumed to behave as a monoprotic acid-base as the pK of the dissociation reaction of HS^-/S^{2-} is high (pK=13.88) and the system will not surpass a pH greater than 9, therefore only Equation 3.18 was included in the kinetic model.



Batstone et al. (2002) defined the equations for determining the acid-base equilibrium for all individual acid-base pairs in algebraic equations. This presented confusion when implementing these kinetic rates as ordinary differential equations (ODEs). This research implemented the methods described by Rosén & Jeppsson (2006) by defining the kinetic equations in terms of the total of the species and one acid-base pair. The concentration of the dissociated species is determined by the changes in both the concentration of the total amount of the species and the pH as shown in Equation 3.19 and Equation 3.20. The value for $K_{A/B,i}$ was assigned to be $10^{10} \text{ M}^{-1} \text{ d}^{-1}$ as suggested by Rosén & Jeppsson (2006).

$$r_{A,SO_4} = K_{A/B_{SO_4}} \left(S_{SO_4, total} \times S_{H^+} - (K_{a,SO_4} + S_{H^+}) \right) \quad \text{Equation 3.19}$$

$$r_{A,H_2S} = K_{A/B_{H_2S}} \left(S_{HS^-} \times (S_{H^+} + K_{a,H_2S}) - K_{a,H_2S} \times S_{H_2S, total} \right) \quad \text{Equation 3.20}$$

The gas transfer rates follow the Van't Hoff equation and apply Henry's law in order to obtain the concentration of the gas in the system. Henry's constant for hydrogen sulfide was taken to be $K_{H(25^\circ \text{C}), H_2S} = 0.105 \text{ kmol m}^{-3} \text{ bar}^{-1}$ (Rosén & Jeppsson, 2006).

3.5.2 Weak acid-base chemistry

The model has been updated, by (Solon et al., 2015), to include the aqueous phase chemistry of chemical species S_i (Kmol m^{-3}) and chemical components $S_{j,}$ (Kmol m^{-3}). The chemical components S_j considered in the model include: acetic acid (S_{ac^-}), butyric acid (S_{bu^-}), propionic acid (S_{pro^-}), valeric acid (S_{va^-}), carbonate ($S_{co_3^{2-}}$), ammonium ($S_{NH_4^+}$), nitrate ($S_{NO_3^-}$), nitrite ($S_{NO_2^-}$), sulfate ($S_{SO_4^{2-}}$), sulfide (S_{HS^-}), iron (II) ($S_{Fe^{2+}}$) and

iron (III) (S_{Fe3^+}). The chemical species S_i is the protonized and deprotonized states (Solon et al., 2015; Flores-Alsina et al., 2015) for the set of chemical components S_j listed above. For each species S_i a law of mass equation (Equation 3.21) and for each component S_j a mass balance (Equation 3.22) are included to determine the equilibrium (Solon et al., 2015; Flores-Alsina et al., 2015).

$$a_i = K_i \prod_{j=1}^{N_c} a_j^{v_{i,j}} \quad j = 1, 2, \dots, N_c \quad \text{Equation 3.21}$$

$$S_{j,tot} = S_j + \sum_{i=1}^{N_{SP}} v_{i,j} S_i = \frac{a_j}{\gamma} + \sum_{i=1}^{N_{SP}} v_{i,j} \frac{a_j}{\gamma} \quad i = 1, 2, \dots, N_{SP} \quad \text{Equation 3.22}$$

The set of non-linear equation, described above, are used to formulate the equilibrium equations. The activity of the i th species and the j th component is denoted by a_i and a_j respectively and γ is the activity coefficient (Solon et al., 2015; Flores-Alsina et al., 2015). N_{SP} and N_c represent the number of species and the number of components respectively. The equilibrium K_i is adjusted using the van't Hoff equation shown in Equation 3.23 (Solon et al., 2015; Flores-Alsina et al., 2015).

$$\ln \frac{K_2}{K_1} = \frac{DH}{R} \left(\frac{1}{T_1} - \frac{1}{T_2} \right) \quad \text{Equation 3.23}$$

The charge balance of the cations S_{cat} and anions S_{an} can be used to fully specify the set of algebraic equations (Solon et al., 2015; Flores-Alsina et al., 2015).

$$\sum S_{cat} - \sum S_{an} = 0 \quad \text{Equation 3.24}$$

3.5.3 Iron Transformations

Iron (III) is chemically converted to iron (II) in an anaerobic environment (Flores-Alsina et al., 2016). Mamais et al. (1994), Nielsen et al. (2005) and Zhang et al. (2008) have all reported that the reduction of Fe^{3+} and Fe^{2+} takes place rapidly and can occur fully in a few hours from the start of the process. Sulfides and hydrogen present in the system are used as electron donors for the conversion of Fe^{3+} to Fe^{2+} . The model compensates for the reduction iron (III) to iron (II) and adjusts the pH accordingly (Solon et al., 2015; Flores-Alsina et al., 2015). Biochemical rates and stoichiometric coefficients are provided

in Appendix A. At neutral pH ranges, Fe^{2+} reacts with sulfide and precipitates as ferrous sulfide (FeS) as shown in the Equation 3.25 (Mbamba et al., 2015):

$$r_{\text{crist}} = k_{\text{crist, FeS}} X_{\text{FeS}} \left[\left(\frac{a_{\text{Fe}^{2+}} - \partial_{\text{S}^{2-}}}{K_{\text{sp, FeS}}} \right)^{\frac{1}{2}} - 1 \right]^2 \quad \text{Equation 3.25}$$

The mineral precipitation rate (r_{crist}) ($\text{kmol m}^{-3} \text{ day}^{-1}$) is the product of the particulate concentration of Ferrous Sulfide (X_{FeS}) and the rate coefficient of FeS ($k_{\text{crist, FeS}}$) (day^{-1}) (Mbamba et al., 2015). $K_{\text{sp, FeS}}$ is the solubility factor of ferrous sulfide and $a_{\text{Fe}^{2+}}$ is the activity coefficient of iron(II). The activity coefficient of sulfide is determined by:

$$\gamma_{\text{S}^{2-}} = \frac{a_{\text{HS}}}{a_{\text{S}}} \quad \text{Equation 3.26}$$

3.5.4 Leach Bed Reactor (LBR)

A batch reactor design equation was used to describe the bioremediation of AMD the breakdown of the indigenous South African grass. Previously in the ADM1, the composite material (X_c) described the breakdown in to carbohydrates (X_{ch}), proteins (X_{pr}), lipids (X_{li}) and particulate inerts (X_i) through disintegration using first order kinetics. The disintegration of the composite material (X_c) has been removed and the products (X_{ch} , X_{pr} , X_{li} and X_i) have been mapped directly into the model (Flores-Alsina et al., 2016). For each liquid phase and solid phase component the mass balance is as follows:

$$\frac{dS_{\text{liq},i}}{dt} = \sum_{j=1-39} \rho_j v_{i,j} \quad \text{Equation 3.27}$$

$$\frac{dX_i}{dt} = \sum_{j=1-39} \rho_j v_{i,j} \quad \text{Equation 3.28}$$

Where $v_{i,j}$ is the biochemical rate coefficient and ρ_j is the process kinetic rate equation. All biochemical rate coefficients ($v_{i,j}$) and process kinetic rate equations (ρ_j) are provided in supplementary matrices given in Appendix A. The mass balance on the gasses H_2 , CH_4 , CO_2 , H_2S is shown in Equation 3.29.

$$\frac{dS_{gas,i}}{dt} = R_{T,i} \frac{V_{liq}}{V_{gas}} \quad \text{Equation 3.29}$$

Where $P_{T,i}$ is the liquid/gas transfer kinetic rate which can be found in Appendix A. State variables are listed in Table 3.6.

Table 3.6 State variables as obtained from the IWA

Soluble Substrates			Particulate Substrates		
i		Unit	i		Unit
1	Monosaccharaides (S_{su})	kg COD m ⁻³	14	Composites (X_c)	kg COD m ⁻³
2	Amino acids (S_{aa})	kg COD m ⁻³	15	Carbohydrates (X_{ch})	kg COD m ⁻³
3	LFCA (S_{fa})	kg COD m ⁻³	16	Proteins (X_{pr})	kg COD m ⁻³
4	Total valerate (S_{va})	kg COD m ⁻³	17	Lipids (X_{li})	kg COD m ⁻³
5	Total butyrate (S_{bu})	kg COD m ⁻³	18	Sugar degraders (X_{su})	kg COD m ⁻³
6	Total propionate (S_{pro})	kg COD m ⁻³	19	Amino acid degraders (X_{aa})	kg COD m ⁻³
7	Total acetate (S_{ac})	kg COD m ⁻³	20	LFCA degraders (X_{fa})	kg COD m ⁻³
8	Hydrogen (S_{h2})	kg COD m ⁻³	21	Valerate & butyrate degraders (X_{c4})	kg COD m ⁻³
9	Sulfate (S_{so4})	Kmol m ⁻³	22	Butyrate SRB (X_{bSRB})	kg COD m ⁻³
10	Methane (S_{ch4})	kg COD m ⁻³	23	Propionate degraders (X_{pro})	kg COD m ⁻³
11	Total inorganic sulfide (S_{is})	kg COD m ⁻³	24	Propionate SRB (X_{pSRB})	kg COD m ⁻³
12	Inorganic carbon (S_{ic})	Kmol C m ⁻³	25	Acetate degraders (X_{ac})	kg COD m ⁻³
13	Inorganic nitrogen (S_{in})	Kmol N m ⁻³	26	Acetate SRB (X_{aSRB})	kg COD m ⁻³
Ion State Variable			27	Hydrogen degraders (X_{h2})	kg COD m ⁻³
i		Unit	28	Hydrogenotrophic SRB (X_{hSRB})	kg COD m ⁻³
31	Cations (S_{cat})	Kmol m ⁻³	29	Particulate inerts (X_i)	kg COD m ⁻³
32	Anions (S_{an})	Kmol m ⁻³	30	Ferrous sulfide (X_{fes})	Kmol FeS m ⁻³
33	Hydrogen ion (S_{h+})	Kmol H ⁺ m ⁻³	Gas State Variable		
34	Iron (II) (S_{fe2+})	kg COD m ⁻³	36	Hydrogen ($S_{gas,h2}$)	Kmol H ₂ S m ⁻³
35	Iron (III) (S_{fe3+})	kg COD m ⁻³	37	Carbon dioxide ($S_{gas,co2}$)	Kmol CO ₂ m ⁻³
			38	Methane ($S_{gas,ch4}$)	Kmol CH ₄ m ⁻³
			39	Hydrogen Sulfide ($S_{gas,h2s}$)	Kmol H ₂ S m ⁻³

The model was simulated in MATLAB® and Simulink® as shown in Appendix C. An ODE15s solver was used in the implementation of the model. The results of the model were compared to the experimental results of the synthetic AMD leachate and AMD leachate to test the validity of the model.

4 RESULTS AND DISCUSSION

4.1 Optimal Solid/Liquid Ratio for Anaerobic Digestion and Increased Sulfate Removal Efficiencies

In order for anaerobic digestion to optimally take place the preferred optimal solid/liquid ratio is required. The main objectives of the experiment were:

- an increase in pH of AMD;
- a decrease in sulfate concentration of AMD; and
- A decrease the availability of COD.

The availability of COD in the system signifies the concentration of organic compounds that are available as food source for the sulfate reducing bacteria. The starting solid/liquid ratio was 1.5 g grass/L synthetic AMD. This ratio was adopted from Thamsirioj et al. (2012). The focus of their research was to develop a process to produce methane through AD using grass silage and a neutral leachate in sequentially fed LBRs and an UASB model configuration. Due to the fact that a neutral leachate without the presence of sulfates was used by Thamsirioj et al. (2012), this ratio was used only as a starting point and this research tested two other ratios of 6 and 30.

Over a period of 30 days, the grass was acid washed using synthetic AMD in the reactor configuration as shown in Figure 3.1 at different solid/liquid ratios. Synthetic AMD was used to test all three solid/liquid ratios.

Figure 4.1 shows the change in pH for all three solid/liquid ratios over the period of 30 days. The lowest solid to liquid ratio had no significant increase in pH over time. The initial pH was 2.83, the pH remained acidic (below 3.5) and did not increase substantially for the duration of the experiment. The pH for both the solid/liquid ratios of 6 and 30 increased to neutral. The pH for both solid/liquid ratios remained between 6 and 8 for the duration of the run time. It can be concluded that having a higher solid/liquid ratio is optimal for the treatment of AMD and at higher ratios there is a rapid increase in the pH. The pH change could be a result of the neutralising ability of the grass determined by the X-ray fluorescence (XRF) analysis of the grass.

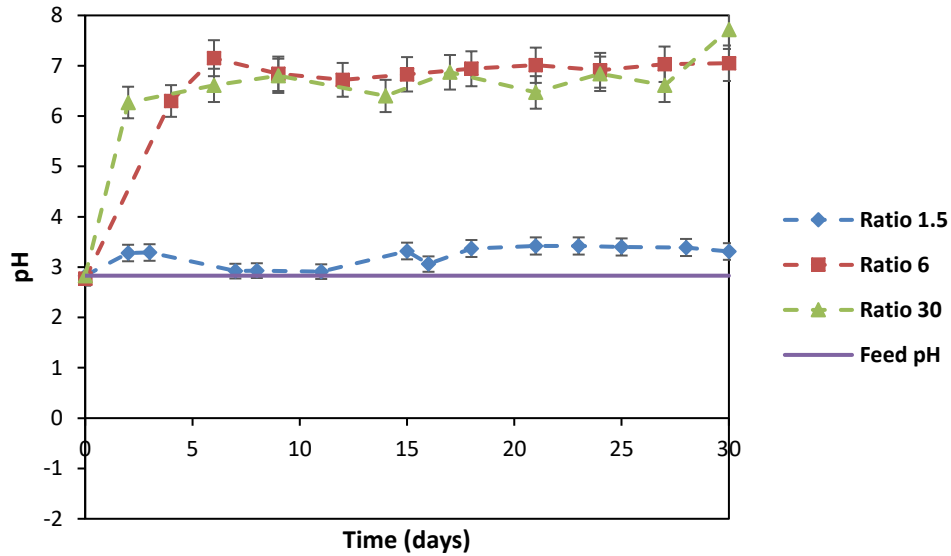


Figure 4.1 pH vs time for the solid/liquid ratios 1.5, 6 and 30

An X-ray fluorescence (XRF) analysis was performed on the grass in order to determine elemental composition of the grass before and after the experiments.

Table 4.1 depicts the percentage of cations present on the grass at day 0 (before the experiment was conducted) and at day 30 (after the experiment was conducted) for the solid/liquid ratio of 6. The pH remained relatively neutral at this ratio over the period. The amount of heavy metals, present of the grass, increased. Indigenous South African grass is an inexpensive sorbent and dead biomass has been used for the removal of heavy metals in wastewater through the process of adsorption (Bhattacharyya & Sen Gupta, 2006; Motsi et al., 2009). Through adsorption, heavy metals are exchanged with cations on the surface of the grass. At a low pH, H^+ ions are preferentially adsorbed from synthetic AMD than heavy metals (Motsi et al., 2009). Hydrogen ion is a strongly adsorbed cation and has a higher binding strength than potassium ions (Inglezakis et al., 2003). It should be noted that during the experimental runs, the pH of the solution increased to above 6 after an hour of run time. Table 4.1 shows a large decrease in potassium (K^+) ions (an exchangeable cation) which contributed to the decrease of hydrogen ions in the synthetic AMD solution through cation exchange. The adsorption of H^+ ions accounted for the rapid increase in pH at the start of the experiment. Figure 4.1 concluded that the indigenous South African grass present in the system has a significant role to play in the change in the concentration of hydrogen ions. Further

investigation needs to be made into the adsorption of H^+ ions and the cation exchange of potassium ions with hydrogen ions to fully explain the rapid increase of the pH in the system.

Table 4.1 Percentage of Cations present in grass using a solid/liquid ratio of 6 before and after experiment was conducted

Cation	Cation% at day 0	Cation% at day 30
Si	25.507	35.142
Al	0.743	1.192
Fe	0.397	0.618
Mn	0.731	0.529
Mg	6.243	4.119
Ca	6.591	5.066
Na	3.946	0.454
K	11.684	3.051
Ti	0.062	0.111
P	1.505	1.142
Cr	0.003	0.006
Ni	0.011	0.007

The sulfate concentration profile of the experiment for the solid/liquid ratio is shown in Figure 4.2. Large quantities of synthetic AMD had to be prepared to run the experiments and producing synthetic AMD solutions with exactly the same initial sulfate concentrations was difficult. The sulfate concentrations were hence indexed in order to provide a clear analysis of the data. Figure 4.2 indicates that the concentration of sulfate decreased for all three solid/liquid ratios.

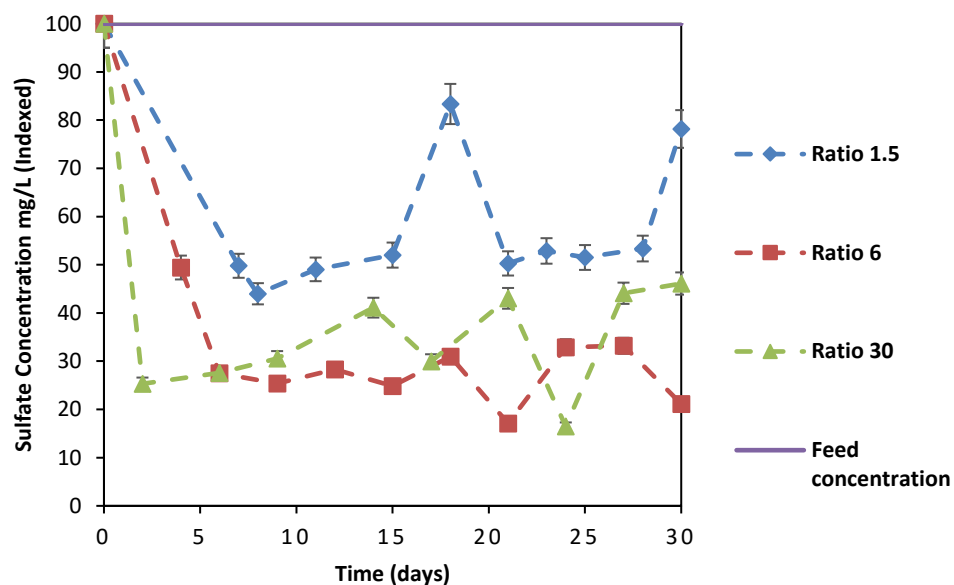


Figure 4.2 Sulfate concentration for the solid/liquid ratios 1.5, 6 and 30

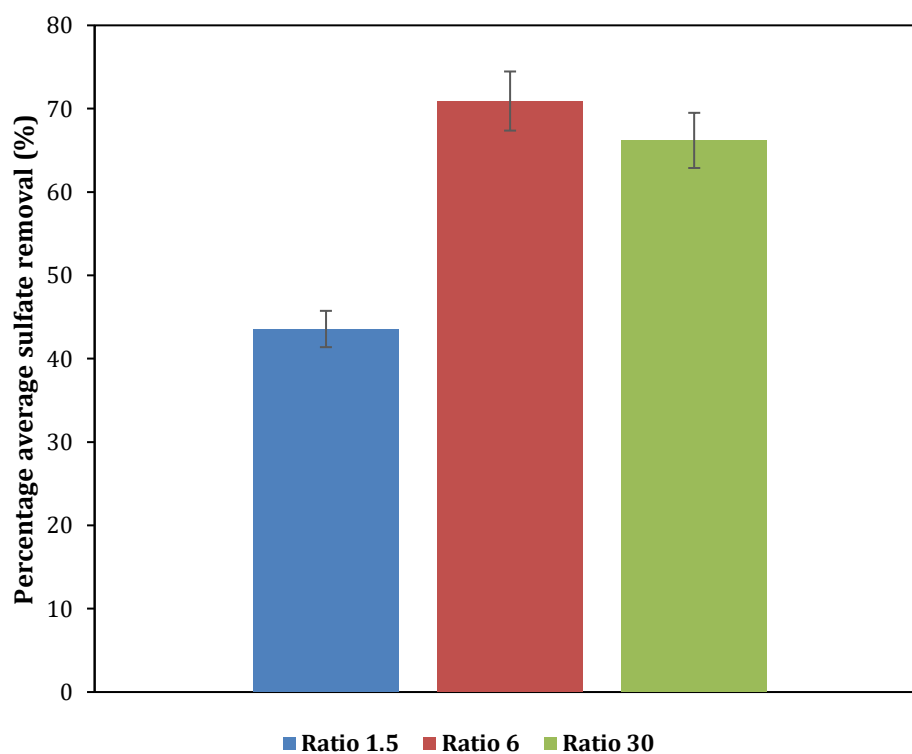


Figure 4.3 The percentage average sulfate removed for solid/liquid ratios of 1.5, 6 and 30

Figure 4.3 represents the average percentage sulfate removed for all tested solid/liquid ratios. As expected, the lowest solid/liquid ratio removed the least sulfate from the system

(43.5%). This is due to the low pH shown in Figure 4.1. At a low pH, SRB are inhibited and therefore there will be a lower rate of conversion of sulfate to sulphides (Barrera et al., 2013). However, a solid/liquid ratio of 6 yielded a greater percentage of sulfate removed (70.9%) than that of a solid/liquid ratio of 30 (66.19%). At higher solid/liquid ratios, more soluble organic food sources become readily available. There will be a greater growth of methanogenic bacteria in the system which outcompetes the SRB for organic food sources and decreases the removal of SRB (Fedorovich et al., 2003).

The relationship between the removal of sulfate and the production of COD is illustrated in Figure 4.4. For the purpose of this analysis, the sulfate concentration for all three solid/liquid ratios was compared to the concentration of soluble COD produced per gram of grass.

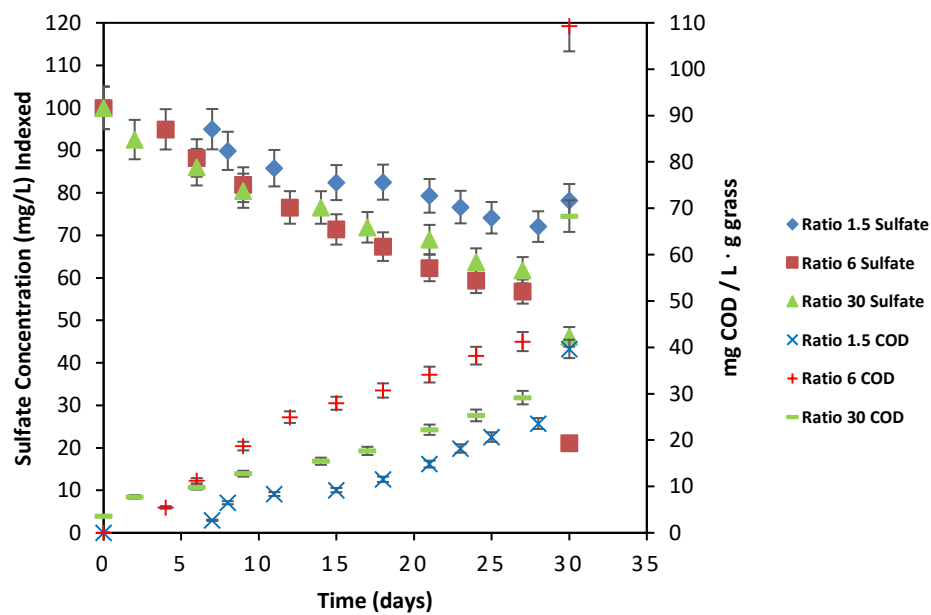


Figure 4.4 The relationship between the removal of Sulfate and the production of COD

It is evident that there is a direct relationship between the removed sulfate concentration and the amount of COD produced. As more soluble COD becomes readily available as an organic food source for the sulfate-reducing bacteria more sulfate is removed by the sulfate-reducing bacteria and converted into sulfide. The decrease of the sulfate concentration is dependant of the amount of soluble COD present in the system.

Figure 4.5 shows the concentration of soluble COD produced per a gram of grass. Before the analysis was conducted samples were placed in a centrifuge in order to separate the liquids from solid particles to prevent grass from influencing the COD test. The sample was then passed through 0.45 μm filter to remove any remaining solid particles. The dissolved COD was comprised of monosaccharides (S_{su}), amino acids (S_{aa}), long fatty chain acids (S_{fa}), valerate (S_{va}), butyrate (S_{bu}), propionate (S_{pro}) and acetate (S_{ac}) (Batstone et al., 2002; Barrera et al., 2013; Fedorovich et al., 2003).

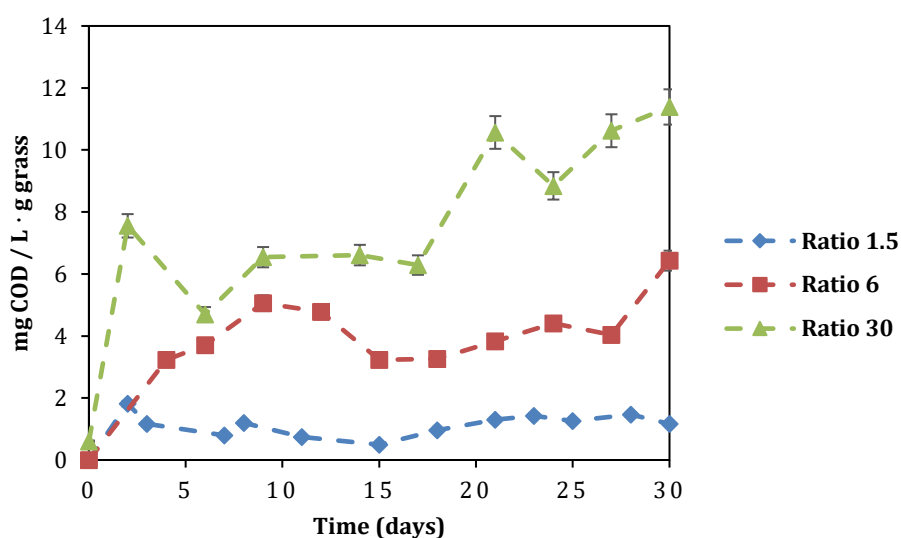
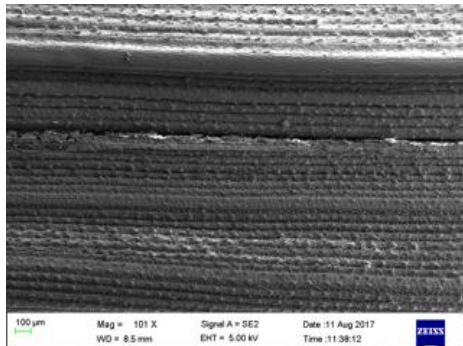


Figure 4.5 Soluble COD produced per gram of grass for the solid to liquid ratio of 1.5, 6 and 30

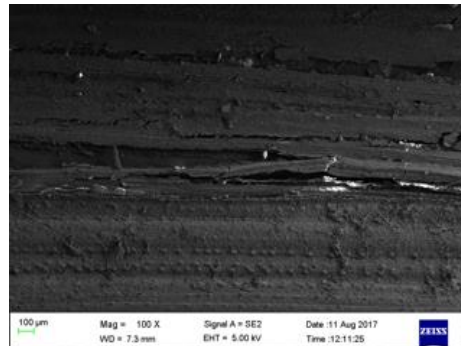
Initially, it was assumed that acid hydrolysis would take place due to the low pH of AMD and this would cause the grass to break down exposing the cellulose and semi-cellulose and allowing the production of monosaccharides and other organic compounds. It is evident from Figure 4.5 that at a solid/liquid ratio of 1.5, acid hydrolysis produces only a small concentration of soluble COD. The experiment was undertaken at room temperature and acid hydrolysis would yield a greater concentration of soluble COD at significantly higher temperatures (Burman et al., 2018). Figure 4.1 illustrates that the pH remains relatively neutral using a liquid/solid ratio of 6 and 30. At a neutral pH, the growth of bacteria is not inhibited, and it can be posited that the breakdown of the organic substrate into soluble COD is due to bacterially induced enzymatic hydrolysis. Grass from the experiments using a solid/liquid ratio of 1.5 and 6 was analysed under a scanning electron microscope (SEM). Figure 4.6 and Figure 4.7 illustrates the breakdown of the grass structures from day 0 to day 30. The images compare and contrast the acid

hydrolysis of the grass at a solid/liquid ratio of 1.5 and the enzymatic hydrolysis of the grass at a solid/liquid ratio of 6.

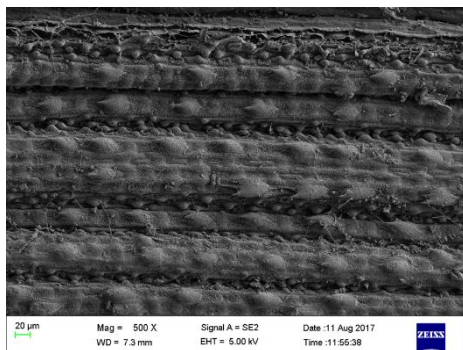
(a) Ratio 1 day 0 (100x)



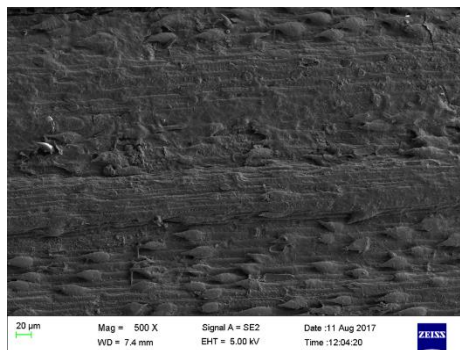
(b) Ratio 1 day 30 (100x)



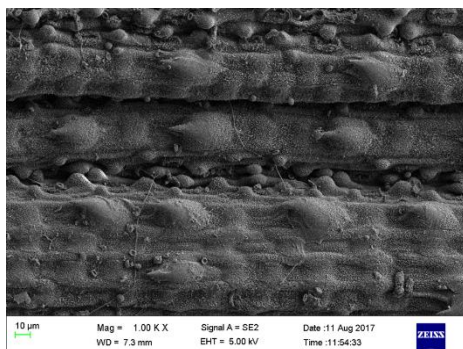
(c) Ratio 1 day 0 (500x)



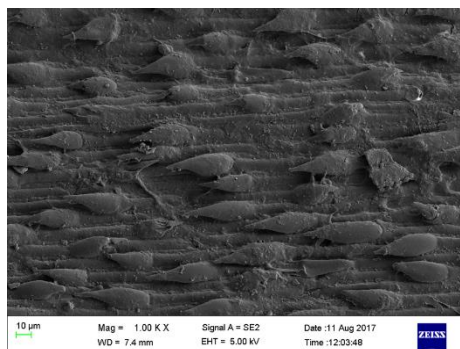
(d) Ratio 1 day 30 (500x)



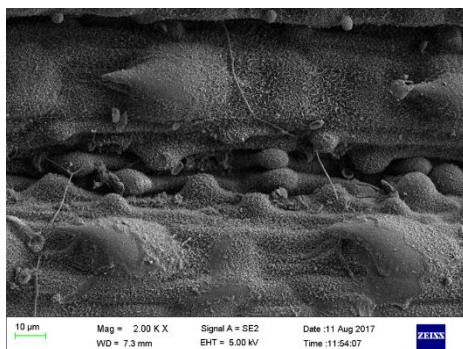
(e) Ratio 1 day 0 (1000x)



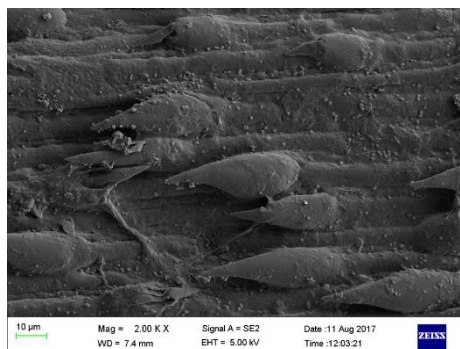
(f) Ratio 1 day 30 (1000x)



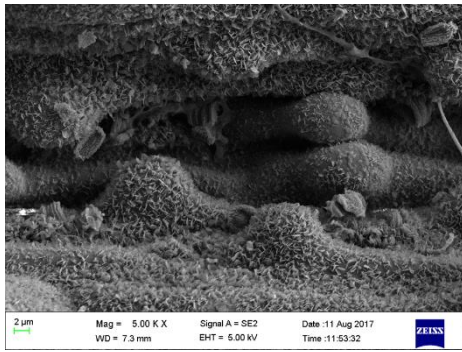
(g) Ratio 1 day 0 (2000x)



(h) Ratio 1 day 30 (2000x)



(i) Ratio 1 day 0 (5000x)



(j) Ratio 1 day 30 (5000x)

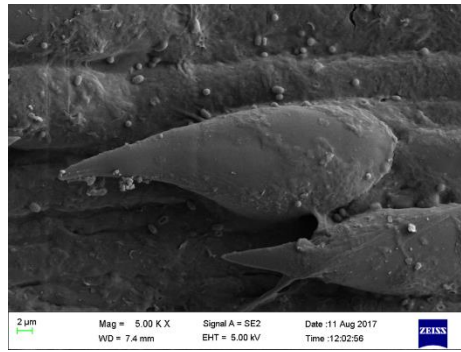
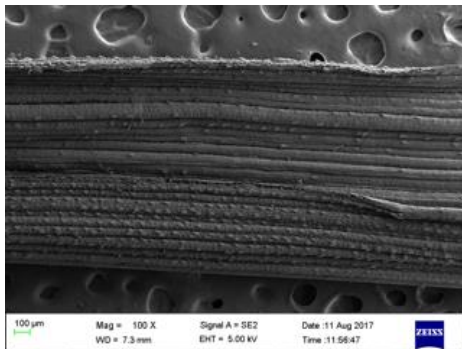
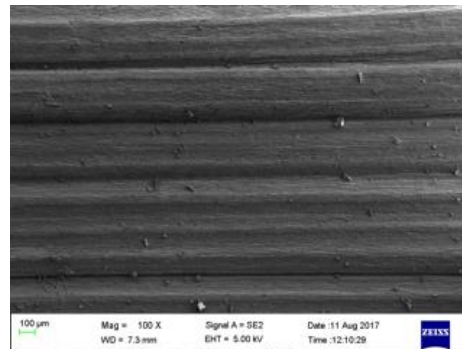


Figure 4.6 SEM analyses of grass, before and after the experiment was conducted, for the solid/liquid ratios of 1.5 at a magnification of 100x (a-b), 500x (c-d), 200x (g-h) and 5000x (i-j).

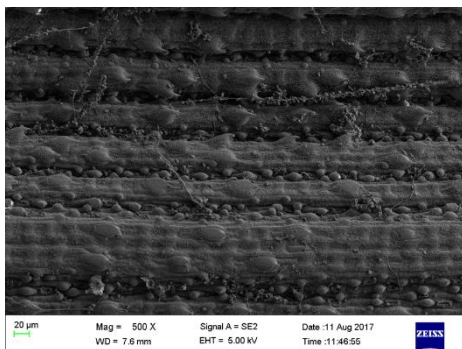
(a) Ratio 6 day 0 (100x)



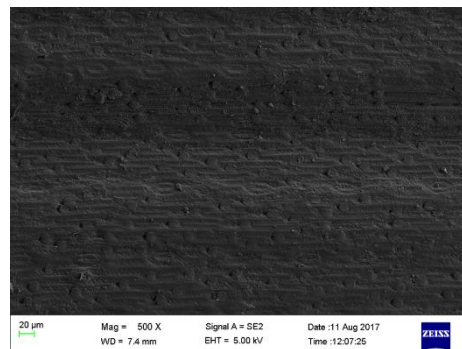
(b) Ratio 6 day 30 (100x)



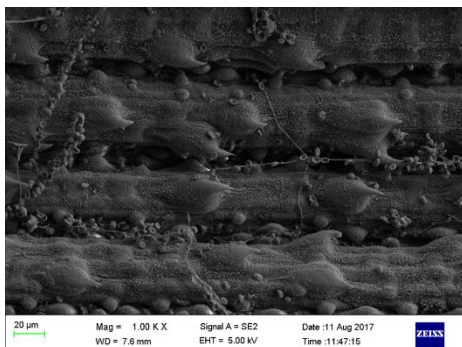
(c) Ratio 6 day 0 (500x)



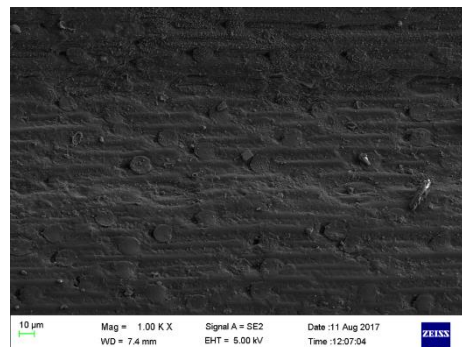
(d) Ratio 6 day 30 (500x)



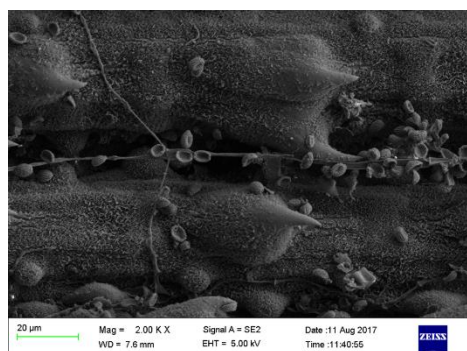
(e) Ratio 6 day 0 (1000x)



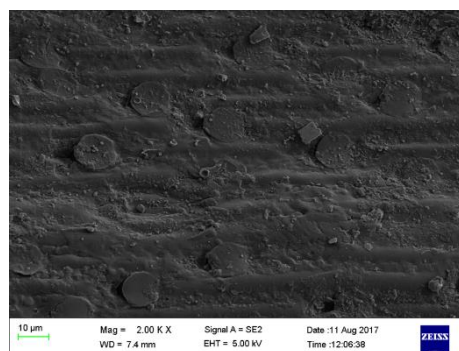
(f) Ratio 6 day 30 (1000x)



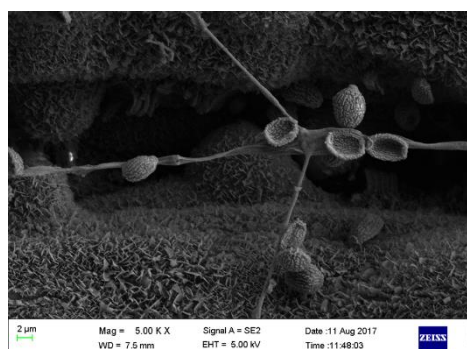
(g) Ratio 6 day 0 (2000x)



(h) Ratio 6 day 30 (2000x)



(i) Ratio 6 day 0 (5000x)



(j) Ratio 6 day 30 (5000x)

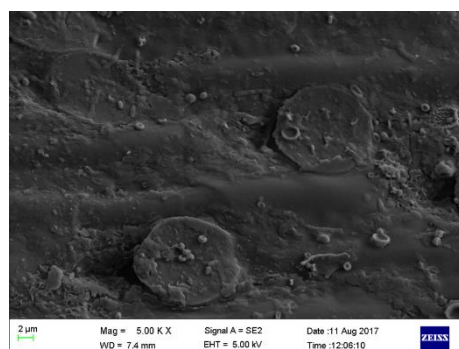


Figure 4.7 SEM analyses of grass, before and after the experiment was conducted, for the solid/liquid ratios of 6 at a magnification of 100x (a-b), 500x (c-d), 1000x (e-f), 200x (g-h) and 5000x (i-j).

The grass SEM micrographs were magnified to various scales in order to show the variations of the structures of the grass when undergoing enzymatic (ratio 6) and acid (ratio 1.5) hydrolysis. A similar trend was depicted by Tsapekos et al. (2017) when determining the effects of mechanical pre-treatment of ensiled meadow grass for the production of methane. The images showed a greater structural breakdown of the grass when undergoing enzymatic hydrolysis. Due to the fact that the grass structure is more rigorously broken down by enzymatic hydrolysis, the biodegradable organic compounds are exposed as a food source for the bacteria as there is a larger exposed surface area of the grass. This contributed to a larger availability of soluble COD being produced when there was a larger solid/liquid ratio, resulting in a higher removal of sulfates. A comparison of the Bioremediation of AMD and Synthetic AMD. . It can be seen in Figure 4.7, that at low temperatures (as is the case here) there is little difference between the untreated and treated grass. This implies that little to none observable breakdown of the grass by sulfuric acid occurred and that the rate of hydrolysis is very slow . Conducting

the experiment at higher temperatures would lead to a higher rate of hydrolysis and a further breakdown of the grass (Burman et al., 2018).

In order to compare the bioremediation of AMD and synthetic AMD the experiments described in section 4.1 were repeated. The first reactor contained AMD leachate which was obtained from the Komdraai dam, Limpopo South Africa. The second reactor contained a synthetic AMD solution as the leachate. The reaction configuration is shown in Figure 3.1 in Section 3.1.

Both reactors contained a solid/liquid ratio of 30. This ratio was chosen as the availability of AMD was limited and a ratio of 30 required a smaller volume of liquid. However, over the course of the experiment the pH did not rise to neutral for the AMD leachate as shown in Figure 4.8.

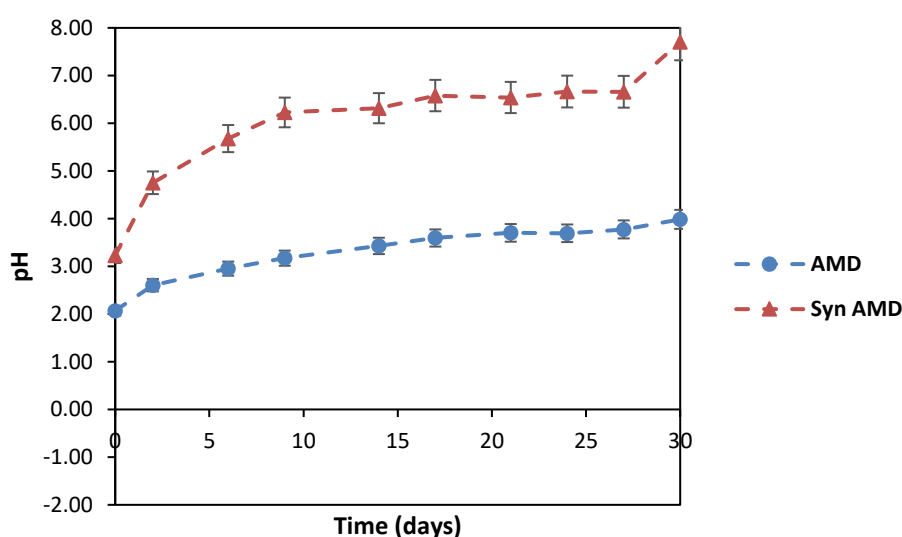


Figure 4.8 Averaged pH vs time for AMD and Synthetic AMD

The pH for the synthetic AMD increased to above 6 after 9 days and remained relatively neutral throughout the course of the experiment. The pH of the synthetic AMD increased again after 27 days. The pH of the AMD gradually increased but remained acidic throughout the course of the experiment, never surpassing a pH of 4.5. This is likely due to the complex chemistry and heavy metal content of AMD. An Inductively Coupled Plasma (ICP) elemental analysis was conducted and the results are shown in Table 4.2. The full ICP analysis is presented in Appendix B.

An examination has been made into the initial heavy metal composition of AMD and the total acidity of AMD. The term ‘acid’ refers to the concentration of H^+ ions while ‘acidity’ takes into account both the concentration of H^+ ions and the mineral acidity (Moodley et al., 2018). Mineral acidity is the likelihood of the production of H^+ ions through the formation of metal hydroxides (Moodley et al., 2018; Taylor et al., 2005).

Table 4.2 Heavy metal concentrations in AMD (ICP analysis)

Element	Heavy metal concentration (mg/L)
Mn	266.4
Fe	262.7
Co	9.8
Ir	3.8
Ni	8.3
Pd	3.9
Pt	0.02
Cu	0.06
Ag	0.0
Au	0.1
Zn	18.8
Cd	0.07
Hg	7.2
Al	192.2

Table 4.2 indicates the high concentration of heavy metals present in the AMD. These heavy metals have a high solubility at a low pH (Taylor et al., 2005) and acidity is generated through the formation of metal hydroxides. Precipitation of heavy metals to form hydroxides and sulfides are dependent on the metal and can be determined according to their corresponding Pourbaix diagrams. Synthetic AMD contains a lower mineral acidity than AMD as it is made up of fewer heavy metals.

Due to the acidic nature of the AMD leachate, it is expected that the disintegration of the grass would be a physiochemical process. The driving force of the breakdown of the grass exposed to the AMD leachate would be largely due to acid hydrolysis. It was expected that there would be a greater production of COD using the Synthetic AMD solution,

however this was not the case as shown in Figure 4.9. During acid hydrolysis, H^+ ion attacks the surface of the grass, thereafter the surface of the grass was exposed to the bacteria and enzymatic hydrolysis occurs (Esteghlalian et al., 1997).

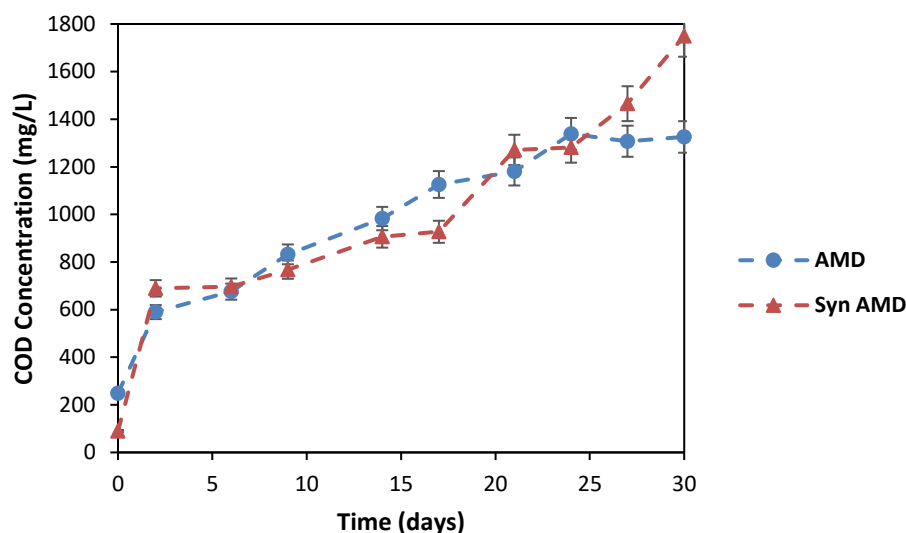


Figure 4.9 Averaged concentration of COD produced for AMD and Synthetic AMD

Over the course of the experiment, the reactor containing an AMD leachate produced an average concentration of soluble COD of 960.64 mg COD/L ($p=0.0005$). The reactor containing a Synthetic AMD leachate produced an average concentration of 984.39 mg COD/L ($p=0.03$). The deviation from the expected behaviour can be explained by analysing the acidic behaviour of the AMD. At an acidic pH, microbial activity is inhibited (Batstone et al., 2002). As there is a slower rate of growth of bacteria (Batstone & Keller, 2003) in the reactor containing an AMD leachate, there is a similar concentration of COD being produced as less soluble organic substances is being consumed by the bacteria. This means that that the soluble COD in the reactor containing the leachate with the synthetic AMD is being consumed by the bacteria present in the system. Figure 4.10 illustrates the sugar concentration produced by the two reactors. The reactor containing an AMD leachate produces an average of 4 times the amount of sugars than the reactor containing the synthetic AMD. As the reactor with the synthetic AMD is operating at a higher pH, the sugars are constantly being utilised by the bacteria in the system.

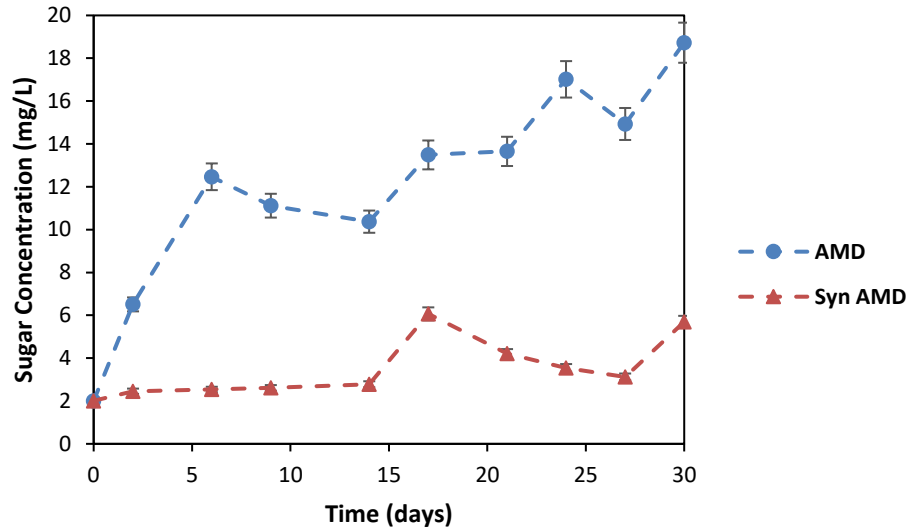


Figure 4.10 Averaged sugar concentration for AMD and Synthetic AMD

This theory can be re-enforced by looking at the sulfate and the dissolved sulfate concentration in the two reactors. Figure 4.11 shows the concentration of sulfate for the duration of 30 days and Figure 4.12 shows the percent average of sulfate removed overall.

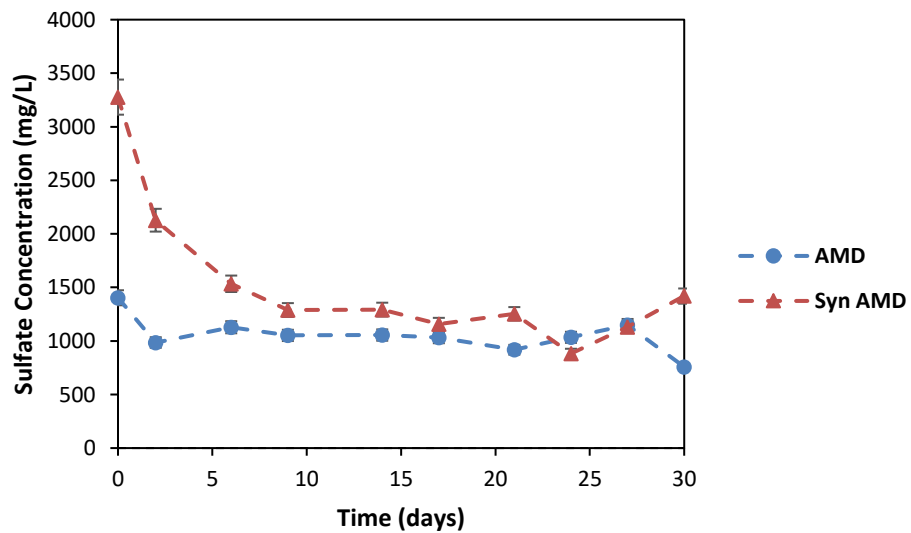


Figure 4.11 Averaged concentration of sulfate for AMD and Synthetic AMD

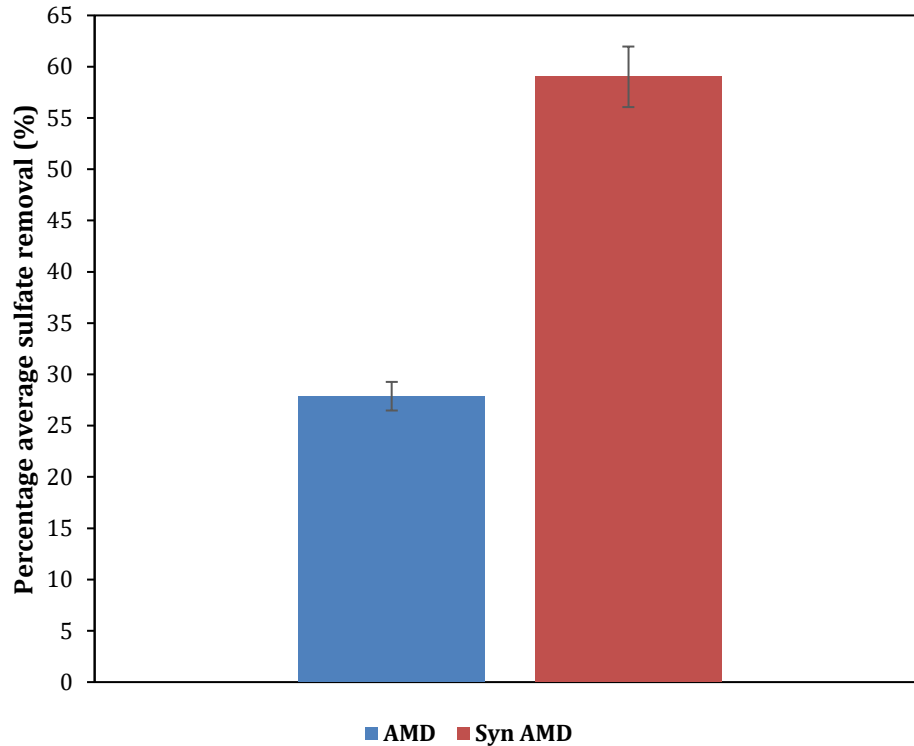


Figure 4.12 Percentage of average sulfate removed

A greater concentration of sulfate was removed from the reactor containing the synthetic AMD leachate. The reactor containing a synthetic AMD leachate had an average sulfate removal efficiency of 59% while reactor containing an AMD leachate had an average sulfate removal efficiency of 28%. This is due to the fact that the SRB were inhibited by the low pH and metal toxicity in the system containing an AMD leachate. This idea can be re-enforced by examining the amount of dissolved sulfide in the system shown in Figure 4.13 and Figure 4.14. As more sulfate is available in the reactor containing a synthetic AMD leachate it is expected that more sulfide will be produced during dissimilatory sulfate reduction. The overall average percentage of sulfide produced for the reactor containing an AMD leachate is lower in comparison and aligns with the lower averaged sulfate removed. The acidic nature of the AMD inhibits the microbial activity in the reactor. There was a high removal efficiency of sulfate, however the sulfate concentration still remained between 600 – 1000 mg/L. The effluent of the LBR would require further treatment to reach the target water quality of 0 – 200 mg/L identified by (Department of Water Affairs and Forestry., 1996).

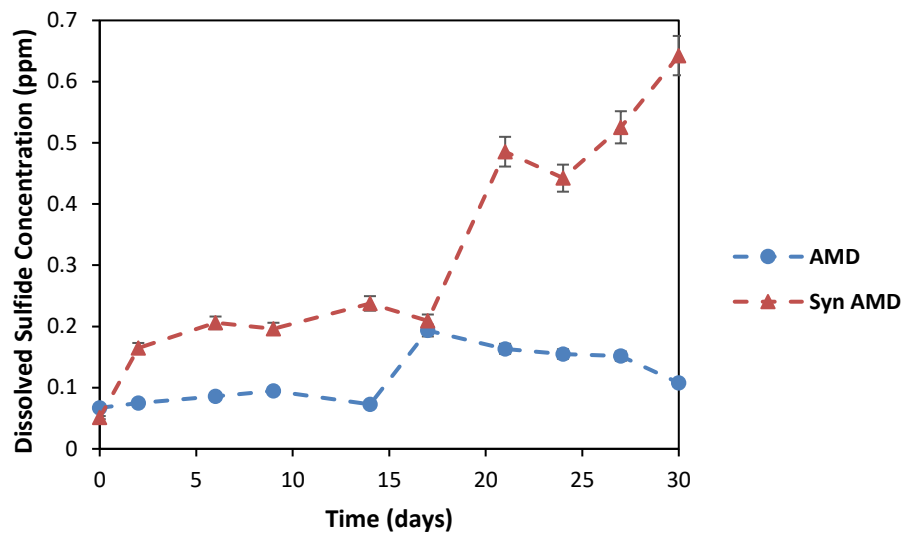


Figure 4.13 Averaged dissolved sulfide concentration for AMD and Synthetic AMD

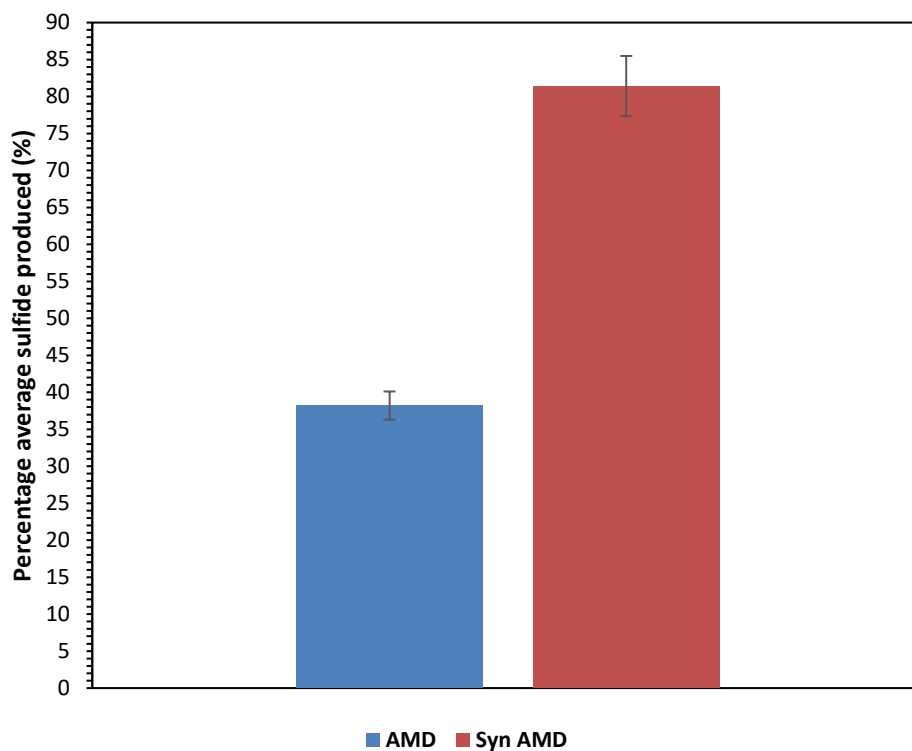


Figure 4.14 Percentage average dissolved sulfide produced

Figure 4.11 illustrates that after day 20, the average sulfate concentration in the system increased and Figure 4.13 shows that the after day 20 the average dissolved sulfide was

at its peak. As the concentration of dissolved sulfide increases in the system there will be a decrease in the bacterial activity due to the inhibitive properties of sulfide.

The concentration of dissolved iron and total iron was measured throughout the course of 30 days. A low pH range increases the solubility of heavy metals including iron. The pH range of the AMD leachate remained relatively low throughout the experiment and hence the iron present in the system remained soluble as shown in Figure 4.15.

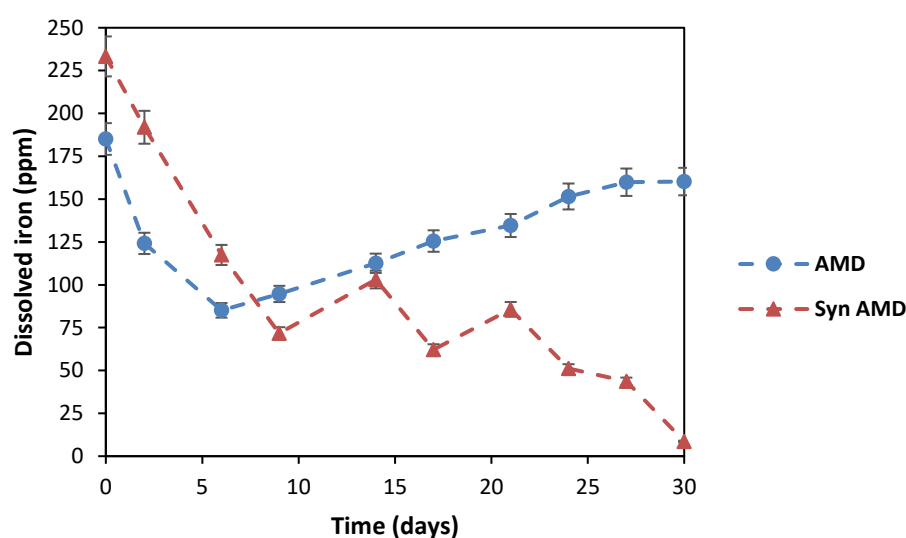


Figure 4.15 Averaged dissolved iron concentration for AMD and Synthetic AMD

Due to the acidity present in the AMD leachate, the sulfate reducing bacteria are inhibited and there is a low production of sulfides. Dissolved sulfides occur at a pH range of 6.5 – 8 (Rinzema & Lettinga, 1988). At a neutral pH range iron reacts with the dissolved sulfide in the system to form a FeS precipitate. Figure 4.15 illustrates the decrease of the average dissolved iron in the system containing synthetic AMD. The average dissolved iron removal efficiency of the process containing a synthetic AMD and an AMD leachate was 31% and 65% respectively. The neutral pH of the synthetic AMD gave rise to a higher transformation of iron to iron (II) sulfide thereby increasing the average removal of iron as shown in Figure 4.16

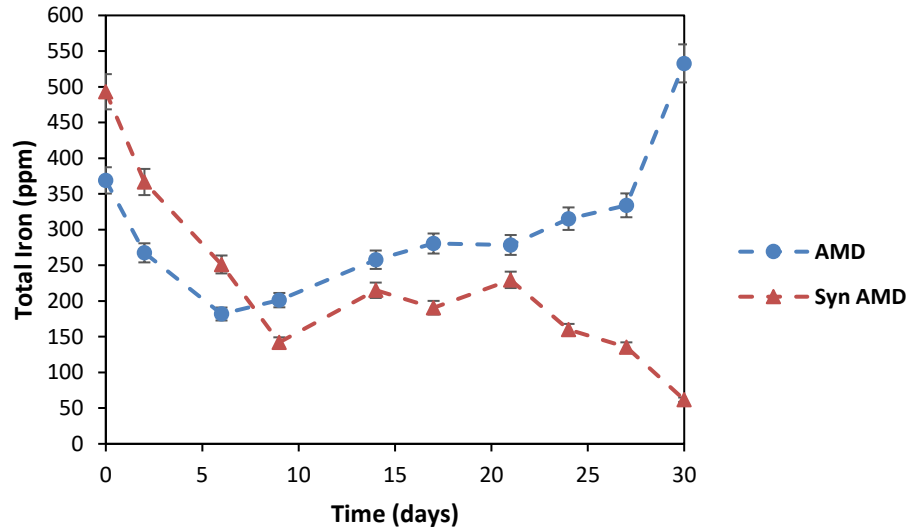


Figure 4.16 Averaged total iron concentration for AMD and Synthetic AMD

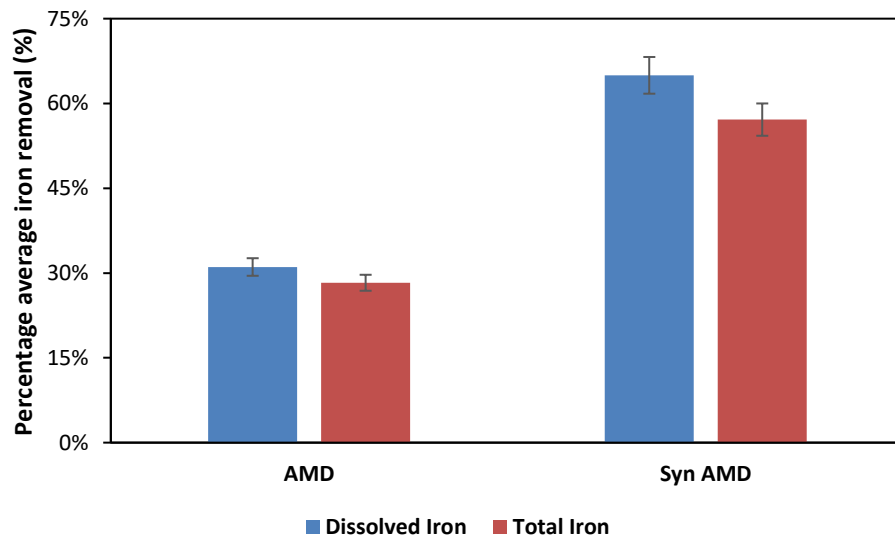


Figure 4.17 Percentage of average iron removed

During the experiment a black coating was formed on the sides of the reactor and on the grass as a result of the produced iron (II) sulfide. It can be concluded that the black coating was iron (II) sulfide. The black coating appeared on the reactor containing synthetic AMD. As the only heavy metal that synthetic AMD contains is iron it is reasonable to conclude that the black coating is iron (II) sulphide. There was a larger production of iron (II) sulfide in the system using a synthetic AMD and this is evident in Figure 4.18. The reactor on the left contained an AMD leachate while the reactor on the right contained a synthetic AMD leachate. The reactor on the right had a thick black coating on the grass,

on the side of the reactor and in the pipes. The production of iron (II) sulfide gave rise to clogging issues in the pipes and the sieve plate during the experiment. The concentration of dissolved iron of the treated effluent is not in the water quality range set out by Department of Water Affairs and Forestry (1996) and further treatment is recommended.

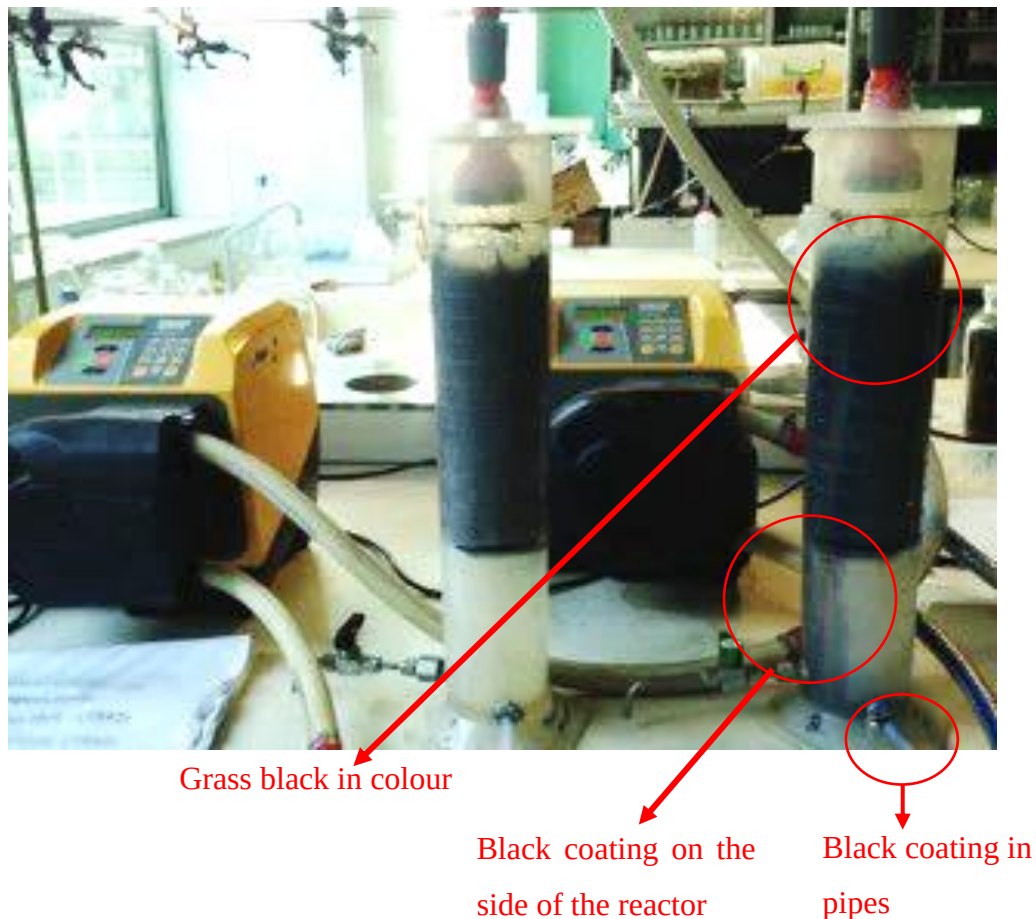


Figure 4.18 A visual comparison of the production of iron (II) sulfide between the reactor containing and AMD leachate (left) and the reactor containing synthetic AMD (right)

The bioremediation of synthetic AMD is possible utilising an LBR and indigenous South African grasses. However, the effluent from the reactor did not meet the water quality standards set out by Department of Water Affairs and Forestry (1996). Further treatment of the water is required to meet the drinking standards set out by DWAF. It is recommended that the LBR should be coupled with an upflow anaerobic sludge bed (UASB) as described by Thamsirioj et al. (2012). The two-phase digestion system of a leach bed coupled with an UASB promotes the destruction of the volatile solids and the production of COD in the first phase. The second phase of the system (the UASB) receives liquid rich in COD and converts the COD into biogas. Multiple leach beds could

be used and fed sequentially to simulate a continuous process (Greenway et al., 2017). However, the use of an UASB will mean that the system will have higher conversion of sulfates, but will no longer be as passive.

The process was not successful in the treatment of AMD. Due to the complexity of AMD the pH remained low and inhibited biochemical processes taking place. The AMD influent would need to be dosed with an alkaline chemical to raise the pH. However, this would mean a less semi-passive process would be achieved.

There are implications when using synthetic AMD for experimental work as the behaviour and chemistry of AMD differs greatly from synthetic AMD. AMD's chemistry varies greatly from location to location and this should be considered when selecting treatment methods for the remediation of AMD (Moodley et al., 2018).

4.2 Model Results

4.2.1 Indigenous South African grass characterisation

The AMD1 model requires an input of the composition of the grass used in the study. Table 4.3 summaries the composition of the grass which was determined by Nvirotek Laboratories, South Africa. The data represented in Table 4.3 was used to determine the biodegradable fractions, carbohydrates (X_{ch}), protein (X_{pr}), lignin (X_{li}) and inerts (X_I), of indigenous South African grass using the methods described in section 2.6.2.

Table 4.3 Composition of indigenous South African grass determined by the extended Weender analysis

Parameter	Unit	Indigenous SA grass
Total solids (TS)	%	93.1
Volatile solids (VS)	%[TS]	88.1
Raw Protein (RP)	%[TS]	5.3
Raw fibre (RF)	%[TS]	35.2
Raw lipids (RL)	%[TS]	1.6
N-free Extracts (NfE)*	%[TS]	46.0
Neutral Detergent Fibre (NDF)	%[TS]	60.7
Acid Detergent Fibre (ADF)	%[TS]	38.8

Acid Detergent Lignin (ADL)	%[TS]	15.8
-----------------------------	-------	------

*Calculated

Originally, in the ADM1 the breakdown of the composite material (X_c) was modelled using the disintegration rate and the stoichiometric f-factors of the organic fraction (f_{pr_Xc} , f_{li_Xc} , f_{ch_Xc} , f_{I_Xc}). Flores-Alsina et al. (2016) recommended that the composite material (X_c) be removed from the model and the composition of the particulates (X_{ch} , X_{pr} , X_{li} , X_I) be directly mapped into the model. Table 4.4 summarizes all calculated parameters of Indigenous South African grass the stoichiometric f-factors used in the upgraded ADM1 model.

Table 4.4 Summary of parameters used to describe the characterisation of Indigenous South African grass

Parameter	Description	Indigenous SA grass	Unit
m_{grass}	Mass of grass	0.15	kg
ρ_{grass}	Density of grass	107.14	kg/m ³
f_{pr_Xc}	Fraction of protein from X_c	0.06	-
f_{li_Xc}	Fraction of lipids from X_c	0.02	-
f_{ch_Xc}	Fraction of carbohydrates from X_c	0.52	-
f_{I_Xc}	Fraction of inerts from X_c	0.40	-
X_c	Composite material	11.39	kgCOD/m ³
X_{ch}	Carbohydrates	5.94	kgCOD/m ³
X_{pr}	Protein	0.69	kgCOD/m ³
X_{li}	Lipids	0.21	kgCOD/m ³
X_I	Inerts	4.56	kgCOD/m ³

4.3 Model Testing

The upgraded ADM1 model was simulated in MATLAB® and Simulink® as shown in Appendix C. In order to perform a cross validation, the simulation results from the AMD model were compared and contrasted to the experimental results using both the AMD leachate and the synthetic AMD leachate. The aim was to derive a model that could predict the anaerobic digestion of grass and the bioremediation of AMD. Since the composition of AMD differs from site to site, the aim was to produce a model that could reasonably predict the bioremediation of AMD using different initial starting concentrations. This research also investigated if the model could predict the behaviour of the treatment of AMD using synthetic AMD and AMD and whether this behaviour deviates from the model. The deviation between the experimental results and the model predictions was determined using the mean absolute percent error (MAPE). In Table 4.5 the initial starting concentrations of AMD and synthetic AMD are given. Total soluble COD, pH, sulfate, dissolved sulfide, sugars and dissolved iron concentrations were analysed in order to evaluate the performance of the model. The initial starting concentrations were inputted into the upgraded AMD.

Table 4.5 Initial starting conditions of AMD and Synthetic AMD

	AMD	Synthetic AMD
pH	2.1	3.2
Total soluble COD concentration (mg COD/L)	248.8	89.7
Sulfate concentration (g/L)	1.4	3.3
Dissolved sulfide concentration (ppb)	67.0	51.0
Dissolved iron concentration (ppm)	185.0	233.3
Sugar concentration (mg/L)	2.0	2.0
Mass of grass (g)	150	150

The modelled and experimental concentrations of the total dissolved COD were in agreement for both the synthetic and real AMD as shown in Figure 4.19 and Figure 4.20. Both the model and the experiment showed a positive increase in the production of dissolved COD over the course of 30 days. The MAPE used to analyse the deviation between the change in soluble COD concentration predicted by the model and

demonstrated by the experiment. Both the AMD (MAPE=12.1%) and synthetic AMD (MAPE=6.9%) correlated well to model. There is a high quantitative prediction accuracy when the calculated MAPE is below 10% (Batstone & Keller, 2003; Barrera et al., 2013) and a medium quantitative prediction accuracy is considered to be between 10% - 30% (Batstone & Keller, 2003; Barrera et al., 2013).

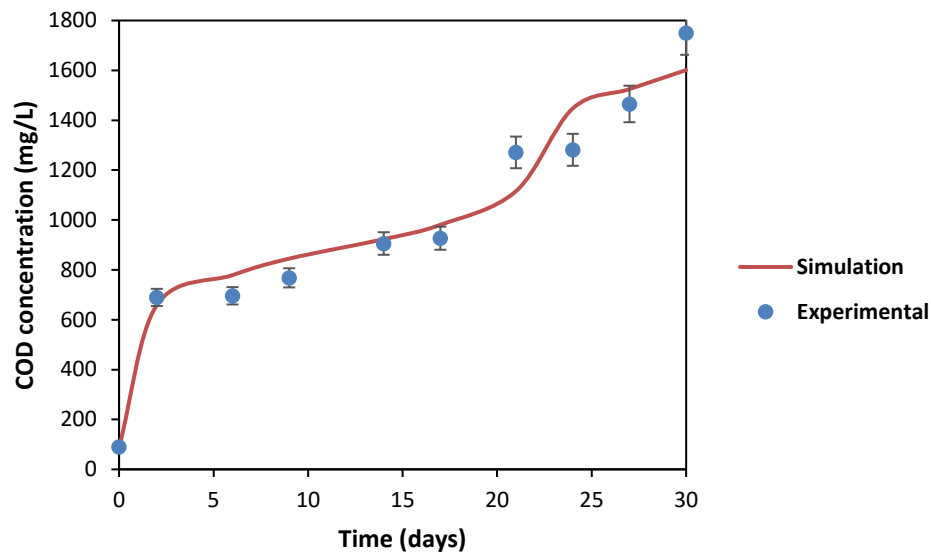


Figure 4.19 Soluble COD synthetic AMD experimental concentration vs simulated concentration

The soluble COD concentration produced by the AMD leachate deviates from the predicted concentration from day 15 onwards as shown in Figure 4.20. The upgraded ADM1 model assumes that soluble COD is being produced by the breakdown of the composite material (the indigenous South African grass) and is being consumed by the bacteria in the system.

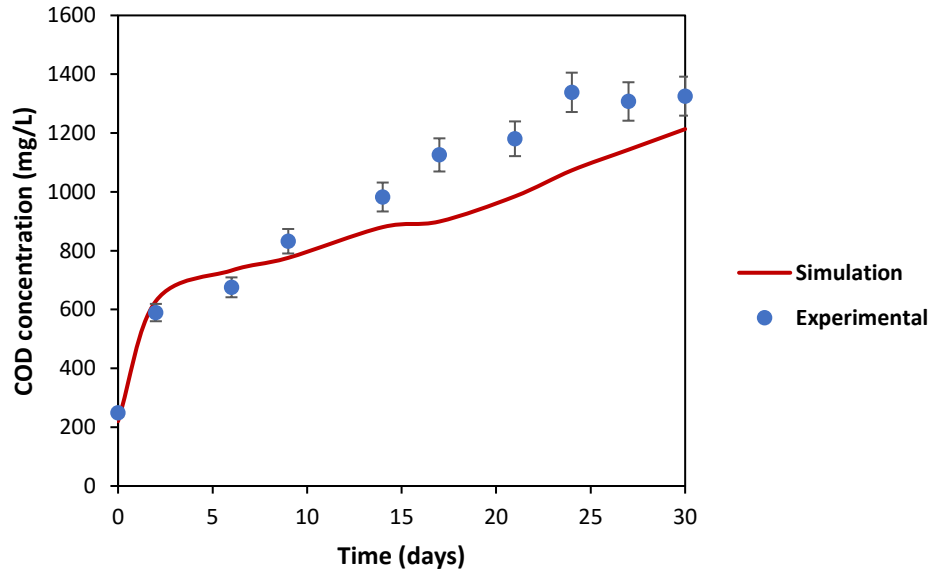


Figure 4.20 Soluble COD AMD experimental concentration vs simulated concentration

The deviation in the soluble COD concentrations, using the AMD leachate and that of the upgraded ADM1 model is explained upon the analysis of the predicted and actual sugar concentrations. The model shows a correlation to the experimental sugar concentration with synthetic AMD up to day 15 (Figure 4.21). The MAPE of the experimental sugar concentration to the predicted model is 16.9% using the synthetic AMD leachate. At day 17 and 30 the experimental concentration deviates by 89% and 77% respectively. The standard deviation of the experimental data is ± 1.36 mg/L. The simulated results do not account for the observed increase of sugar concentrations at day 17 and 30 and it is difficult to explain why these spikes occurred without additional work in future.

The upgraded model was not able to predict the sugar concentration, produced (MAPE=47%) by the breakdown of the grass and consumed by the microbes, using the AMD leachate (Figure 4.22). The experimental results analysed in Section 4.2, shows that the pH using the AMD leachate does not exceed 4 and the system remains acidic. At low pH values sugars and soluble COD were being produced through acid hydrolysis. The calculated linear regression (R^2) of the experimental data was 0.80. However, at these pH ranges bacterial activity is inhibited and there is very little to no consumption of COD and sugars.

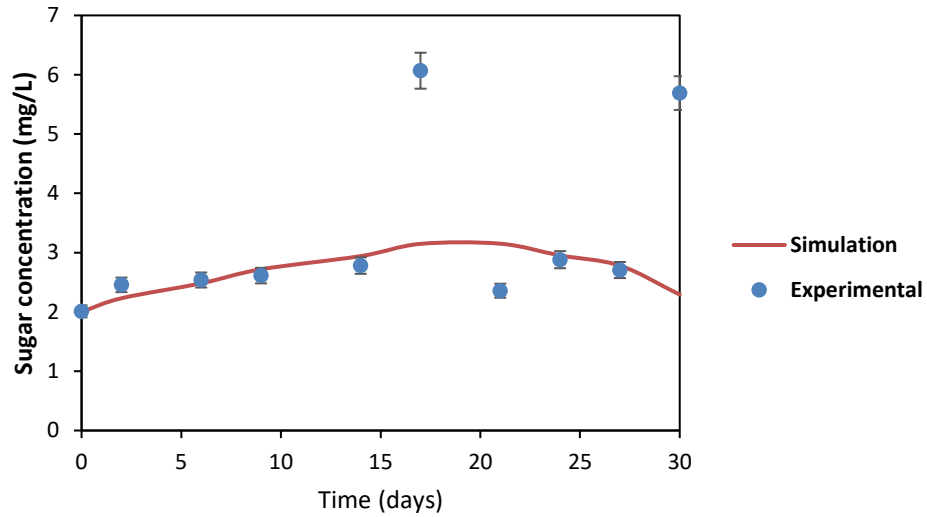


Figure 4.21 Sugar experimental concentration produced by the synthetic AMD vs simulated concentration

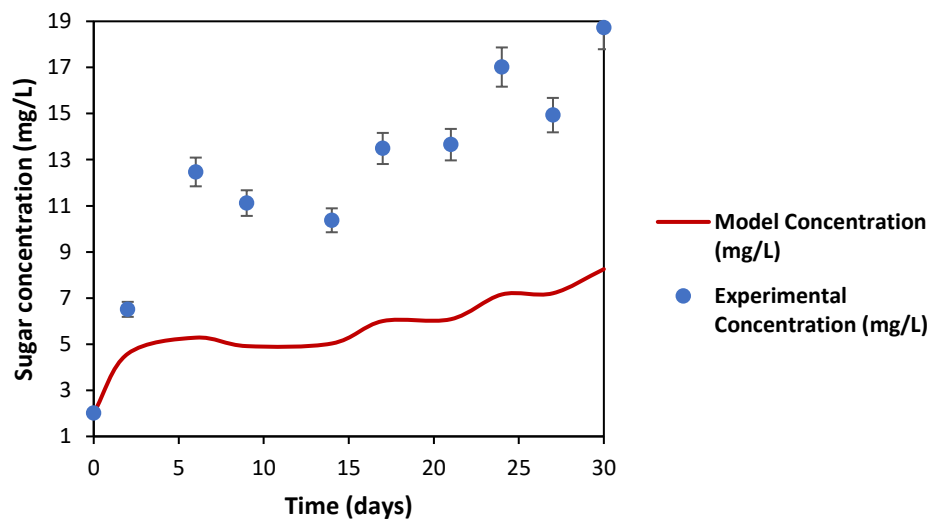


Figure 4.22 Sugar experimental concentration produced by the AMD vs simulated concentration

The upgrade of the ADM1 model had a high accuracy (MAPE=3.24%) in predicting the change in pH using the synthetic AMD leachate (Figure 4.23). The ADM1 model used an ODE15s numerical solver to determine pH shown in Appendix C.

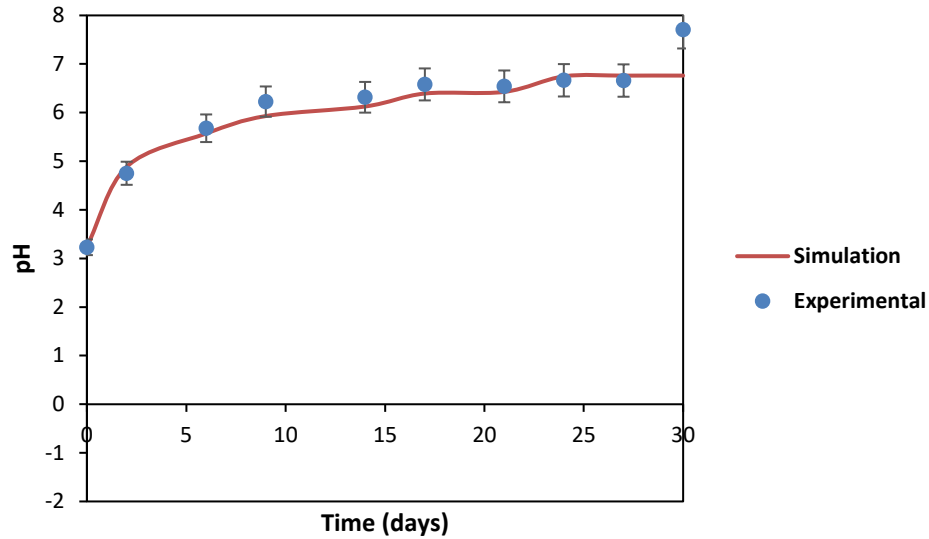


Figure 4.23 Simulated pH vs synthetic AMD pH

The model had a medium accuracy (MAPE=20%) in predicting the behaviour of the pH using AMD as shown in Figure 4.24. The predicted pH was greater than the actual pH. This can account for the variations in the predicted and actual sugar concentration. As the predicted pH is higher a smaller inhibition function was determined, thereby the model predicted a greater consumption of sugars than in reality.

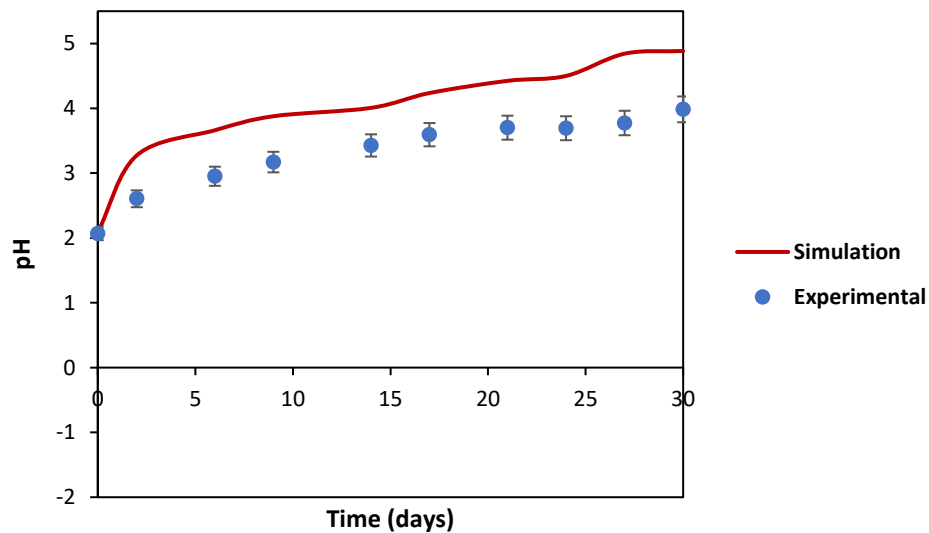


Figure 4.24 Simulated pH vs AMD pH

The upgraded ADM1 model predicted the removal of sulfate, using the synthetic AMD leachate, with a MAPE of 11%. The model identified a decrease in the sulfate concentration as shown in Figure 4.25. The model was able to accurately predict the reduction of sulfate by the SRB and the competition for organic compounds by the SRB and methanogens.

The MAPE for the sulfide experimental concentration and the predicted model was 24%. This MAPE is still considered to be qualitative prediction to the response to the system (Batstone & Keller, 2003). Deviations from the model, as shown in Figure 4.27, arose as the temperature of the LBR was not regulated and the experiment was conducted in the peak of summer. At higher temperatures there is an increase in the solubility of sulfides. The model did not take into account the fluctuations of temperatures but assumed a constant temperature of 25 °C.

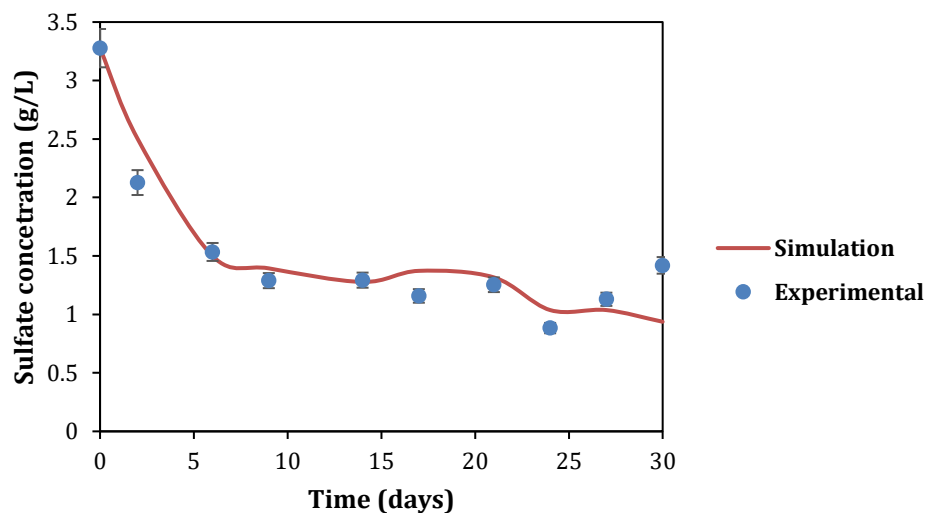


Figure 4.25 Sulfate experimental concentration produced by the synthetic AMD vs simulated concentration

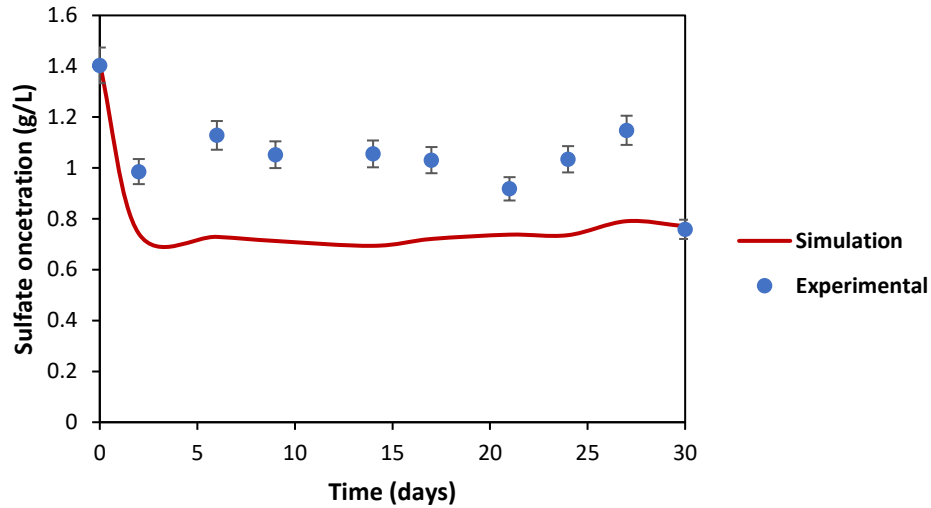


Figure 4.26 Sulfate experimental concentration produced by the AMD vs simulated concentration

The model qualitatively predicted the sulfate removal efficiency for the AMD leachate with a medium accuracy (MAPE=23.8%). In reality the low pH of the AMD inhibits the sulfate reducing bacteria. The model does not have a high accuracy in predicting the pH of the AMD system and therefore does not determine the correct inhibition factor that the pH has on the SRB. This is evident again in Figure 4.28 as the deviation of the sulfides formed using an AMD leachate compared to the predicted dissolved sulfide concentration is 23.5%. Sulfide in the system was present in the form of dissolved sulfide (liquid), hydrogen sulfide (gas) and metal sulfides (solid precipitate). Only dissolved sulfide was measured during the experiments, but the model accounted for all the sulfides in system (all three forms of sulfides).

The dissolved iron concentration had a MAPE=42.9% for the synthetic AMD leachate. There was no correlation between the dissolved iron in the system and the predicted concentration as shown in Figure 4.29. The batch nature of the LBR resulted in an oscillation in the concentration of dissolved iron in the system which caused the experimental results to deviate from the predicted behaviour of the upgraded ADM1. The predicted outlet dissolved concentration of model was greater than that of the system.

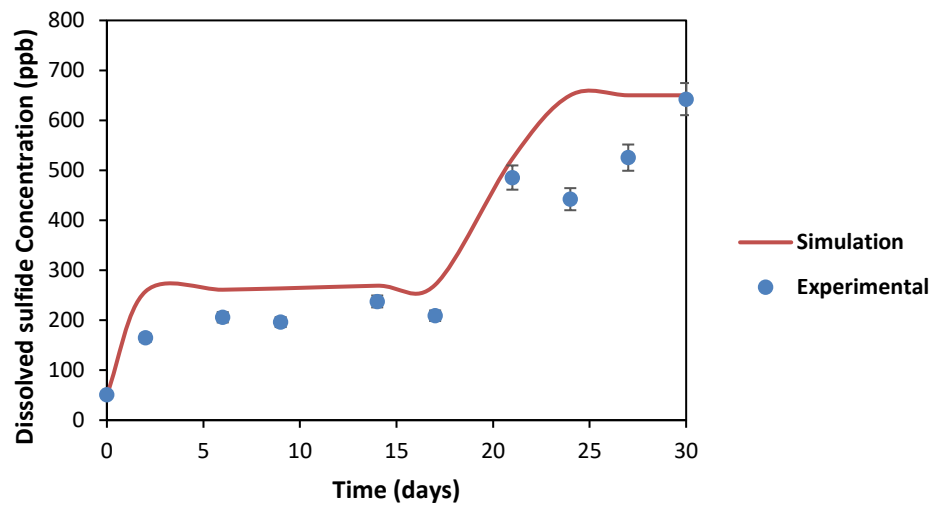


Figure 4.27 Sulfide experimental concentration produced by the synthetic AMD vs simulated concentration

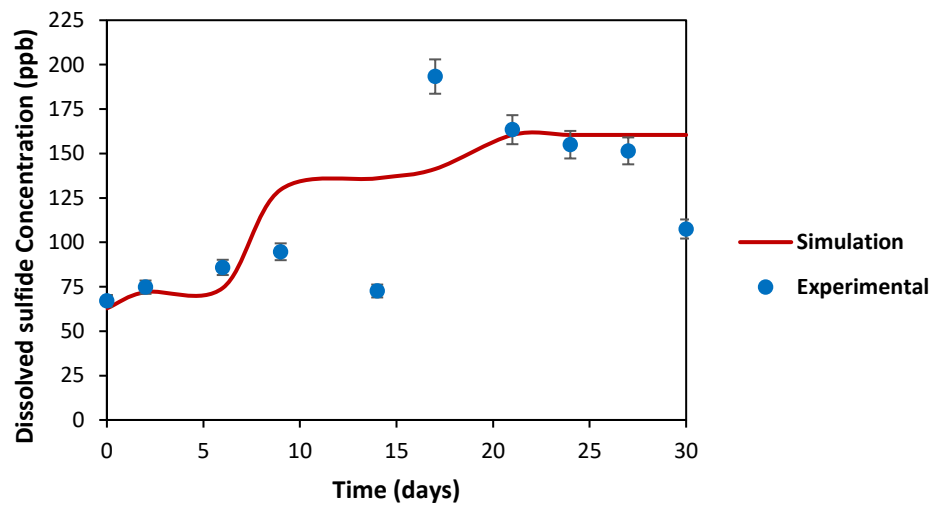


Figure 4.28 Sulfide experimental concentration produced by the AMD vs simulated concentration

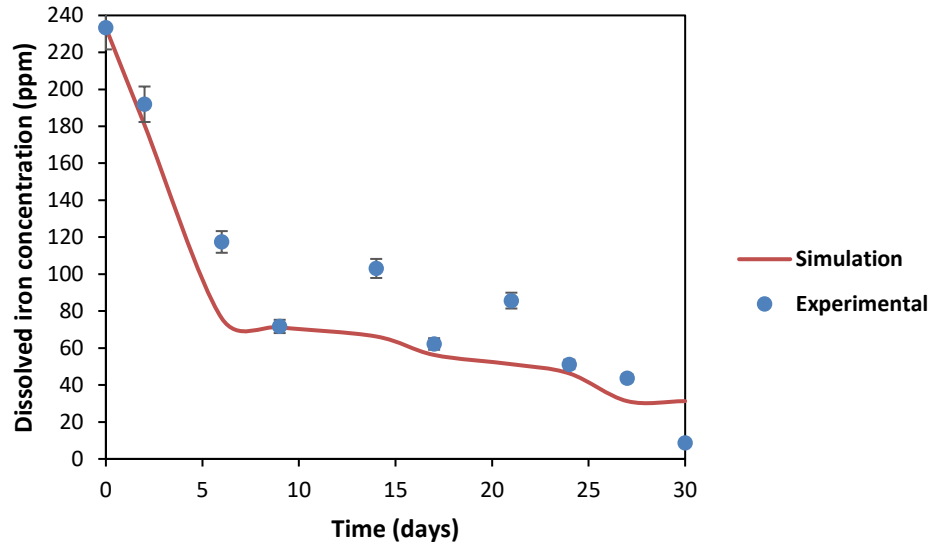


Figure 4.29 Dissolved iron experimental concentration produced by the synthetic AMD vs simulated concentration

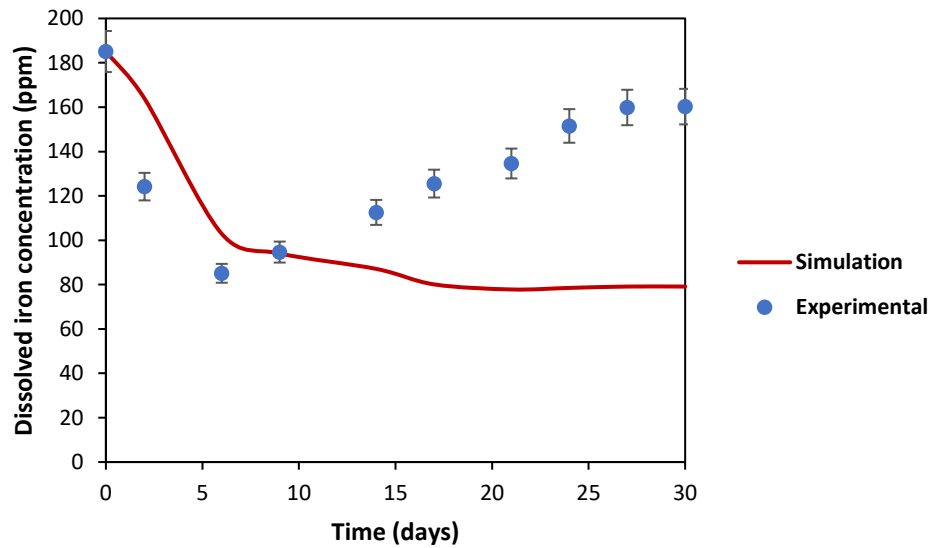


Figure 4.30 Dissolved iron experimental concentration produced by the AMD vs simulated concentration

The model showed a decrease in the dissolved iron concentration for the AMD leachate as depicted in Figure 4.30. The model could not predict the behaviour of the dissolved iron concentration. The pH of the AMD leachate remained lower than that of the predicted pH. This led to a higher solubility of iron in the leachate which was not predicted by the model. The AMD leachate is complex in its chemistry and is comprised of numerous heavy metals which play a role in the pH of the solution and the solubility

of iron. Appendix B demonstrates the composition of the AMD, through an ICP analysis, and further investigation needs to be made to the weak-acid base chemistry of the treatment of AMD in anaerobic digestion and the transformation of all heavy metals present in AMD.

The upgraded AMD1 could predict the treatment of synthetic AMD but it was unable to predict the behaviour of iron in the treatment of synthetic AMD. The chemical composition of AMD is complex and comprised of many heavy metals, as shown in Appendix B, which play a role in the overall pH of the system. If the pH of the system is not predicted by the model, the model will be unable to determine the pH inhibition, and this results in a failure to predict the biochemical process associated with anaerobic digestion including sulphate dissimilatory reduction.

5 CONCLUSION AND RECOMMENDATIONS

Characteristics of acid mine drainage (AMD) includes low pH, high concentrations of heavy metals and a high sulfate concentration. It imposes risks to the economy, human health and environment, and necessitates the treatment thereof. Active treatment is costly and requires maintenance and passive treatment promises to be a viable alternative solution to the treatment of AMD. Ramla & Sheridan (2015) demonstrated that AMD can be treated passively, using indigenous South African grass as an organic carbon source for the process of dissimilatory sulfate reduction.

This study reviewed bioreactor types and configurations that would be most suitable for the incorporation of indigenous South African grass as carbon source. One-stage dry batch digesters have advantages of being cost-effective, requiring lower maintenance and do not need dilution and maceration pre-treatment of the grass. Due to the solid nature of indigenous South African grass and the aim to achieve a semi-passive process, a one-stage leach bed reactor (LBR) was selected for the bioremediation of AMD.

A laboratory-scale LBR was constructed in order to test the grass' treatment potential of AMD. Experiments were conducted on both an AMD leachate and a synthetic AMD leachate. The leachate was trickled through the LBR and recirculated for a period of 30 days. The research investigated the optimal solid/liquid ratios using a synthetic AMD leachate. It found that higher solid/liquid ratios proved to be effective in treating AMD and synthetic AMD. At lower solid/liquid ratios the pH of the system did not increase. The optimal solid/liquid ratio was 6. At a solid/liquid ratio of 6, 6 kg of grass would be required to treat 1000 kL of AMD. Additional experimental work have to be conducted the determine the residence time of the grass to assess the viability of the process.

Through an X-ray fluorescence (XRF) analysis of the grass it was determined that the instantaneous increase in the pH of the synthetic AMD leachate was due to a cation exchange of potassium (present on the grass) and hydrogen ions. Further investigation is recommended into this phenomenon in order to explain the cause of a rapid increase in pH, as a neutral pH is optimal for anaerobic digestion (AD). Once this research has been conducted the process can be further optimized. Scanning electron microscopy (SEM) showed that there was a greater breakdown of the structure of the grass at neutral pH. A

greater destruction of the grass led to a higher surface area exposed to microbes. It can be concluded that the breakdown of grass takes place through acid hydrolysis at a low pH range and enzymatic hydrolysis occurred at a high pH range. Further research has been conducted by Burman et al. (2018) into the acid hydrolysis of indigenous South African grass at low temperatures using sulfuric acid.

The Anaerobic Digestion Model No.1 (ADM1) was selected as a basis to model the bioremediation of AMD, as the model is able to predict complex physiochemical and biochemical reactions in anaerobic digestion. Indigenous South African grass was characterised using the Weender analysis in order to determine the composition of carbohydrates, proteins, lipids and inerts on the grass. The model was then upgraded to include dissimilatory sulfate reduction and the inhibition of sulfide on biochemical processes. Lastly, the model was adapted to include iron transformation and the formation of iron (II) sulfide. Batch design equations were used to describe the leach bed reactor (LBR) and the model was simulated in MATLAB® and Simulink®.

The simulation results were compared to the experimental results of the process using both the synthetic AMD leachate and AMD in order to validate the model. The model was able to provide a high ($\text{MAPE} = \pm 10\%$) accuracy in predicting the pH and soluble COD of the process containing the synthetic AMD leachate. There was a medium accuracy ($10\% < \text{MAPE} < 30\%$) in the model's prediction of sugar, sulfate and the dissolved sulfide concentrations. However, due to the oscillating nature of the batch reactor there were some discrepancies in the prediction of the dissolved iron concentration. The model was able to qualitatively ($10\% < \text{MAPE} < 30\%$) predict the behaviour of the bioremediation of AMD. AMD has complex chemistry and comprises of numerous heavy metals which are not accounted for in the model. A further upgrade of the model is required to account for the various heavy metal species present in the system, their transformations and the weak acid-base chemistry.

Indigenous South African grass using an LBR did not prove to have a high potential for the bioremediation of AMD. After 30 days, the process did not prove to have discharge concentrations of iron and sulfate that could meet the water standards set out by the DWAF. At present, the system does not appear to be viable as a large amount of grass is required to treat a small discharge of water. The proposed process may be used for heavily

contaminated AMD where reverse osmosis cannot be used directly, and a pre-treatment step is required to reduce the sulphide concentrations first. At lower temperatures the rate of hydrolysis of the grass is inhibited. It is recommended to operate the process at higher temperatures in order to promote the breakdown of the grass by the AMD and increase the availability of the soluble COD in the system, which is utilised by the SRB. Alternatively, an investigation should be made into using sequentially fed LBRs coupled with an upflow anaerobic sludge bed (UASB) to examine if higher sulfate and iron removal efficiencies could be achieved to meet the water standards set out by DWAF.

6 REFERENCES

- Akcil, A. & Koldas, S., 2006. Acid Mine Drainage (AMD): causes, treatment and case studies. *Journal of Cleaner Production*, Volume 14, pp. 1139-1145.
- Banks, D. et al., 1997. Mine-water chemistry: the good, the bad and the ugly. *Environmental Geology*, Volume 32, pp. 157-174.
- Barnett, A., Pyle, L. & Subramanian, S. K., 1978. *Biogas Technology in the third world: a multi-disciplinary review*, Ottawa: Ottawa: International Development Research Centre.
- Barrera, E. L. et al., 2013. The sulfur chain in biogas production from sulfate-rich liquid substrates: a review on dynamic modeling with vinasse as model substrate. *Journal of Chemical Technology & Biotechnology*, Volume 88, pp. 1405-1420.
- Batstone, D. J., 2006. Mathematical Modelling of Anaerobic Reactors Treating Domestic Wastewater: Rational Criteria for Model Use. *Reviews in Environmental Science and Bio/Technology*, Volume 5, pp. 57-71.
- Batstone, D. J. & Keller, J., 2003. Industrial applications of the IWA anaerobic digestion model No. 1 (ADM1). *Water Science and Technology*, 47(12), pp. 199-206.
- Batstone, D. J. et al., 2002. The IWA Anaerobic Digestion Model No 1 (ADM 1). *Water Science & Technology*, Volume 45, pp. 65-73.
- Bhattacharyya, K. G. & Sen Gupta, S., 2006. Adsorption of chromium (VI) from water by clays. *Industrial & engineering chemistry research*, 45(21), pp. 7232-7240.
- Burgers, C. L., 2002. Synthesis and Characterisation of Sesquioxidic Precipitates Formed by the Reaction of Acid Mine Drainage with Fly Ash Lechate. *M.Sc. thesis, University of Stellenbosch, South Africa*, pp. 21-23.

Burman, N. W., Sheridan, C., van Dyk, L. & Harding, K. G., 2018. Modelling of low temperature dilute sulfuric acid pre-treatment of South African grass. *Bioresource Technology Reports*, Volume 4, pp. 21-28.

Cobbing, J. E., 2008. Institutional Linkages and Acid Mine Drainage: The Case of the Western Basin in South Africa. *International Journal of Water Resources Development* , 24(3), pp. 451 - 462.

Coetzee, H. & Winde, F., 2006. *n Assessment of Sources, Pathways, Mechanisms and Risks of Current and Potential Future Pollution of Water and Sediments in Gold-Mining Areas of the Wonderfonteinspruit Catchment: WRC Report No. 1214/1/06*, Pretoria: Water Research Commission (WRC).

Coulton , R., Bullen C & Hallet, C., 2003b. The Design and Optimisation of Active Mine Water Treatment Plants. *Land Contam Reclaim*, Volume 11, pp. 273-279.

De Baere, L. & Mattheeuws, B., 2000. Anaerobic digestion of solid waste: State-of-the-art. *Water Science & Technology*, 41(3), pp. 283-290.

Department of Water Affairs and Forestry., 1996. *South African Water Quality Guidelines (second edition), Volume 3: Industrial Use*, Pretoria, South Africa: CSIR Environmental Services.

Dieterich, B., 2008. *Energy crops for anaerobic digestion (AD) in Westray* , Orkney, UK: Heat and Power Ltd., Westray, Orkney.

Esteghlalian, A., Hashimoto, A. G., Fenske, J. J. & Penner, M. H., 1997. Modeling and optimization of the dilute-sulfuric-acid pretreatment of corn stover, poplar and switchgrass. *Bioresource Technology*, Volume 59, pp. 129-136.

Fedorovich, V., Lens, P. & Kalyuzhnyi, S., 2003. Extension of Anaerobic Digestion Model No. 1 with processes of sulfate reduction. *Applied Biochemistry and Biotechnology*, Volume 109, pp. 33-45.

Flores-Alsina, X. et al., 2015. A plant-wide aqueous phase chemistry module describing pH variations and ion speciation/pairing in wastewater treatment process models. *Water research*, Volume 85, pp. 255-265.

Flores-Alsina, X. et al., 2016. Modelling phosphorus (P), sulfur (S) and iron (Fe) interactions for dynamic simulations of anaerobic digestion processes. *Water Research*, Volume 95, pp. 370-382.

Gaikwad, R. W. & Gupta, D. V., 2008. Review on removal of heavy metals from acid mine drainage. *Applied ecology and environmental research*, Volume 6, pp. 81-98.

Gerike, P., 1984. The biodegradability testing of poorly water soluble compounds. *Chemosphere*, Volume 13, pp. 169-190.

Greben, H. A., Baloyi, J., Sigama, J. & Venter, S. N., 2009. Bioremediation of sulphate rich mine effluents using grass cuttings and. *Journal of Geochemical Exploration*, Volume 100, pp. 163-168.

Greenway, N. et al., 2017. *Process design for the treatment of acid mine drainage utilising lignocellulosic material as an organic carbon source for the process of dissimilatory sulfate reduction*. Mauritius, International Conference on Energy, Environment and Climate - ICEECC 2017.

Inglezakis, V. J., Loizidou, M. D. & Grigoropoulou, H. P., 2003. Ion exchange of Pb^{2+} , Cu^{2+} , Fe^{3+} , and Cr^{3+} on natural clinoptilolite: selectivity determination and influence of acidity on metal uptake. *Journal of colloid and Interface Science*, 261(1), pp. 49-54.

IS 3025: Part 24, 2003. *Methods of Sampling and Test (Physical and Chemical) for Water and Wastewater: Part 24 Sulfates*, s.l.: s.n.

Jennings, S. R., Neuman, D. R. & Blicher, P. S., 2008. *Acid Mine Drainage and Effects on Fish Health and Ecology: A Review*, Bozeman, MT: Reclamation Research Group Publication.

Johnson, D. B. & Hallberg, K. B., 2005. Acid mine drainage remediation options: a review. *Science of The Total Environment*, Volume 338, pp. 3-14.

Kadlech, R. H. et al., 2000. Constructed Wetlands for Pollution Control: Processes, Performance, Design and Operation. In: *IWA Scientific and Technical Report No. 8*. London: IWA Publishing.

Kaksonen, A. H. & Puhakka, J. A., 2007. Sulfate Reduction Based Bioprocesses for the Treatment of Acid Mine Drainage and the Recovery of Metals. *Engineering Life Science*, 7(6), pp. 541-564.

Koch, K. et al., 2010. Biogas from grass silage – Measurements and modeling with ADM1. *Bioresource Technology*, Volume 101, pp. 8158-8165.

Kootstra, A. M., Beeftink, H. H., Scott, E. L. & Sanders, J. P., 2009. Comparison of dilute mineral and organic acid pretreatment for enzymatic hydrolysis of wheat straw. *Biochemical Engineering Journal*, Volume 46, pp. 126-131.

Liefferink, M., 2009. *Comments on the environmental impact assessment (EIA) for the proposed re-mining and processing of tailings facilities at the operations of Harmony Gold Mining Company Limited in the Welkom area*, s.l.: The Federation for a Sustainable Environment.

Lindsay, M. B., Blowes, D. W., Condon, P. D. & Ptacek, C. J., 2011. Organic carbon amendments for passive in situ treatment of mine drainage: Field experiments. *Applied Geochemistry*, Volume 26, pp. 1169-1183.

Lübken, M., Blowes, D. W., Condon, P. D. & Ptacek, C. J., 2011. Organic carbon amendments for passive in situ treatment of mine drainage: Field experiments. *Applied Geochemistry*, Volume 26, pp. 1169-1183.

Madzivire, G., 2015. *Removal of sulphates from South African mine water using coal fly ash*, Cape Town: Doctoral dissertation, University of Cape Town.

Magowo, E., Rumbold, K. & Sheridan, C., 2015. *The Utilization of Cellulosic Biomass in treating AMD and the Subsequent Generation of*, Johannesburg: Doctoral dissertation, University of the Witwatersrand.

Mamais, D. et al., 1994. Determination of ferric chloride dose to control struvite precipitation in anaerobic sludge digesters. *Water Environment Research*, 66(7), pp. 912-918.

Mambo, P. M., 2011. *Towards a Sustainable Bioprocess for the Remediation of Acid Mine Drainage*, Grahamstown: Doctoral dissertation, Rhodes University.

Marchaim, U., 1992. *Biogas processes for sustainable development*. s.l.:FFood & Agriculture Org.

Mbamba, C. K., Tait, S., Flores-Alsina, X. & Batstone, D. J., 2015. A systematic study of multiple minerals precipitation modelling in wastewater treatment. *Water research*, Volume 85, pp. 359-370.

McCarthy, T. S., 2011. The impact of acid mine drainage in South Africa. *South African Journal of Science*, Volume 107, pp. 1-7.

Moodley, I., Sheridan, C., Kappelmeyer, U. & Akcil, A., 2018. Environmentally sustainable acid mine drainage remediation: Research developments with a focus on waste/by-products. *Minerals Engineering*, Volume 126, pp. 207-220.

Motsi, T., Rowson, N. A. & Simmons, M. J., 2009. Adsorption of heavy metals from acid mine drainage by natural zeolite. *International Journal of Mineral Processing*, Volume 92, pp. 42-48.

Name, T. & Sheridan, C., 2014. Remediation of acid mine drainage using metallurgical slags. *Minerals Engineering*, Volume 64, pp. 15-22.

Nielsen, A. H., Lens, P., Vollertsen, J. & Hvitved-Jacobsen, T., 2005. Sulfide-iron interactions in domestic wastewater from a gravity sewer. *Water research*, 39(12), pp. 2747-2755.

Nizami, A.-S. & Murphy, J. D., 2010. What type of digester configurations should be employed to produce biomethane from grass silage?. *Renewable and Sustainable Energy Reviews*, Volume 14, pp. 1558-1568.

Ochieng, G. M., Seanego, E. S. & Nkwonta, O. I., 2010. Impacts of mining on water resources in South Africa: A review. *Scientific Research and Essays*, Volume 5, pp. 3351-3357.

Peppas, A., Komnitsas, K. & Halikia, I., 2000. Use of organic covers for acid mine drainage control. *Minerals Engineering*, Volume 13, pp. 563-574.

Poinapen, J. & Ekama, G., 2010. Biological sulphate reduction with primary sewage sludge in an upflow anaerobic sludge bed reactor-Part 6: Development of a kinetic model for BSR. *Water SA*, Volume 36, pp. 203-214.

Potgieter-Vermaak, S. S., Potgieter, J. H., Monama, P. & Van Grieken, R., 2006. Comparison of limestone, dolomite and fly ash as pre-treatment agents for acid mine drainage. *Minerals Engineering*, Volume 19, pp. 454-462.

Ramla, B. & Sheridan, C., 2015. The potential utilisation of indigenous South African grasses for acid mine drainage remediation. *Water SA*, Volume 41, pp. 247-252.

Rinzema, A. & Lettinga, G., 1988. Anaerobic treatment of sulfate containing. *Biotreatment systems*, Volume 3, pp. 65-109.

Rosén, C. & Jeppsson, U., 2006. *Aspects on ADM1 Implementation within the BSM2 Framework*, Lund, Sweden: Department of Industrial Electrical Engineering and Automation, Lund University.

Sheridan, C., Harding, K., Koller, E. & De Pretto, A., 2013. A comparison of charcoal- and slag-based constructed wetlands for acid mine drainage remediation. *Water SA*, Volume 39, pp. 369-374.

Simate, G. S. & Ndlovu, S., 2014. Acid mine drainage: Challenges and opportunities. *Journal of Environmental Chemical Engineering*, Volume 2, pp. 1785-1803.

Solon, K. et al., 2015. Effects of ionic strength and ion pairing on (plant-wide) modelling of anaerobic digestion. *Water Research*, Volume 70, pp. 235-245.

Speece, R. E., 1983. Anaerobic biotechnology for industrial wastewater treatment. *Environmental science & technology*, 17(9), pp. 416A-427A.

Sun, Y. & Cheng, J., 2002. Hydrolysis of lignocellulosic materials for ethanol production: a review. *Bioresource Technology*, Volume 83, pp. 1-11.

Sun, Y. & Cheng, J., 2005. Dilute acid pretreatment of rye straw and bermudagrass for ethanol production. *Bioresource Technology*, Volume 96, pp. 1599-1606.

Taylor, J., Pape, S. & Murphy, N., 2005. *A Summary of Passive and Active Treatment Technologies for Acid and Metalliferous Drainage (AMD)*. Fremantle, Australia, Australian Centre for Minerals Extension and Research (ACMER) 2 Fifth Australian Workshop on Acid Drainage.

Taylor, M., 2006. *An Assessment of Iron and Steel Slag for treatment of Stormwater Pollution*, Hamilton, New Zealand: Landcare Research.

Thamsirioj, T., Nizami, A. S. & Murphy, J. D., 2012. Use of modelling to aid design of a two-phase grass digestion system. *Bioresource Technology*, Volume 110, pp. 379-389.

Tichy, R., Lens, P., Grotenhuis, J. C. & Bos, P., 1998. Solid-state reduced sulfur compounds: Environmental aspects and bio-remediation. *Crit. Rev. Environ. Sci. Technol.*, Volume 28, pp. 1-40.

Tsapekos, P. et al., 2017. Process performance and comparative metagenomic analysis during co-digestion of manure and lignocellulosic biomass for biogas production. *Applied energy*, Volume 185, pp. 126-135.

Tutu, H., McCarthy, T. S. & Cukrowska, E., 2008. The chemical characteristics of acid mine drainage with particular reference to sources, distribution and remediation: The Witwatersrand Basin, South Africa as a case study. *Applied Geochemistry*, 23(12), pp. 3666-3684.

Vandevivere, P., De Baere, L. & Verstraete, W., 2003. Types of anaerobic digester for solid wastes. In: J. Mata-Alvarez, ed. *Biomethanization of the organic fraction of municipal solid wastes*. s.l.:Iwa Publishing, p. 336.

Yu, H., Samani, Z., Hanson, A. & Smith, G., 2002. Energy recovery from grass using two-phase anaerobic digestion. *Waste management*, Volume 22, pp. 1-5.

Zhang, L. et al., 2008. Chemical and biological technologies for hydrogen sulfide emission control in sewer systems: a review. *Water research*, Volume 42, pp. 1-12.

APPENDIX A THE UPGRADED ANAEROBIC DIGESTION MODEL NO. 1 (ADM1)

Supplementary information and matrices for the extended Anaerobic Digestion Model No.1 (ADM1)

Table 6.1	Biochemical rate coefficients ($v_{i,j}$) for soluble components
Table 6.2	Biochemical rate coefficients ($v_{i,j}$) for particulate components
Table 6.3	Biochemical rate coefficients ($v_{i,j}$) for S model extension components
Table 6.4	Biochemical rate coefficients ($v_{i,j}$) for Fe model extension components
Table 6.5	Stoichiometric matrix for precipitation process
Table 6.6	Stoichiometric matrix for liquid-gas transfer process
Table 6.7	Process kinetic rate equations (ρ_j)
Table 6.8	Parameter values for the modified model

Table 6.1 Biochemical rate coefficients ($v_{i,j}$) for soluble components

Component →		i	1	2	3	4	5	6	7	8	9	10	11	12
j	Process		S_{su}	S_{aa}	S_{fa}	S_{va}	S_{bu}	S_{pro}	S_{ac}	S_{h2}	S_{ch4}	S_{ic}	S_{in}	S_i
1	Hydrolysis of Carbohydrates		1									$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,1}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,1}$	
2	Hydrolysis of Proteins			1								$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,2}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,2}$	
3	Hydrolysis of Lipids		$1 - f_{fa,li}$		$f_{fa,li}$							$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,3}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,3}$	
4	Uptake of Sugars		-1				$(1-Y_{su}) f_{bu,su}$	$(1-Y_{su}) f_{pro,su}$	$(1-Y_{su}) f_{ac,su}$	$(1-Y_{su}) f_{h2,su}$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,4}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,4}$	
5	Uptake of Amino Acids			-1		$(1-Y_{aa}) f_{va,aa}$	$(1-Y_{aa}) f_{bu,aa}$	$(1-Y_{aa}) f_{pro,aa}$	$(1-Y_{aa}) f_{ac,aa}$	$(1-Y_{aa}) f_{h2,aa}$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,5}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,5}$	
6	Uptake of LCFA				-1				$(1-Y_{fa}) 0.7$	$(1-Y_{fa}) 0.30$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,6}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,6}$	
7	Uptake of Valerate					-1		$(1-Y_{c4}) 0.54$	$(1-Y_{c4}) 0.31$	$(1-Y_{c4}) 0.15$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,7}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,7}$	
8	Uptake of Butyrate						-1		$(1-Y_{c4}) 0.80$	$(1-Y_{c4}) 0.20$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,8}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,8}$	
9	Uptake of Propionate							-1	$(1-Y_{pro}) 0.57$	$(1-Y_{pro}) 0.43$		$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,9}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,9}$	
10	Uptake of Acetate								-1		$(1-Y_{ac})$	$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,10}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,10}$	
11	Uptake of Hydrogen									-1	$(1-Y_{h2})$	$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,11}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,11}$	
12	Decay of X_{su}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,12}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,12}$	$f_{sl,xb}$
13	Decay of X_{aa}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,13}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,13}$	$f_{sl,xb}$
14	Decay of X_{fa}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,14}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,14}$	$f_{sl,xb}$
15	Decay of X_{c4}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,15}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,15}$	$f_{sl,xb}$
16	Decay of X_{pro}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,16}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,16}$	$f_{sl,xb}$
17	Decay of X_{ac}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,17}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,17}$	$f_{sl,xb}$
18	Decay of X_{h2}											$- \frac{\hat{a}}{i=t-9,12-30} C_i v_{i,18}$	$- \frac{\hat{a}}{i=t-9,12-30} N_i v_{i,18}$	$f_{sl,xb}$
			Monosaccharides (kg COD.m ⁻³)	Amino acids (kg COD.m ⁻³)	Long chain fatty acids (kg COD.m ⁻³)	Total valerate (kg COD.m ⁻³)	Total butyrate (kg COD.m ⁻³)	Total propionate (kg COD.m ⁻³)	Total acetate (kg COD.m ⁻³)	Hydrogen (kg COD.m ⁻³)	Methane (kg COD.m ⁻³)	Inorganic carbon (kmol C.m ⁻³)	Inorganic nitrogen (kmol N.m ⁻³)	Soluble inerts (kg COD.m ⁻³)

Table 6.2 Biochemical rate coefficients ($v_{i,j}$) for particulate components

Component →		i	13	14	15	16	17	18	19	20	21	22	23
j	Process ↓		X_{ch}	X_{pr}	X_{li}	X_{su}	X_{aa}	X_{fa}	X_{c4}	X_{pro}	X_{ac}	X_{h2}	X_I
1	Hydrolysis of Carbohydrates		-1										
2	Hydrolysis of Proteins			-1									
3	Hydrolysis of Lipids				-1								
4	Uptake of Sugars					Y_{su}							
5	Uptake of Amino Acids						Y_{aa}						
6	Uptake of LCFA							Y_{fa}					
7	Uptake of Valerate								Y_{c4}				
8	Uptake of Butyrate								Y_{c4}				
9	Uptake of Propionate									Y_{pro}			
10	Uptake of Acetate										Y_{ac}		
11	Uptake of Hydrogen											Y_{h2}	
12	Decay of X_{su}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$	-1							$f_{xl,xb}$
13	Decay of X_{aa}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$		-1						$f_{xl,xb}$
14	Decay of X_{fa}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$			-1					$f_{xl,xb}$
15	Decay of X_{c4}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$				-1				$f_{xl,xb}$
16	Decay of X_{pro}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$					-1			$f_{xl,xb}$
17	Decay of X_{ac}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$						-1		$f_{xl,xb}$
18	Decay of X_{h2}		$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$							-1	$f_{xl,xb}$
			Carbohydrates (kg COD.m ⁻³)	Proteins (kg COD.m ⁻³)	Lipids (kg COD.m ⁻³)	Sugar degraders (kg COD.m ⁻³)	Amino acids degraders (kg COD.m ⁻³)	LCFA degraders (kg COD.m ⁻³)	Valerate and butyrate degraders (kg COD.m ⁻³)	Propionate degraders (kg COD.m ⁻³)	Acetate degraders (kg COD.m ⁻³)	Hydrogen degraders (kg COD.m ⁻³)	Particulate inerts (kg COD.m ⁻³)

Table 6.3 Biochemical rate coefficients (vi,j) for S model extension components

Component →	i	4	5	6	7	8	12	13	14	15	23	24	25	26	27	28	29
j	Process ↓	S_{va}	S_{bu}	S_{pro}	S_{ac}	S_{h2}	S_l	X_{ch}	X_{pr}	X_{li}	X_l	S_{SO4}	S_{IS}	X_{hSRB}	X_{aSRB}	X_{pSRB}	X_{c4SRB}
19	Growth of X_{hSRB} on S_{h2}					-1						$-\frac{(1-Y_{hSRB})}{64}$	$(1-Y_{hSRB})$	Y_{hSRB}			
20	Decay of X_{hSRB}						$f_{sl,xb}$	$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$	$f_{xl,xb}$			-1			
21	Growth of X_{aSRB} on S_{ac}				-1							$-\frac{(1-Y_{aSRB})}{64}$	$(1-Y_{aSRB})$		Y_{aSRB}		
22	Decay of X_{aSRB}						$f_{sl,xb}$	$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$	$f_{xl,xb}$				-1		
23	Growth of X_{pSRB} on S_{pro}			-1	$0.57(1-Y_{pSRB})$							$-\frac{(1-Y_{pSRB})}{64}$	$0.43(1-Y_{pSRB})$			Y_{pSRB}	
24	Decay of X_{pSRB}						$f_{sl,xb}$	$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$	$f_{xl,xb}$					-1	
25	Growth of X_{c4SRB} on S_{bu}		-1		$0.8(1-Y_{c4SRB})$							$-\frac{(1-Y_{c4SRB})}{64}$	$0.2(1-Y_{c4SRB})$				Y_{c4SRB}
26	Growth of X_{c4SRB} on S_{va}	-1		$0.54(1-Y_{c4SRB})$	$0.31(1-Y_{c4SRB})$							$-\frac{(1-Y_{c4SRB})}{64}$	$0.15(1-Y_{c4SRB})$				Y_{c4SRB}
27	Decay of X_{c4SRB}						$f_{sl,xb}$	$f_{ch,xb}$	$f_{pr,xb}$	$f_{li,xb}$	$f_{xl,xb}$						-1
		Total valerate (kg COD.m ⁻³)	Total butyrate (kg COD.m ⁻³)	Total propionate (kg COD.m ⁻³)	Total acetate (kg COD.m ⁻³)	Hydrogen (kg COD.m ⁻³)	Soluble inerts (kg COD.m ⁻³)	Carbohydrates (kg COD.m ⁻³)	Proteins (kg COD.m ⁻³)	Lipids (kg COD.m ⁻³)	Particulate inerts (kg COD.m ⁻³)	Sulfate (kmol.S.m ⁻³)	Hydrogen sulphide (kg COD.m ⁻³)	H ₂ -degrading SRB (kg COD.m ⁻³)	ac-degrading SRB (kg COD.m ⁻³)	pro-degrading SRB (kg COD.m ⁻³)	C4-degrading SRB (kg COD.m ⁻³)

Table 6.4 Biochemical rate coefficients (vi,j) for Fe model extension components

Component →		i	8	25	30	31	32
j	Process ↓		S _{H2}	S _{IS}	S _{Fe3}	S _{Fe2}	S ⁰
28	Iron (II/III) Oxidation/Reduction (H ₂)		-1		$-\frac{1}{i_{Fe2}}$	$\frac{1}{i_{Fe2}}$	
29	Iron (II/III) Oxidation/Reduction (S _{IS})			-1	$-\frac{2 * i_{Fe2}}{i_{SIS}} \times \frac{1}{i_{Fe2}}$	$\frac{2 * i_{Fe2}}{i_{SIS}}$	$\frac{i_{S^0}}{i_{SIS}}$
			Hydrogen (kg COD.m ⁻³)	Hydrogen sulphide (kg COD.m ⁻³)	Iron (II) (kmol.m ⁻³)	Iron (II) (kg COD.m ⁻³)	Sulfur (kg COD.m ⁻³)

Table 6.5 Stoichiometric matrix for precipitation process

Processes		Chemical States			Rate Equations	pKa	k_{cryst}
		S _{H+}	S _{IS}	S _{Fe2}			
30	FeS precipitation	-1	1	1	$k_{cryst,FeS} X_{FeS} \left[\left(\frac{a_{Fe^{2+}} \partial_{s^{2-}}}{K_{sp,FeS}} \right)^{1/2} - 1 \right]^2$	2.95	100

Table 6.6 Stoichiometric matrix for liquid-gas transfer process

Process		Chemical States								Rate Equations
		S _{h2}	S _{ch4}	S _{iC}	S _{iS}	G _{h2}	G _{ch4}	G _{co2}	G _{h2s}	
31	H ₂ stripping	-1				1				$K_L a (S_{h_2} - 16K_{K,h_2} P_{h_{2(g)}})$
32	CH ₄ stripping		-1				1			$K_L a (S_{ch_4} - 16K_{K,ch_4} P_{ch_{4(g)}})$
33	CO ₂ stripping			-1				1		$K_L a (S_{co_2} - 16K_{K,co_2} P_{co_{2(g)}})$
34	H ₂ S stripping				-1				1	$K_L a (S_{h_2s} - 16K_{K,h_2s} P_{h_{2s(g)}})$

Table 6.7 Process kinetic rate equations (pj)

Process		Kinetic rate expression (ρ_j , kg COD m ⁻³ d ⁻¹)
1	Hydrolysis of Carbohydrates	$k_{hyd,ch} \times X_{ch}$
2	Hydrolysis of Proteins	$k_{hyd,pr} \times X_{pr}$
3	Hydrolysis of Lipids	$k_{hyd,li} \times X_{li}$
4	Uptake of Sugars	$k_{m,su} \times \frac{S_{su}}{K_S + S_{su}} \times X_{su} \times I_1$
5	Uptake of Amino acids	$k_{m,aa} \times \frac{S_{aa}}{K_S + S_{aa}} \times X_{aa} \times I_1$
6	Uptake of LFCA	$k_{m,fa} \times \frac{S_{fa}}{K_S + S_{fa}} \times X_{fs} \times I_2$
7	Uptake of Valerate	$k_{m,c4} \times \frac{S_{va}}{K_S + S_{va}} \times X_{c4} \times \frac{1}{1 + \frac{S_{bu}}{S_{va}}} \times I_3 \times I_{h2s,c4}$
8	Uptake of Butyrate	$k_{m,c4} \times \frac{S_{bu}}{K_S + S_{bu}} \times X_{c4} \times \frac{1}{1 + \frac{S_{va}}{S_{bu}}} \times I_3 \times I_{h2s,c4}$
9	Uptake of Propionate	$k_{m,pro} \times \frac{S_{pro}}{K_S + S_{pro}} \times X_{pro} \times I_4 \times I_{h2s,pro}$
10	Uptake of Acetate	$k_{m,ac} \times \frac{S_{ac}}{K_S + S_{ac}} \times X_{ac} \times I_5 \times I_{h2s,ac}$
11	Uptake of Hydrogen	$k_{m,h2} \times \frac{S_{h2}}{K_S + S_{h2}} \times X_{h2} \times I_6 \times I_{h2s,h2}$
12	Decay of X_{su}	$k_{dec,X_{su}} \times X_{su}$
13	Decay of X_{aa}	$k_{dec,X_{aa}} \times X_{aa}$
14	Decay of X_{fa}	$k_{dec,X_{fa}} \times X_{fa}$
15	Decay of X_{c4}	$k_{dec,X_{c4}} \times X_{c4}$
16	Decay of X_{pro}	$k_{dec,X_{pro}} \times X_{pro}$
17	Decay of X_{ac}	$k_{dec,X_{ac}} \times X_{ac}$
18	Decay of X_{h2}	$k_{dec,X_{h2}} \times X_{h2}$
19	Growth of X_{hSRB} on S_{h2}	$k_{m,hSRB} \times \frac{S_{SO4}}{K_{S,SO4,hSRB} + S_{SO4}} \times \frac{S_{h2}}{K_{S,hSRB} + S_{h2}} \times X_{hSRB} \times I_{SRB6} \times I_{h2s,hSRB}$
20	Decay of X_{hSRB}	$k_{dec,X_{hSRB}} \times X_{hSRB}$
21	Growth of X_{aSRB} on S_{ac}	$k_{m,aSRB} \times \frac{S_{SO4}}{K_{S,SO4,aSRB} + S_{SO4}} \times \frac{S_{ac}}{K_{S,aSRB} + S_{ac}} \times X_{aSRB} \times I_{SRB5} \times I_{h2s,aSRB}$
22	Decay of X_{aSRB}	$k_{dec,X_{aSRB}} \times X_{aSRB}$
23	Growth of X_{pSRB} on S_{pro}	$k_{m,pSRB} \times \frac{S_{SO4}}{K_{S,SO4,pSRB} + S_{SO4}} \times \frac{S_{pro}}{K_{S,pSRB} + S_{pro}} \times X_{pSRB} \times I_{SRB4} \times I_{h2s,pSRB}$
24	Decay of X_{pSRB}	$k_{dec,X_{pSRB}} \times X_{pSRB}$
25	Growth of X_{c4SRB} on S_{bu}	$k_{m,c4SRB} \times \frac{S_{SO4}}{K_{S,SO4,c4SRB} + S_{SO4}} \times \frac{S_{va}}{K_{S,c4SRB} + S_{va}} \times \frac{S_{bu}}{S_{bu} + S_{va}} \times X_{c4SRB} \times I_{SRB3} \times I_{h2s,c4SRB}$
26	Growth of X_{c4SRB} on S_{bu}	$k_{m,c4SRB} \times \frac{S_{SO4}}{K_{S,SO4,c4SRB} + S_{SO4}} \times \frac{S_{bu}}{K_{S,c4SRB} + S_{bu}} \times \frac{S_{bu}}{S_{bu} + S_{va}} \times X_{c4SRB} \times I_{SRB3} \times I_{h2s,c4SRB}$

Process		Kinetic rate expression (ρ_i , kg COD m ⁻³ d ⁻¹)
27	Decay of X_{c4SRB}	$k_{dec, X_{c4SRB}} \times X_{c4SRB}$
28	Iron (II/III) Oxidation/Reduction (S_{H2})	$\frac{K_{Fe3, Fe2} \times S_{Fe3} \times S_{h2}}{16}$
29	Iron (II/III) Oxidation/Reduction (S_{IS})	$\frac{K_{Fe3, Fe2} \times S_{Fe3} \times S_{IS}}{16}$

Inhibition factors:

$$I_1 = I_{pH, aa} \times I_{IN, lim}$$

$$I_2 = I_{pH, aa} \times I_{IN, lim} \times I_{h2, fa}$$

$$I_3 = I_{pH, aa} \times I_{IN, lim} \times I_{h2, c4}$$

$$I_4 = I_{pH, aa} \times I_{IN, lim} \times I_{h2, pro}$$

$$I_5 = I_{pH, ac} \times I_{IN, lim} \times I_{NH3, Xac}$$

$$I_6 = I_{pH, h2} \times I_{IN, lim}$$

Where:

$$I_{h2s, ac} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, ac}}}$$

$$I_{h2s, c4} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, c4}}}$$

$$I_{h2s, h2} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, h2}}}$$

$$I_{h2s, pro} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, pro}}}$$

Inhibition factors:

$$I_{SRB3} = I_{pH, aa} \times I_{IN, lim}$$

$$I_{SRB4} = I_{pH, aa} \times I_{IN, lim}$$

$$I_{SRB5} = I_{pH, ac} \times I_{IN, lim}$$

$$I_{SRB6} = I_{pH, h2} \times I_{IN, lim}$$

Where:

$$I_{h2s, aSRB} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, aSRB}}}$$

$$I_{h2s, c4SRB} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, c4SRB}}}$$

$$I_{h2s, hSRB} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, hSRB}}}$$

$$I_{h2s, pSRB} = \frac{1}{1 + \frac{Z_{h2s}}{K_{I, h2s, pSRB}}}$$

Table 6.8 Parameter values for the modified model

	Symbol	Description	Value	Units	Reference
1	$f_{ch,xb}$	Fraction of carbohydrates from biomass	0.275	-	(Flores-Alsina et al., 2016)
2	$f_{li,xb}$	Fraction of lipids from biomass	0.350	-	(Flores-Alsina et al., 2016)
3	$f_{pr,xb}$	Fraction of protein from biomass	0.275	-	(Flores-Alsina et al., 2016)
4	$f_{sl,xb}$	Fraction of soluble inerts from biomass	0.0	-	(Flores-Alsina et al., 2016)
5	$f_{xl,xb}$	Fraction of particulate inerts from biomass	0.1	--	(Flores-Alsina et al., 2016)
6	$K_{i,h2s,pro}$	50% inhibitory concentration of H ₂ S on propionate degraders	$460 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
7	$K_{i,h2s,ac}$	50% inhibitory concentration of H ₂ S on acetogens	$481 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
8	$K_{i,h2s,ch4}$	50% inhibitory concentration of H ₂ S on c4 degraders	$400 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
9	$K_{i,h2s,h2}$	50% inhibitory concentration of H ₂ S on hydrogenotrophic methanogens	$481 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
10	$k_{dec,X_{aSRB}}$	Decay rate of aSRB	0.02	d ⁻¹	(Batstone et al., 2002)
11	$k_{dec,X_{c4SRB}}$	Decay rate of c4SRB	0.02	d ⁻¹	(Batstone et al., 2002)
12	$k_{dec,X_{hSRB}}$	Decay rate of hSRB	0.02	d ⁻¹	(Batstone et al., 2002)
13	$k_{dec,X_{pSRB}}$	Decay rate of pSRB	0.02	d ⁻¹	(Batstone et al., 2002)
14	$K_{i,h2s,aSRB}$	50% inhibitory concentration of H ₂ S on aSRB	$499 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
15	$K_{i,h2s,c4SRB}$	50% inhibitory concentration of H ₂ S on c4SRB	$520 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
16	$K_{i,h2s,hSRB}$	50% inhibitory concentration of H ₂ S on hSRB	$499 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
17	$K_{i,h2s,pSRB}$	50% inhibitory concentration of H ₂ S on pSRB	$520 \cdot 10^{-3}$	kg COD.m ⁻³	(Fedorovich et al., 2003)
18	$k_{m,aSRB}$	Maximum specific uptake of aSRB	10	kg COD.kg COD ⁻¹ .d ⁻¹	(Flores-Alsina et al., 2016)
19	$k_{m,c4SRB}$	Maximum specific uptake of c4SRB	23	kg COD.kg COD ⁻¹ .d ⁻¹	(Flores-Alsina et al., 2016)
20	$k_{m,hSRB}$	Maximum specific uptake of hSRB	41.125	kg COD.kg COD ⁻¹ .d ⁻¹	(Flores-Alsina et al., 2016)
21	$k_{m,pSRB}$	Maximum specific uptake of pSRB	16.25	kg COD.kg COD ⁻¹ .d ⁻¹	(Flores-Alsina et al., 2016)
22	$K_{S,S04,aSRB}$	Half saturation constant for sulfate of aSRB	0.0002	kmol.m ⁻³	(Fedorovich et al., 2003)
23	$K_{S,S04,c4SRB}$	Half saturation constant for sulfate of c4SRB	0.0002	kmol.m ⁻³	(Fedorovich et al., 2003)
24	$K_{S,S04,hSRB}$	Half saturation constant for sulfate of hSRB	0.000104	kmol.m ⁻³	(Fedorovich et al., 2003)
25	$K_{S,S04,pSRB}$	Half saturation constant for sulfate of pSRB	0.0002	kmol.m ⁻³	(Fedorovich et al., 2003)
26	$K_{S,aSRB}$	Half saturation constant of aSRB	0.176	kg COD.m ⁻³	(Fedorovich et al., 2003)
27	$K_{S,c4SRB}$	Half saturation constant of c4SRB	0.1739	kg COD.m ⁻³	(Flores-Alsina et al., 2016)
28	$K_{S,hSRB}$	Half saturation constant of hSRB	$5.96 \cdot 10^{-6}$	kg COD.m ⁻³	(Flores-Alsina et al., 2016)
29	$K_{S,pSRB}$	Half saturation constant of pSRB	0.088	kg COD.m ⁻³	(Batstone et al., 2002)
30	Y_{aSRB}	Yield of aSRB	0.5	kg COD.kg COD ⁻¹ .d ⁻¹	(Batstone et al., 2002)
31	Y_{c4SRB}	Yield of c4SRB	0.06	kg COD.kg COD ⁻¹ .d ⁻¹	(Batstone et al., 2002)
32	Y_{hSRB}	Yield of hSRB	0.05	kg COD.kg COD ⁻¹ .d ⁻¹	(Batstone et al., 2002)
33	Y_{pSRB}	Yield of pSRB	0.04	kg COD.kg COD ⁻¹ .d ⁻¹	(Batstone et al., 2002)
34	$\dot{I}_{Fe2}$	Conversion of Fe ²⁺ into COD	8.0	kg COD.kmol ⁻¹	(Flores-Alsina et al., 2016)
35	$\dot{I}_{S^0}$	Conversion of S ⁰ into COD	48.0	kg COD.kmol ⁻¹	(Flores-Alsina et al., 2016)

36	i_{SIS}	Conversion of S_{IS} into COD	64.0	kg COD.kmol ⁻¹	(Flores-Alsina et al., 2016)
37	$K_{Fe3,Fe2}$	Conversion rate of Fe^{3+} to Fe^{2+}	10^9	m ³ .kmol ⁻¹ .day ⁻¹	(Flores-Alsina et al., 2016)

APPENDIX B THE CHEMICAL COMPOSITION OF ACID MINE DRAINAGE

Table 6.9 The Chemical Composition of Acid Mine Drainage

Element	Concentration (mg/L)	Element	Concentration (mg/L)	Element	Concentration (mg/L)
Li	0.214	W	260.427	Ga	0.692
Be	0.044	Mn	266.360	In	0.000
Na	5.930	Re	1.131	Tl	0.000
Mg	646.752	Fe	262.703	Si	0.037
K	5.052	Ru	2.587	Ge	0.000
Ca	0.000	Os	7.935	Sn	0.126
Sr	1.035	Co	9.825	Pb	0.000
Cs	0.856	Rh	0.027	P	0.122
Ba	0.000	Ir	3.778	As	0.000
Sc	0.016	Ni	8.330	Sb	0.684
Y	3.794	Pd	3.839	Bi	0.886
La	1.837	Pt	0.016	S	2801.679
Ti	0.000	Cu	0.060	Se	0.000
Zr	0.019	Ag	0.000	Te	8.216
Hf	0.000	Au	0.103	U	5.415
V	0.088	Zn	18.752	Eu	0.049
Nb	0.033	Cd	0.070	Dy	0.000
Ta	0.000	Hg	7.203	Yb	0.229
Cr	0.102	B	0.420	Th	0.407
Mo	0.637	Al	192.179	Ce	4.294

APPENDIX C MATHEMATICAL MODELLING IN MATLAB® AND SIMULINK®

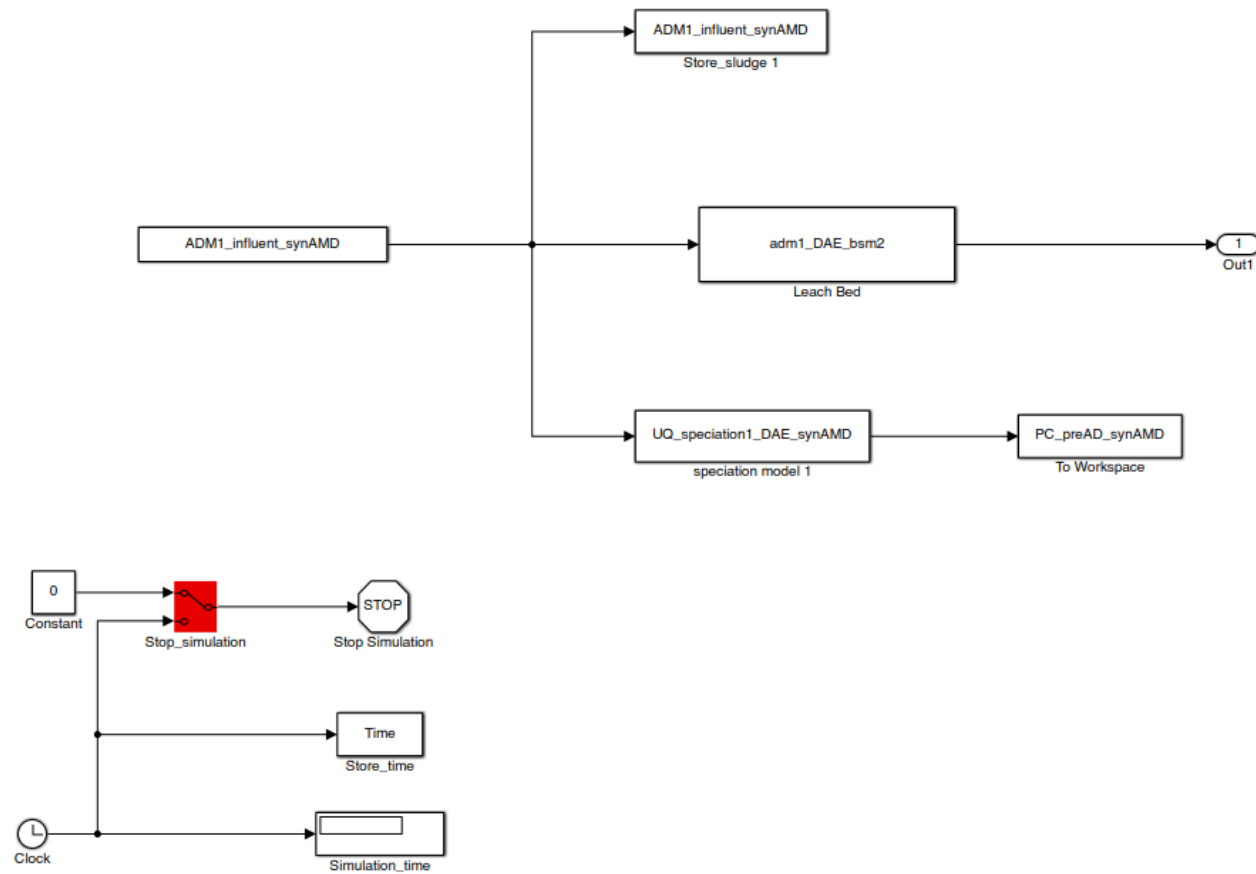


Figure 6.1 Simulink Model Page 1

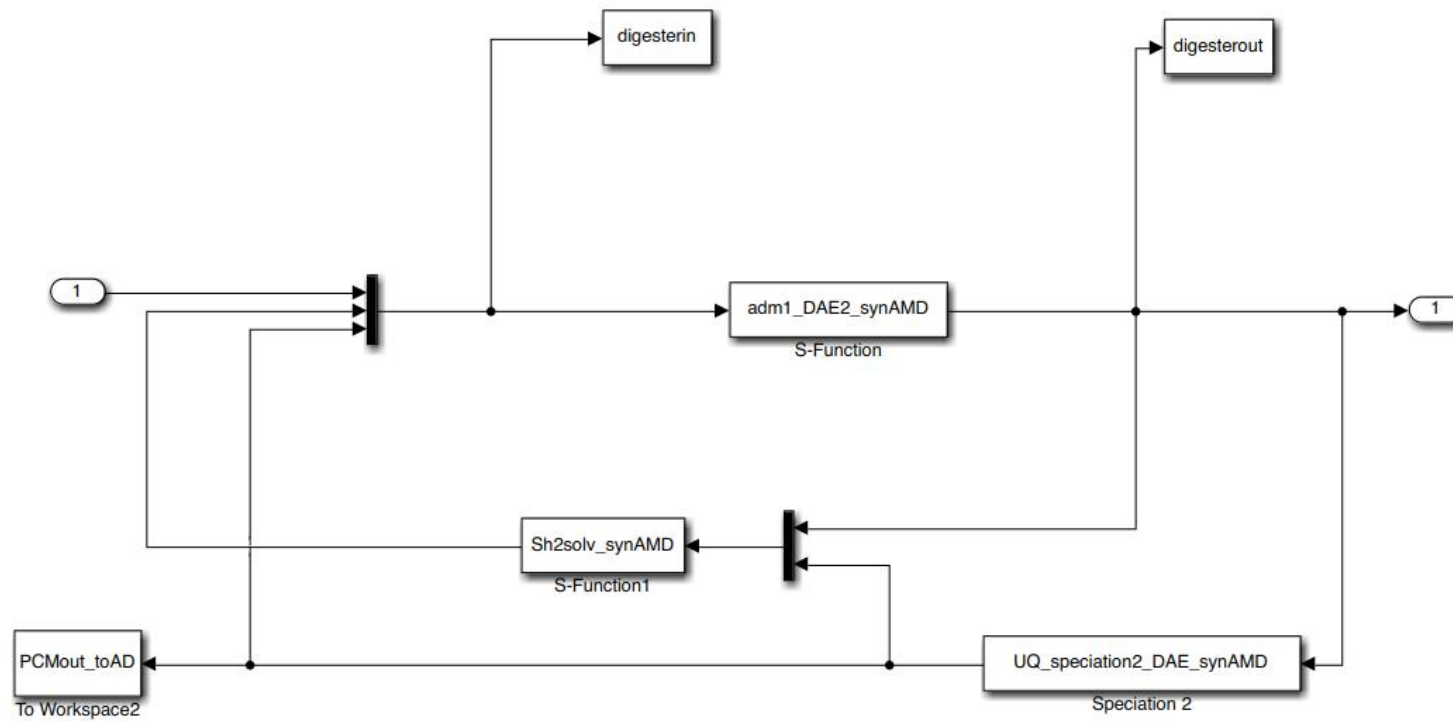


Figure 6.2 Simulink Model Page 2 (Leach Bed)

```

% This file initiates parameter values and sets initial conditions.
% Note that some of the parameter values deviate from the values given in the ADM1-ASTR
% (Batstone et al., 2002).
%
% Copyright 2006:
% Dr Christian Rosen, Dr Darko Vrecko and Dr Ulf Jeppsson
% Dept. Industrial Electrical Engineering and Automation
% Lund University, Sweden
% http://www.iea.lth.se
%
% The state values are based on BSM2 openloop results using the constant input file.
%
% Copyright: Ulf Jeppsson, IEA, Lund University, Lund, Sweden
%
% Modified 2017:
% Ms. Natalie Greenway
% Dept. Chemical Engineering
% University of the Witwatersrand, South Africa
% m file

% SOLUBLE AND PARTICULATES STATES %
%-----%
%Variables          Comments
%-----%
S_su = 0.02144;      %|
S_aa = 0.0055;       %|
S_fa = 0.1074;       %|
S_va = 0.0123;       %|
S_bu = 0.0140;       %|
S_pro = 0.0176;      %|
S_ac = 0.0893;       %|
S_h2 = 2.5055e-7;    %|
S_ch4 = 0.0555;      %|
S_IC = 0.0951;       %|
S_IN = 0.0945;       %|
S_I = 0.0000;        %|
X_xc = 0.0;          %| Eliminated X_xc from the model
X_ch = 5.94760;      %|
X_pr = 0.68850;      %|
X_li = 0.20710;      %|
X_su = 0.3122;       %|
X_aa = 0.9317;       %|
X_fa = 0.3384;       %|
X_c4 = 0.3258;       %|
X_pro = 0.1011;      %|
X_ac = 0.6772;       %|
X_h2 = 0.2848;       %|
X_I = 4.56210;       %|

% GAS STATES %
%-----%
%Variables          Comments
%-----%
S_gas_h2 = 1.1032e-5; %| H2
S_gas_ch4 = 1.6535;   %| CH4
S_gas_co2 = 0.0135;   %| CO2
S_gas_D2 = 0.01;      %| H2S

% SULFATES %
%-----%
%Variables          Comments
%-----%
S_S1_D = 0.03413;     %| S_so4
S_S2_D = 1.018e-4;    %| S_IS 0.02
S_S3_D = 0.2848;      %| X_srb_h2
S_S4_D = 0.6772;      %| X_srb_ac
S_S5_D = 0.1011;      %| X_srb_pro
S_S6_D = 0.3258;      %| X_srb_c4
S_S7_D = 0;           %| S0 elemental S

% ION STATES %
%-----%

```



```

=====
%-----
%-----
%Variables                               Comments
%-----
%-----
k_dis      = 0;                          %| Not included in this model
k_hyd_ch    = 0.7366;                     %| Biernacki et al. (2013)
k_hyd_pr    = 0.0104;                     %| Biernacki et al. (2013)
k_hyd_li    = 0.0149;                     %| Biernacki et al. (2013)
K_S_IN      = 1.0e-4;                     %| Batstone et al. (2002)
k_m_su      = 30.0;                       %| Batstone et al. (2002)
K_S_su      = 0.5;                         %| Batstone et al. (2002)
pH_UL_aa    = 5.5;                         %| Batstone et al. (2002)
pH_LL_aa    = 4.0;                         %| Batstone et al. (2002)
k_m_aa      = 50.0;                       %| Batstone et al. (2002)
K_S_aa      = 0.3;                         %| Batstone et al. (2002)
k_m_fa      = 6.0;                         %| Batstone et al. (2002)
K_S_fa      = 0.4;                         %| Batstone et al. (2002)
K_Ih2_fa    = 5.0e-6;                     %| Batstone et al. (2002)
k_m_c4      = 20.0;                       %| Batstone et al. (2002)
K_S_c4      = 0.2;                         %| Batstone et al. (2002)
K_Ih2_c4    = 1.0e-5;                     %| Batstone et al. (2002)
k_m_pro     = 13.0;                       %| Batstone et al. (2002)
K_S_pro     = 0.1;                         %| Batstone et al. (2002)
K_Ih2_pro   = 3.5e-6;                     %| Batstone et al. (2002)
k_m_ac      = 8.0;                         %| Batstone et al. (2002)
K_S_ac      = 0.15;                       %| Batstone et al. (2002)
K_I_nh3     = 0.0018;                     %| Batstone et al. (2002)
pH_UL_ac    = 7.0;                         %| Batstone et al. (2002)
pH_LL_ac    = 6.0;                         %| Batstone et al. (2002)
k_m_h2      = 35.0;                       %| Batstone et al. (2002)
K_S_h2      = 7.0e-6;                     %| Batstone et al. (2002)
pH_UL_h2    = 6.0;                         %| Batstone et al. (2002)
pH_LL_h2    = 5.0;                         %| Batstone et al. (2002)
k_dec_Xsu   = 0.02;                       %| Batstone et al. (2002)
k_dec_Xaa   = 0.02;                       %| Batstone et al. (2002)
k_dec_Xfa   = 0.02;                       %| Batstone et al. (2002)
k_dec_Xc4   = 0.02;                       %| Batstone et al. (2002)
k_dec_Xpro  = 0.02;                       %| Batstone et al. (2002)
k_dec_Xac   = 0.02;                       %| Batstone et al. (2002)
k_dec_Xh2   = 0.02;                       %| Batstone et al. (2002)
=====
%PHYSICOCHEMICAL PARAMETER VALUES
=====
%-----
%Variables                               Comments
%-----
R_cte      = 0.083145;                    %| universal gas constant dm3*bar/(mol*K) = 8.3145
J/(mol*K)
T_base     = 298.15;                       %| 25 degC = 298.15 K
T_op       = 298.15;                       %| operational temperature of AD
pK_w_base  = 14.0;                         %| Batstone et al. (2002)
pK_a_va_base = 4.86;                       %| Batstone et al. (2002)
pK_a_bu_base = 4.82;                       %| Batstone et al. (2002)
pK_a_pro_base = 4.88;                       %| Batstone et al. (2002)
pK_a_ac_base = 4.76;                       %| Batstone et al. (2002)
pK_a_co2_base = 6.35;                       %| Batstone et al. (2002)
pK_a_IN_base = 9.25;                       %| Batstone et al. (2002)
k_A_Bva    = 1.0e10;                       %| 1e8; according to STR
k_A_Bbu    = 1.0e10;                       %| 1e8; according to STR
k_A_Bpro   = 1.0e10;                       %| 1e8; according to STR
k_A_Bac    = 1.0e10;                       %| 1e8; according to STR
k_A_Bco2   = 1.0e10;                       %| 1e8; according to STR
k_A_BIN    = 1.0e10;                       %| 1e8; according to STR
P_atm      = 1.013;                         %| bar
kLa        = 200.0;                         %| Batstone et al. (2002)
K_H_h2o_base = 0.0313;                     %| Batstone et al. (2002)
K_H_co2_base = 0.035;                       %| Batstone et al. (2002)
K_H_ch4_base = 0.0014;                     %| Batstone et al. (2002)
K_H_h2_base = 7.8e-4;                       %| Batstone et al. (2002)
k_P        = 5.0e4;                         %| Batstone et al. (2002)

N_pr = 0.0;

=====
%SULFATE PARAMETER VALUES

```

```

%=====
%-----
%Variables                                     Comments
%-----
Y_c4SRB = 0.0329;                             %| Fedorovich & Kalyuzhnyi (2003)
Y_pSRB = 0.0329;                             %| Fedorovich & Kalyuzhnyi (2003)
Y_aSRB = 0.0324;                             %| Fedorovich & Kalyuzhnyi (2003)
Y_hSRB = 0.0366;                             %| Fedorovich & Kalyuzhnyi (2003)

k_m_c4SRB = 20.0;                             %| assumed as ADM1 * correction factor to outcompute
X_va
k_m_pSRB = 13.0;                             %| assumed as ADM1 * correction factor to outcompute
X_pro
k_m_aSRB = 8.0;                             %| assumed as ADM1 * correction factor to outcompute
X_ac
k_m_hSRB = 35.0*1.175;                       %| assumed as ADM1 * correction factor to outcompute
X_H2

K_S_c4SRB = 0.20;                             %| assumed as ADM1 * correction factor to outcompute
X_va
K_S_pSRB = 0.11;                             %| assumed as ADM1 * correction factor to outcompute
X_pro
K_S_aSRB = 0.22;                             %| assumed as ADM1 * correction factor to outcompute
X_ac
K_S_hSRB = 7.0e-6/1.175;                     %| assumed as ADM1 * correction factor to outcompute
X_H2

K_S_so4_c4SRB = 0.00021;                     %| Fedorovich & Kalyuzhnyi (2003)
K_S_so4_pSRB = 0.0002;                       %| Fedorovich & Kalyuzhnyi (2003)
K_S_so4_aSRB = 0.0001;                       %| Fedorovich & Kalyuzhnyi (2003)
K_S_so4_hSRB = 0.000104;                     %| Fedorovich & Kalyuzhnyi (2003)

K_I_h2s_c4 = 7.53e-3*64.0*1.5;               %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_pro = 7.53e-3*64.0*1.5;              %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_ac = 7.19e-3*64.0*0.75;              %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_h2 = 6.25e-3*64.0*0.75;              %| Fedorovich & Kalyuzhnyi (2003);

K_I_h2s_c4SRB = 8.13e-3*64.0*2;              %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_pSRB = 8.13e-3*64.0*2;              %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_aSRB = 7.80e-3*64.0*0.75;           %| Fedorovich & Kalyuzhnyi (2003);
K_I_h2s_hSRB = 7.80e-3*64.0*0.75;           %| Fedorovich & Kalyuzhnyi (2003);

pH_UL_c4SRB = 7.0;                           %| Flores-Alsina et al. (2016);
pH_LL_c4SRB = 6.0;                           %| Flores-Alsina et al. (2016);
pH_UL_pSRB = 7.0;                           %| Flores-Alsina et al. (2016);
pH_LL_pSRB = 6.0;                           %| Flores-Alsina et al. (2016);
pH_UL_aSRB = 7.0;                           %| Flores-Alsina et al. (2016);
pH_LL_aSRB = 6.0;                           %| Flores-Alsina et al. (2016);
pH_UL_hSRB = 6.0;                           %| Flores-Alsina et al. (2016);
pH_LL_hSRB = 5.0;                           %| Flores-Alsina et al. (2016);

k_dec_Xc4SRB = 0.02;                         %| Batstone et al. (2002)
k_dec_XpSRB = 0.02;                         %| Batstone et al. (2002)
k_dec_XaSRB = 0.02;                         %| Batstone et al. (2002)
k_dec_XhSRB = 0.02;                         %| Batstone et al. (2002)

K_H_h2s_base = 0.105;                       %| Flores-Alsina et al. (2016)

%=====
%IRON PARAMETER VALUES
%=====
%-----
%Variables                                     Comments
%-----
COD_Fe3 = 0.0;                               %| Flores-Alsina et al. (2016)
COD_Fe2 = 8.0;                               %| Flores-Alsina et al. (2016)
i_Fe3_Fe2 = 8.0;                             %| Flores-Alsina et al. (2016)
K1_Fe3_Fe2 = 1e10;                           %| Flores-Alsina et al. (2016)
K2_Fe3_Fe2 = 1e5;                            %| Flores-Alsina et al. (2016);

%:.....%
DIGESTERPAR = [ f_sI_xc f_xI_xc f_ch_xc f_pr_xc f_li_xc N_xc N_I N_aa C_xc C_sI C_ch
C_pr C_li C_xI C_su C_aa f_fa_li C_fa ...
f_h2_su f_bu_su f_pro_su f_ac_su N_bac C_bu C_pro C_ac C_bac Y_su f_h2_aa f_va_aa
f_bu_aa f_pro_aa f_ac_aa C_va Y_aa Y_fa ...
Y_c4 Y_pro C_ch4 Y_ac Y_h2 k_dis k_hyd_ch k_hyd_li K_S_IN k_m_su K_S_su
pH_UL_aa pH_LL_aa k_m_aa K_S_aa k_m_fa ...

```

```

K_S_fa K_Ih2_fa k_m_c4 K_S_c4 K_Ih2_c4 k_m_pro K_S_pro K_Ih2_pro k_m_ac K_S_ac K_I_nh3
pH_UL_ac pH_LL_ac k_m_h2 K_S_h2 ...
pH_UL_h2 pH_LL_h2 k_dec_Xsu k_dec_Xaa k_dec_Xfa k_dec_Xc4 k_dec_Xpro k_dec_Xac k_dec_Xh2
R_cte T_base T_op pK_w_base pK_a_va_base pK_a_bu_base ...
pK_a_pro_base pK_a_ac_base pK_a_co2_base pK_a_IN_base k_A_Bva k_A_Bbu k_A_Bpro k_A_Bac
k_A_Bco2 k_A_BIN P_atm kLa ...
K_H_h2o_base K_H_co2_base K_H_ch4_base K_H_h2_base k_P ...
N_pr...
Y_c4SRB Y_pSRB Y_aSRB Y_hSRB k_m_c4SRB k_m_pSRB k_m_aSRB k_m_hSRB K_S_c4SRB K_S_pSRB
K_S_aSRB K_S_hSRB K_S_so4_c4SRB K_S_so4_pSRB K_S_so4_aSRB K_S_so4_hSRB...
K_I_h2s_c4 K_I_h2s_pro K_I_h2s_ac K_I_h2s_h2 K_I_h2s_c4SRB K_I_h2s_pSRB K_I_h2s_aSRB
K_I_h2s_hSRB...
pH_UL_c4SRB pH_LL_c4SRB pH_UL_pSRB pH_LL_pSRB pH_UL_aSRB pH_LL_aSRB pH_UL_hSRB
pH_LL_hSRB...
k_dec_Xc4SRB k_dec_XpSRB k_dec_XaSRB k_dec_XhSRB K_H_h2s_base...
COD_Fe3 COD_Fe2 i_Fe3_Fe2 K1_Fe3_Fe2 K2_Fe3_Fe2];
%:.....%

%=====
% PHYSICAL PARAMETER VALUES
%=====
%-----
%Variables Unit Comments
%-----
V_liq = 0.005; % m^3 %|
V_gas = 0.0004; % m^3 %|

DIM_D = [ V_liq V_gas ]; %|

K_Fe3_Fe2 = 1000000000.0; %|
K_Fe3_Fe2_PAR = [K_Fe3_Fe2]; %|

% Input initial guesses for S-function UQ_speciation1_DAE_synAMD (m file)
%=====
%-----
%Species Comments
%-----
x0 = zeros(1,13);
%-----
x0(1) = 1.28E-07; %| H+
x0(2) = 0.069889004; %| NH4+
x0(3) = 0.001038179; %| Ac-
x0(4) = 0.000116706; %| Pro-
x0(5) = 1.637e-05; %| CO3-2
x0(6) = 2.932e-08; %| SO4-2
x0(7) = 10e-12; %| Fe2+
x0(8) = 10e-12; %| Fe3+
x0(9) = 4.101e-06; %| HS-
x0(10) = 6.50852E-05; %| Bu-
x0(11) = 4.40839E-05; %| Va-
x0(12) = 0.0; %| NO-3; need to confirm
x0(13) = 0.0; %| NO-2; need to confirm
x0(14) = 0.089170442; %| mu
x0(15) = 0.745779332; %| gamma

```

```

CHEM_INIT_AD2 = [x0];

%-----%
%Species          Comments
%-----%
x1 = zeros(1,15);
%-----%
x1(1) = 2.1490E-07;    %| H+
x1(2) = 3.2467E-04;    %| NH4+
x1(3) = 1.8612E-04;    %| Ac-
x1(4) = 6.5936E-06;    %| Pro-
x1(5) = 5.4803E-06;    %| CO3-2
x1(6) = 2.5111E-04;    %| SO4-2
x1(7) = 10e-12;        %| Fe2+
x1(8) = 10e-12;        %| Fe3+
x1(9) = 10e-12;        %| HS-
x1(10) = 6.50852E-05;   %| Bu-
x1(11) = 4.40839E-05;   %| Va-
x0(12) = 0.0;          %| NO-3; need to confirm
x0(13) = 0.0;          %| NO-2; need to confirm
x1(14) = 0.089170442;   %| mu
x1(15) = 0.745779332;   %| gamma

CHEM_INIT_AD1 = [x1];

% Equilibrium constant at 25 oC (MINTEQA database in PHREEQC)
%=====

% -logK          Equilibrium constant K          Reaction
%=====
pK_FeAc2 = -7.57;    K_FeAc2 = 10^(-pK_FeAc2);    % Fe+2 + 2Ac- <-->
Fe(Ac)2+
pK_FeAc3_aq = -9.5867; K_FeAc3_aq = 10^(-pK_FeAc3_aq);    % Fe+3 + 3Ac- <-->
Fe(Ac)3
pK_FeN2H6 = 16.24;    K_FeN2H6 = 10^(-pK_FeN2H6);    % Fe+2 + 2NH4+ <-->
Fe(NH3)2+2 + 2H+
pK_FeN3H9 = 25.05;    K_FeN3H9 = 10^(-pK_FeN3H9);    % Fe+2 + 3NH4+ <-->
Fe(NH3)3+2 + 2H+
pK_FeN4H12 = 34.23;    K_FeN4H12 = 10^(-pK_FeN4H12);    % Fe+2 + 4NH4+ <-->
Fe(NH3)4+2 + 2H+
pK_FeOH2_aq = 20.494;    K_FeOH2_aq = 10^(-pK_FeOH2_aq);    % Fe+2 + 2OH- <-->
Fe(OH)2(aq)
pK_FeOH2 = 5.75;    K_FeOH2 = 10^(-pK_FeOH2);    % Fe+3 + 2OH- <-->
Fe(OH)2+
pK_FeOH3 = 30.991;    K_FeOH3 = 10^(-pK_FeOH3);    % Fe+2 + 3OH- <-->
Fe(OH)3-
pK_FeOH3_aq = 15;    K_FeOH3_aq = 10^(-pK_FeOH3_aq);    % Fe+3 + 3OH- <-->
Fe(OH)3_aq
pK_FeOH4 = 22.7;    K_FeOH4 = 10^(-pK_FeOH4);    % Fe+3 + 4OH- <-->
Fe(OH)4-
pK_FeS2O8 = -5.38;    K_FeS2O8 = 10^(-pK_FeS2O8);    % Fe+3 + 2SO4-2 <-->
Fe(SO4)2-
pK_Fe2O2H2 = 2.894;    K_Fe2O2H2 = 10^(-pK_Fe2O2H2);    % 2Fe+2 + 2OH- <-->
Fe2(OH)2+4
pK_Fe3O4H4 = 6.288;    K_Fe3O4H4 = 10^(-pK_Fe3O4H4);    % 3Fe+2 + 4OH- <-->
Fe3(OH)4+5
pK_FeAc = -1.4;    K_FeAc = 10^(-pK_FeAc);    % Fe+2 + Ac- <-->
FeAc+
pK_FeAc_2 = -4.24;    K_FeAc_2 = 10^(-pK_FeAc_2);    % Fe+3 + Ac- <-->
FeAc+2
pK_FeBu_2 = -3.41;    K_FeBu_2 = 10^(-pK_FeBu_2);    % Fe+3 + Bu- <-->
FeBu+2
pK_FeHCO3 = -11.429;    K_FeHCO3 = 10^(-pK_FeHCO3);    % Fe+2 + H+ + CO3-2 <-->
-> FeHCO3+
pK_FeHS = -5.62;    K_FeHS = 10^(-pK_FeHS);    % Fe+2 + HS- <-->
FeHS+
pK_FeNH3 = 7.84;    K_FeNH3 = 10^(-pK_FeNH3);    % Fe+2 + NH4+ <-->
FeNH3+2 + H+
pK_FeOH = 9.397;    K_FeOH = 10^(-pK_FeOH);    % Fe+2 + OH- <-->
FeOH+
pK_FeOH_2 = 2.02;    K_FeOH_2 = 10^(-pK_FeOH_2);    % Fe+3 + OH- <-->
FeOH+2
pK_FePr = -3.71;    K_FePr = 10^(-pK_FePr);    % Fe+3 + Pr- <-->
FePr+2
pK_FeSO4_aq = -2.39;    K_FeSO4_aq = 10^(-pK_FeSO4_aq);    % Fe+2 + SO4-2 <-->
FeSO4(aq)

```

```

pK_FeSO4 = -4.25;      K_FeSO4 = 10^(-pK_FeSO4);      % Fe+3 + SO4-2 <-->
FeSO4+
pK_FeVa_2 = -3.51;      K_FeVa_2 = 10^(-pK_FeVa_2);      % Fe+3 + Va- <-->
FeVa+2
pK_H2CO3_aq = -16.681;  K_H2CO3_aq = 10^(-pK_H2CO3_aq);  % 2H^+ + CO3-2 <-->
H2CO3
pK_H2S_aq = -7.02;      K_H2S_aq = 10^(-pK_H2S_aq);      % H+ + HS- <-->
H2S(aq)
pK_HAc_aq = -4.757;      K_HAc_aq = 10^(-pK_HAc_aq);      % H+ + Ac- <--> HAc+
pK_HBu_aq = -4.818;      K_HBu_aq = 10^(-pK_HBu_aq);      % H+ + Bu- <--> HBu+
pK_HCO3 = -10.329;      K_HCO3 = 10^(-pK_HCO3);      % H^+ + CO3-2 <-->
HCO3-
pK_HPr_aq = -4.874;      K_HPr_aq = 10^(-pK_HPr_aq);      % H+ + Pr- <--> HPr+
pK_HSO4 = -1.99;      K_HSO4 = 10^(-pK_HSO4);      % H+ + SO4-2 <-->
HSO4-2
pK_HVa_aq = -4.843;      K_HVa_aq = 10^(-pK_HVa_aq);      % H+ + Va- <--> HVa+
pK_NH3_aq = 9.244;      K_NH3_aq = 10^(-pK_NH3_aq);      % NH4+ <--> NH3 + H+
pK_NH4SO4 = -1.03;      K_NH4SO4 = 10^(-pK_NH4SO4);      % NH4+ + SO4-2 <-->
NH4SO4-
pK_H2O = 13.997;      K_H2O = 10^(-pK_H2O);      % H_2 O - H^+ <-->
OH^-
pK_S = 17.4;      K_S = 10^(-pK_S);      % HS- - H+ <--> S-2
pK_CO2 = -18.147;      K_CO2 = 10^(-pK_CO2);      % CO3-2 + 2H+ -->
CO2(g))+H2O
pK_FeNO2_2 = -4.72;      K_FeNO2_2 = 10^(-pK_FeNO2_2);
pK_FeNO2_aq = -6.78;      K_FeNO2_aq = 10^(-pK_FeNO2_aq);
pK_FeNO2 = -3.2;      K_FeNO2 = 10^(-pK_FeNO2);
pK_HNO2_aq = -3.15;      K_HNO2_aq = 10^(-pK_HNO2_aq);

% Heat of reaction (Joule/mole) at 25 oC (MINTEQ database in PHREEQC)
% =====

DH_FeAc2 = 0;      %|
DH_FeAc3_aq = 0;      %|
DH_FeN2H6 = 89000;      %|
DH_FeN3H9 = 133000;      %|
DH_FeN4H12 = 177000;      %|
DH_FeOH2_aq = 119620;      %|
DH_FeOH2 = 37700;      %|
DH_FeOH3 = 126430;      %|
DH_FeOH3_aq = 75300;      %|
DH_FeOH4 = 154800;      %|
DH_FeS2O8 = 19200;      %|
DH_Fe2O2H2 = 56420;      %|
DH_Fe3O4H4 = 65240;      %|
DH_FeAc = 0;      %|
DH_FeAc_2 = 25000;      %|
DH_FeBu_2 = 11000;      %|
DH_FeHCO3 = 0;      %|
DH_FeHS = 0;      %|
DH_FeNH3 = 44100;      %|
DH_FeOH = -55810;      %|
DH_FeOH_2 = 25100;      %|
DH_FePr = 21000;      %|
DH_FeSO4_aq = 8000;      %|
DH_FeSO4 = 25000;      %|
DH_FeVa_2 = 13000;      %|
DH_H2CO3_aq = -32000;      %|
DH_H2S_aq = -22000;      %|
DH_HAc_aq = 410;      %|
DH_HBu_aq = 2800;      %|
DH_HCO3 = -14600;      %|
DH_HPr_aq = 750;      %|
DH_HSO4 = 22000;      %|
DH_HVa_aq = 2800;      %|
DH_NH3_aq = 52000;      %|
DH_NH4SO4 = 0;      %|
DH_H2O = 55810;      %|
DH_S = 49400;      %|
DH_CO2 = -4060;      %|
DH_FeNO2_2 = 0.0;      %|
DH_FeNO2_aq = 0.0;      %|
DH_FeNO2 = 0.0;      %|
DH_HNO2_aq = 0.0;      %|

% -----
% Gas constant

```

```

% -----%
R = 8.314; % J/mole/K

% -----%
% Calculate the temperature in Kelvin
% -----%
Temp_C = 25.0;
Temp_K = Temp_C + 273.15;

CHEMAD_PAR1 =
[K_FeAc2,K_FeAc3_aq,K_FeN2H6,K_FeN3H9,K_FeN4H12,K_FeOH2_aq,K_FeOH2,K_FeOH3,K_FeOH3_aq,K_
FeOH4,K_FeS2O8,K_Fe2O2H2,K_Fe3O4H4,K_FeAc,K_FeAc_2,K_FeBu_2,K_FeHCO3...

K_FeHS,K_FeNH3,K_FeOH,K_FeOH_2,K_FePr,K_FeSO4_aq,K_FeSO4,K_FeVa_2,K_H2CO3_aq,K_H2S_aq,K_
HAc_aq,K_HBu_aq,K_HCO3...
K_HPr_aq,K_HSO4,K_HVa_aq,K_NH3_aq,K_NH4SO4,K_H2O,K_S,K_CO2...
K_FeNO2_2,K_FeNO2_aq,K_FeNO2,K_HNO2_aq];

CHEMAD_PAR2 =
[DH_FeAc2,DH_FeAc3_aq,DH_FeN2H6,DH_FeN3H9,DH_FeN4H12,DH_FeOH2_aq,DH_FeOH2,DH_FeOH3,DH_Fe
OH3_aq,DH_FeOH4,DH_FeS2O8,DH_Fe2O2H2,DH_Fe3O4H4,DH_FeAc,DH_FeAc_2,
DH_FeBu_2,DH_FeHCO3...

DH_FeHS,DH_FeNH3,DH_FeOH,DH_FeOH_2,DH_FePr,DH_FeSO4_aq,DH_FeSO4,DH_FeVa_2,DH_H2CO3_aq,DH
_H2S_aq,DH_HAc_aq,DH_HBu_aq,DH_HCO3...
DH_HPr_aq,DH_HSO4,DH_HVa_aq,DH_NH3_aq,DH_NH4SO4,DH_H2O,DH_S,DH_CO2...
DH_FeNO2_2,DH_FeNO2_aq,DH_FeNO2,DH_HNO2_aq];

% -----%
% Activity coefficient temperature dependent parameter A
Comments
% -----%
d = 1 - (Temp_C - 3.9863)^2*(Temp_C +288.9414)/(508929.2*(Temp_C +
68.12963))+0.011445*exp(-374.3/Temp_C); %| density

%|
eps = 2727.586 + 0.6224107*Temp_K - 466.9151*log(Temp_K) - 52000.87/Temp_K; %|
dielectric constant

%|
A = 1.82483*10^6*d^0.5/(eps*Temp_K)^(3/2); %| Temperature dependent parameter

%|
ACTIVITYAD_2 = [d eps A];
%|

```

```

/*
 * UQ_precipitation_DAEs model block
 * C file – S function UQ_speciation1_DAE_synAMD
 * Copyright: Xavier Flores-Alsina, Ulf Jeppsson, IEA, Lund University, Lund, Sweden
 *           Krist V. Gernaey, DTU, Denmark
 *           November 2013
 *
 * Edited: Natalie Greenway
 *         University of the Witwatersrand
 *         July 2017
 */

#define S_FUNCTION_NAME UQ_speciation1_DAE_synAMD
#include "simstruc.h"
#include "mex.h"
#include "lapack.h"
#include <math.h>
#include "matrix.h"

#define CHEM_INIT    ssGetArg(S,0)    /* initial values */
#define CHEM_PAR     ssGetArg(S,1)    /* chemical parameters */
#define CHEM_PAR1    ssGetArg(S,2)    /* chemical parameters */
#define ACTIVITY     ssGetArg(S,3)    /* activity */

double *umain;
SimStruct *Smain;

/*
 * mdlInitializeSizes - initialize the sizes array
 */

static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumContStates( S, 0); /* number of continuous states */
    ssSetNumDiscStates( S, 15); /* number of discrete states*/
    ssSetNumInputs(     S, 36); /* number of inputs */
    ssSetNumOutputs(    S, 72); /* number of outputs */
}

```

```

ssSetDirectFeedThrough(S, 0); /* direct feed through flag */
ssSetNumSampleTimes( S, 1); /* number of sample times */
ssSetNumSFcnParams( S, 4); /* number of input arguments */
ssSetNumRWork( S, 0); /* number of real work vector elements */
ssSetNumIWork( S, 0); /* number of integer work vector elements*/
ssSetNumPWork( S, 0); /* number of pointer work vector elements*/
}

/*
 * mdlInitializeSampleTimes - initialize the sample times array
 */
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, INHERITED_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0.0);
}

/*
 * mdlInitializeConditions - initialize the states
 */
static void mdlInitializeConditions(double *x0, SimStruct *S)
{
    int i;

    for (i = 0; i < 15; i++) {
        x0[i] = mxGetPr(CHEM_INIT)[i];
    }
}

/*
 * mdlOutputs - compute the outputs
 */
static void mdlOutputs(double *y, double *x, double *u, SimStruct *S, int tid)
{
    double S_act_H, S_act_NH4, S_act_Ac, S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
    S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;
    double gama2, gama3, gama4, gama5;

    double K_FeAc2, K_FeAc3_aq, K_FeN2H6, K_FeN3H9, K_FeN4H12, K_FeOH2_aq, K_FeOH2;
    double
    K_FeOH3, K_FeOH3_aq, K_FeOH4, K_FeS2O8, K_Fe2O2H2, K_Fe3O4H4, K_FeAc, K_FeAc_2, K_FeBu_2, K_FeHCO3, K_
    FeHS, K_FeNH3, K_FeOH, K_FeOH_2, K_FePr, K_FeSO4_aq, K_FeSO4, K_FeVa_2, K_H2CO3_aq, K_H2S_aq, K_HAc_aq,
    K_HBu_aq, K_HCO3, K_HPr_aq, K_HSO4, K_HVa_aq;
    double K_NH3_aq, K_NH4SO4, K_H2O, K_S, K_CO2;
    double K_FeNO2_2, K_FeNO2_aq, K_FeNO2, K_HNO2_aq;

    double DH_FeAc2, DH_FeAc3_aq, DH_FeN2H6, DH_FeN3H9, DH_FeN4H12, DH_FeOH2_aq, DH_FeOH2;
    double
    DH_FeOH3, DH_FeOH3_aq, DH_FeOH4, DH_FeS2O8, DH_Fe2O2H2, DH_Fe3O4H4, DH_FeAc, DH_FeAc_2, DH_FeBu_2,
    DH_FeHCO3, DH_FeHS, DH_FeNH3, DH_FeOH, DH_FeOH_2, DH_FePr, DH_FeSO4_aq, DH_FeSO4, DH_FeVa_2, DH_H2
    CO3_aq, DH_H2S_aq, DH_HAc_aq, DH_HBu_aq, DH_HCO3, DH_HPr_aq, DH_HSO4, DH_HVa_aq;
    double DH_NH3_aq, DH_NH4SO4, DH_H2O, DH_S, DH_CO2;
    double DH_FeNO2_2, DH_FeNO2_aq, DH_FeNO2, DH_HNO2_aq;

    double KT_FeAc2, KT_FeAc3_aq, KT_FeN2H6, KT_FeN3H9, KT_FeN4H12, KT_FeOH2_aq, KT_FeOH2;
    double
    KT_FeOH3, KT_FeOH3_aq, KT_FeOH4, KT_FeS2O8, KT_Fe2O2H2, KT_Fe3O4H4, KT_FeAc, KT_FeAc_2, KT_FeBu_2, KT
    _FeHCO3, KT_FeHS, KT_FeNH3, KT_FeOH, KT_FeOH_2, KT_FePr, KT_FeSO4_aq, KT_FeSO4, KT_FeVa_2, KT_H2CO3_
    aq, KT_H2S_aq, KT_HAc_aq, KT_HBu_aq, KT_HCO3, KT_HPr_aq, KT_HSO4, KT_HVa_aq;
    double KT_NH3_aq, KT_NH4SO4, KT_H2O, KT_S, KT_CO2;
    double KT_FeNO2_2, KT_FeNO2_aq, KT_FeNO2, KT_HNO2_aq;

```

```
double R, Temp_K, Temp_C;
```

```
double xtemp[15];
```

```
int i;
```

```
K_FeAc2   = mxGetPr(CHEM_PAR)[0];  
K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];  
K_FeN2H6   = mxGetPr(CHEM_PAR)[2];  
K_FeN3H9   = mxGetPr(CHEM_PAR)[3];  
K_FeN4H12  = mxGetPr(CHEM_PAR)[4];  
K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];  
K_FeOH2    = mxGetPr(CHEM_PAR)[6];  
K_FeOH3    = mxGetPr(CHEM_PAR)[7];  
K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];  
K_FeOH4    = mxGetPr(CHEM_PAR)[9];  
K_FeS2O8   = mxGetPr(CHEM_PAR)[10];  
K_Fe2O2H2  = mxGetPr(CHEM_PAR)[11];  
K_Fe3O4H4  = mxGetPr(CHEM_PAR)[12];  
K_FeAc     = mxGetPr(CHEM_PAR)[13];  
K_FeAc_2   = mxGetPr(CHEM_PAR)[14];  
K_FeBu_2   = mxGetPr(CHEM_PAR)[15];  
K_FeHCO3   = mxGetPr(CHEM_PAR)[16];  
K_FeHS     = mxGetPr(CHEM_PAR)[17];  
K_FeNH3    = mxGetPr(CHEM_PAR)[18];  
K_FeOH     = mxGetPr(CHEM_PAR)[19];  
K_FeOH_2   = mxGetPr(CHEM_PAR)[20];  
K_FePr     = mxGetPr(CHEM_PAR)[21];  
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];  
K_FeSO4    = mxGetPr(CHEM_PAR)[23];  
K_FeVa_2   = mxGetPr(CHEM_PAR)[24];  
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];  
K_H2S_aq   = mxGetPr(CHEM_PAR)[26];  
K_HAc_aq   = mxGetPr(CHEM_PAR)[27];  
K_HBu_aq   = mxGetPr(CHEM_PAR)[28];  
K_HCO3     = mxGetPr(CHEM_PAR)[29];  
K_HPr_aq   = mxGetPr(CHEM_PAR)[30];  
K_HSO4     = mxGetPr(CHEM_PAR)[31];  
K_HVa_aq   = mxGetPr(CHEM_PAR)[32];  
K_NH3_aq   = mxGetPr(CHEM_PAR)[33];  
K_NH4SO4   = mxGetPr(CHEM_PAR)[34];  
K_H2O      = mxGetPr(CHEM_PAR)[35];  
K_S        = mxGetPr(CHEM_PAR)[36];  
K_CO2      = mxGetPr(CHEM_PAR)[37];  
K_FeNO2_2  = mxGetPr(CHEM_PAR)[38];  
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];  
K_FeNO2    = mxGetPr(CHEM_PAR)[40];  
K_HNO2_aq  = mxGetPr(CHEM_PAR)[41];
```

```
DH_FeAc2   = mxGetPr(CHEM_PAR1)[0];  
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];  
DH_FeN2H6   = mxGetPr(CHEM_PAR1)[2];  
DH_FeN3H9   = mxGetPr(CHEM_PAR1)[3];  
DH_FeN4H12  = mxGetPr(CHEM_PAR1)[4];  
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];  
DH_FeOH2    = mxGetPr(CHEM_PAR1)[6];  
DH_FeOH3    = mxGetPr(CHEM_PAR1)[7];  
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];  
DH_FeOH4    = mxGetPr(CHEM_PAR1)[9];  
DH_FeS2O8   = mxGetPr(CHEM_PAR1)[10];  
DH_Fe2O2H2  = mxGetPr(CHEM_PAR1)[11];  
DH_Fe3O4H4  = mxGetPr(CHEM_PAR1)[12];  
DH_FeAc     = mxGetPr(CHEM_PAR1)[13];  
DH_FeAc_2   = mxGetPr(CHEM_PAR1)[14];
```

```

DH_FeBu_2 = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3 = mxGetPr(CHEM_PAR1)[16];
DH_FeHS = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3 = mxGetPr(CHEM_PAR1)[18];
DH_FeOH = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2 = mxGetPr(CHEM_PAR1)[20];
DH_FePr = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4 = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2 = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq = mxGetPr(CHEM_PAR1)[27];
DH_HBu_aq = mxGetPr(CHEM_PAR1)[28];
DH_HCO3 = mxGetPr(CHEM_PAR1)[29];
DH_HPr_aq = mxGetPr(CHEM_PAR1)[30];
DH_HSO4 = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq = mxGetPr(CHEM_PAR1)[33];
DH_NH4SO4 = mxGetPr(CHEM_PAR1)[34];
DH_H2O = mxGetPr(CHEM_PAR1)[35];
DH_S = mxGetPr(CHEM_PAR1)[36];
DH_CO2 = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2 = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq = mxGetPr(CHEM_PAR1)[41];

R = 8.314; // J/mole/K
Temp_C = 25.0; // 15.0
Temp_K = Temp_C + 273.15;

KT_FeAc2 = pow(1.0E1,log(K_FeAc2)/log(1.0E1)-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1)-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6 = pow(1.0E1,log(K_FeN2H6)/log(1.0E1)-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9 = pow(1.0E1,log(K_FeN3H9)/log(1.0E1)-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12 = pow(1.0E1,log(K_FeN4H12)/log(1.0E1)-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1)-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2 = pow(1.0E1,log(K_FeOH2)/log(1.0E1)-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3 = pow(1.0E1,log(K_FeOH3)/log(1.0E1)-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1)-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4 = pow(1.0E1,log(K_FeOH4)/log(1.0E1)-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8 = pow(1.0E1,log(K_FeS2O8)/log(1.0E1)-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2 = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1)-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4 = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1)-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc = pow(1.0E1,log(K_FeAc)/log(1.0E1)-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2 = pow(1.0E1,log(K_FeAc_2)/log(1.0E1)-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2 = pow(1.0E1,log(K_FeBu_2)/log(1.0E1)-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

```

```

KT_FeHCO3 = pow(1.0E1,log(K_FeHCO3)/log(1.0E1)-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS = pow(1.0E1,log(K_FeHS)/log(1.0E1)-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3 = pow(1.0E1,log(K_FeNH3)/log(1.0E1)-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH = pow(1.0E1,log(K_FeOH)/log(1.0E1)-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2 = pow(1.0E1,log(K_FeOH_2)/log(1.0E1)-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr = pow(1.0E1,log(K_FePr)/log(1.0E1)-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1)-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4 = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeVa_2 = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2S_aq = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

for (i = 0; i < 15; i++) {
    if (x[i] < 1e-15)
        xtemp[i] = 0.0;
    else
        xtemp[i] = x[i];
}

S_act_H = xtemp[0]; // Activity of hydrogen ion
S_act_NH4 = xtemp[1]; // Activity of ammonium ion
S_act_Ac = xtemp[2]; // Activity of acetate ion
S_act_Pr = xtemp[3]; // Activity of propionate ion

```

```

S_act_CO3 = xtemp[4]; // Activity of carbonate ion
S_act_SO4 = xtemp[5]; // Activity of sulfate ion
S_act_Fe2 = xtemp[6];
S_act_Fe3 = xtemp[7];
S_act_HS = xtemp[8];
S_act_Bu = xtemp[9];
S_act_Va = xtemp[10];
S_act_NO3 = xtemp[11];
S_act_NO2 = xtemp[12];
mu = xtemp[13]; // Ionic strength of the solution
gama = xtemp[14]; // Activity coefficient

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

// output activities

y[0] = S_act_H;
y[1] = S_act_NH4;
y[2] = S_act_Ac;
y[3] = S_act_Pr;
y[4] = S_act_CO3;
y[5] = S_act_SO4;
y[6] = S_act_Fe2;
y[7] = S_act_Fe3;
y[8] = S_act_HS;
y[9] = S_act_Bu;
y[10] = S_act_Va;
y[11] = S_act_NO3;
y[12] = S_act_NO2;
y[13] = mu;
y[14] = gama;

// output concentrations

y[15] = S_act_H/gama; // H
y[16] = S_act_NH4/gama; // NH4
y[17] = S_act_Ac/gama; // Ac
y[18] = S_act_Pr/gama; // Prop
y[19] = S_act_CO3/pow(gama,4); // CO3-2
y[20] = S_act_SO4/pow(gama,4); // SO4-2
y[21] = S_act_Fe2/pow(gama,4); //Fe+2
y[22] = S_act_Fe3/pow(gama,9); //Fe+3
y[23] = S_act_HS/gama; //Fe+3
y[24] = S_act_Bu/gama; //Bu
y[25] = S_act_Va/gama; //Va
y[26] = S_act_NO3/gama; //NO3
y[27] = S_act_NO2/gama; //NO2

// Ph values
y[28] = -log10(S_act_H); // based on activity
y[29] = -log10(S_act_H/gama); // based on concentration

//Species composition

y[30] = (KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3)/gama;
y[31] = KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*S_act_Fe3;
y[32] = (KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2)/gama2;
y[33] = (KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2)/gama2;
y[34] =
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2)
/gama2;

```

```

y[35] = KT_FeOH2_aq*1.0/(S_act_H*S_act_H)*S_act_Fe2;
y[36] = (KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3)/gama;
y[37] = (KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2)/gama;
y[38] = KT_FeOH3_aq*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3;
y[39] = (KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama;
y[40] = (KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3)/gama;
y[41] = (KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3))/gama4;
y[42] = (KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3))/gama5;
y[43] = (KT_FeAc*S_act_Ac*S_act_Fe2)/gama;
y[44] = (KT_FeAc_2*S_act_Ac*S_act_Fe3)/gama2;
y[45] = (KT_FeBu_2*S_act_Fe3*S_act_Bu)/gama2;
y[46] = (KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama;
y[47] = (KT_FeHS*S_act_HS*S_act_Fe2)/gama;
y[48] = (KT_FeNH3*S_act_NH4*S_act_Fe2)/(S_act_H*gama2);
y[49] = (KT_FeOH*S_act_Fe2)/(S_act_H*gama);
y[50] = (KT_FeOH_2*S_act_Fe3)/(S_act_H*gama2);
y[51] = (KT_FePr*S_act_Fe3*S_act_Pr)/gama2;
y[52] = KT_FeSO4_aq*S_act_SO4*S_act_Fe2;
y[53] = (KT_FeSO4*S_act_SO4*S_act_Fe3)/gama;
y[54] = (KT_FeVa_2*S_act_Fe3*S_act_Va)/gama2;
y[55] = KT_H2CO3_aq*(S_act_H*S_act_H)*S_act_CO3;
y[56] = KT_H2S_aq*S_act_H*S_act_HS;
y[57] = KT_HAc_aq*S_act_H*S_act_Ac;
y[58] = KT_HBu_aq*S_act_H*S_act_Bu;
y[59] = (KT_HCO3*S_act_H*S_act_CO3)/gama;
y[60] = KT_HPr_aq*S_act_H*S_act_Pr;
y[61] = (KT_HSO4*S_act_H*S_act_SO4)/gama;
y[62] = KT_HVa_aq*S_act_H*S_act_Va;
y[63] = (KT_NH3_aq*S_act_NH4)/S_act_H;
y[64] = (KT_NH4SO4*S_act_NH4*S_act_SO4)/gama;
y[65] = KT_H2O/(S_act_H*gama);
y[66] = (KT_S*S_act_HS)/(S_act_H*gama2);
y[67] = 0.0;
y[68] = KT_FeNO2_2*S_act_Fe3*pow((S_act_NO2),2)/gama;
y[69] = KT_FeNO2_aq*S_act_Fe3*pow((S_act_NO2),3);
y[70] = KT_FeNO2*S_act_Fe3*S_act_NO2/gama2;
y[71] = KT_HNO2_aq*S_act_H*S_act_NO2;

```

```

}

```

```

void MatDiv(int dim, double *A, double *B, double *Bout) //nlhs, mxArray *plhs[], int nrhs, const mxArray *prhs[]
{

```

```

    mwSignedIndex m,n,p;    /* matrix dimensions */
    mwSignedIndex *iPivot;  /* inputs to DGESV */
    mxArray *mxPivot;
    mwSignedIndex info, dims[2];
    unsigned int i;
    double *Awork=malloc(dim*dim*sizeof(*Awork)); //not allowed to use new

```

```

    /* dimensions of input matrices */

```

```

    m = (mwSignedIndex)dim;
    p = (mwSignedIndex)dim;
    n = (mwSignedIndex)1;

```

```

    /* DGESV works in-place, so we copy the inputs first. */

```

```

    for (i=0;i<dim;i++) Bout[i]=B[i];
    for (i=0;i<(dim*dim);i++) Awork[i]=A[i];

```

```

    /* Create pivot for DGESV */

```

```

    dims[0] = m;
    dims[1] = p;

```

```

mxPivot = mxCreateNumericArray(2, dims, mxINT32_CLASS, mxREAL);
iPivot = (mwSignedIndex*)mxGetData(mxPivot);

/* Call LAPACK */
dgesv(&m,&n,Awork,&m,iPivot,Bout,&p,&info);
/* Bout now holds X */
//todo recover status from info

mxDestroyArray(mxPivot);
free(Awork);
}

bool FunEv(const double x[], unsigned int neq, double Equ[])
{
    SimStruct *S;
    double *u;

    double K_FeAc2,K_FeAc3_aq,K_FeN2H6,K_FeN3H9,K_FeN4H12,K_FeOH2_aq,K_FeOH2;
    double
    K_FeOH3,K_FeOH3_aq,K_FeOH4,K_FeS2O8,K_Fe2O2H2,K_Fe3O4H4,K_FeAc,K_FeAc_2,K_FeBu_2,K_FeHCO3,K_
    FeHS,K_FeNH3,K_FeOH,K_FeOH_2,K_FePr,K_FeSO4_aq,K_FeSO4,K_FeVa_2,K_H2CO3_aq,K_H2S_aq,K_HAc_aq,
    K_HBu_aq,K_HCO3,K_HPr_aq,K_HSO4,K_HVa_aq;
    double K_NH3_aq,K_NH4SO4,K_H2O,K_S,K_CO2;
    double K_FeNO2_2,K_FeNO2_aq,K_FeNO2,K_HNO2_aq;

    double DH_FeAc2,DH_FeAc3_aq,DH_FeN2H6,DH_FeN3H9,DH_FeN4H12,DH_FeOH2_aq,DH_FeOH2;
    double
    DH_FeOH3,DH_FeOH3_aq,DH_FeOH4,DH_FeS2O8,DH_Fe2O2H2,DH_Fe3O4H4,DH_FeAc,DH_FeAc_2,DH_FeBu_2,
    DH_FeHCO3,DH_FeHS,DH_FeNH3,DH_FeOH,DH_FeOH_2,DH_FePr,DH_FeSO4_aq,DH_FeSO4,DH_FeVa_2,DH_H2
    CO3_aq,DH_H2S_aq,DH_HAc_aq,DH_HBu_aq,DH_HCO3,DH_HPr_aq,DH_HSO4,DH_HVa_aq;
    double DH_NH3_aq,DH_NH4SO4,DH_H2O,DH_S,DH_CO2;
    double DH_FeNO2_2,DH_FeNO2_aq,DH_FeNO2,DH_HNO2_aq;

    double KT_FeAc2,KT_FeAc3_aq,KT_FeN2H6,KT_FeN3H9,KT_FeN4H12,KT_FeOH2_aq,KT_FeOH2;
    double
    KT_FeOH3,KT_FeOH3_aq,KT_FeOH4,KT_FeS2O8,KT_Fe2O2H2,KT_Fe3O4H4,KT_FeAc,KT_FeAc_2,KT_FeBu_2,KT
    _FeHCO3,KT_FeHS,KT_FeNH3,KT_FeOH,KT_FeOH_2,KT_FePr,KT_FeSO4_aq,KT_FeSO4,KT_FeVa_2,KT_H2CO3_
    aq,KT_H2S_aq,KT_HAc_aq,KT_HBu_aq,KT_HCO3,KT_HPr_aq,KT_HSO4,KT_HVa_aq;
    double KT_NH3_aq,KT_NH4SO4,KT_H2O,KT_S,KT_CO2;
    double KT_FeNO2_2,KT_FeNO2_aq,KT_FeNO2,KT_HNO2_aq;

    double R, Temp_K, Temp_C;

    double d, eps, A;

    double S_act_H, S_act_NH4, S_act_Ac, S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
    S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;
    double Tot_H, Tot_NH4, Tot_Ac, Tot_Pr, Tot_CO3, Tot_SO4, Tot_Fe2, Tot_Fe3, Tot_HS, Tot_Bu, Tot_Va, Tot_NO3,
    Tot_NO2;
    double gama2, gama3, gama4, gama5;

    S=Smain;
    u=uain;

    K_FeAc2   = mxGetPr(CHEM_PAR)[0];
    K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];
    K_FeN2H6   = mxGetPr(CHEM_PAR)[2];
    K_FeN3H9   = mxGetPr(CHEM_PAR)[3];
    K_FeN4H12  = mxGetPr(CHEM_PAR)[4];
    K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];
    K_FeOH2    = mxGetPr(CHEM_PAR)[6];
    K_FeOH3    = mxGetPr(CHEM_PAR)[7];
    K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];

```

```

K_FeOH4   = mxGetPr(CHEM_PAR)[9];
K_FeS2O8  = mxGetPr(CHEM_PAR)[10];
K_Fe2O2H2 = mxGetPr(CHEM_PAR)[11];
K_Fe3O4H4 = mxGetPr(CHEM_PAR)[12];
K_FeAc    = mxGetPr(CHEM_PAR)[13];
K_FeAc_2  = mxGetPr(CHEM_PAR)[14];
K_FeBu_2  = mxGetPr(CHEM_PAR)[15];
K_FeHCO3  = mxGetPr(CHEM_PAR)[16];
K_FeHS    = mxGetPr(CHEM_PAR)[17];
K_FeNH3   = mxGetPr(CHEM_PAR)[18];
K_FeOH    = mxGetPr(CHEM_PAR)[19];
K_FeOH_2  = mxGetPr(CHEM_PAR)[20];
K_FePr    = mxGetPr(CHEM_PAR)[21];
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];
K_FeSO4    = mxGetPr(CHEM_PAR)[23];
K_FeVa_2   = mxGetPr(CHEM_PAR)[24];
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];
K_H2S_aq   = mxGetPr(CHEM_PAR)[26];
K_HAc_aq   = mxGetPr(CHEM_PAR)[27];
K_HBu_aq   = mxGetPr(CHEM_PAR)[28];
K_HCO3     = mxGetPr(CHEM_PAR)[29];
K_HPr_aq   = mxGetPr(CHEM_PAR)[30];
K_HSO4     = mxGetPr(CHEM_PAR)[31];
K_HVa_aq   = mxGetPr(CHEM_PAR)[32];
K_NH3_aq   = mxGetPr(CHEM_PAR)[33];
K_NH4SO4   = mxGetPr(CHEM_PAR)[34];
K_H2O      = mxGetPr(CHEM_PAR)[35];
K_S        = mxGetPr(CHEM_PAR)[36];
K_CO2      = mxGetPr(CHEM_PAR)[37];
K_FeNO2_2  = mxGetPr(CHEM_PAR)[38];
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];
K_FeNO2    = mxGetPr(CHEM_PAR)[40];
K_HNO2_aq  = mxGetPr(CHEM_PAR)[41];

DH_FeAc2   = mxGetPr(CHEM_PAR1)[0];
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];
DH_FeN2H6  = mxGetPr(CHEM_PAR1)[2];
DH_FeN3H9  = mxGetPr(CHEM_PAR1)[3];
DH_FeN4H12 = mxGetPr(CHEM_PAR1)[4];
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];
DH_FeOH2    = mxGetPr(CHEM_PAR1)[6];
DH_FeOH3    = mxGetPr(CHEM_PAR1)[7];
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];
DH_FeOH4    = mxGetPr(CHEM_PAR1)[9];
DH_FeS2O8   = mxGetPr(CHEM_PAR1)[10];
DH_Fe2O2H2  = mxGetPr(CHEM_PAR1)[11];
DH_Fe3O4H4  = mxGetPr(CHEM_PAR1)[12];
DH_FeAc     = mxGetPr(CHEM_PAR1)[13];
DH_FeAc_2   = mxGetPr(CHEM_PAR1)[14];
DH_FeBu_2   = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3   = mxGetPr(CHEM_PAR1)[16];
DH_FeHS     = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3    = mxGetPr(CHEM_PAR1)[18];
DH_FeOH     = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2   = mxGetPr(CHEM_PAR1)[20];
DH_FePr     = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4    = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2   = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq   = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq   = mxGetPr(CHEM_PAR1)[27];
DH_HBu_aq   = mxGetPr(CHEM_PAR1)[28];
DH_HCO3     = mxGetPr(CHEM_PAR1)[29];

```

```

DH_HPr_aq = mxGetPr(CHEM_PAR1)[30];
DH_HSO4 = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq = mxGetPr(CHEM_PAR1)[33];
DH_NH4SO4 = mxGetPr(CHEM_PAR1)[34];
DH_H2O = mxGetPr(CHEM_PAR1)[35];
DH_S = mxGetPr(CHEM_PAR1)[36];
DH_CO2 = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2 = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq = mxGetPr(CHEM_PAR1)[41];

R = 8.314; // J/mole/K
Temp_C = 25.0; // 15.0; Why is the temp 15 and not 35
Temp_K = Temp_C + 273.15;

KT_FeAc2 = pow(1.0E1,log(K_FeAc2)/log(1.0E1)-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1)-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6 = pow(1.0E1,log(K_FeN2H6)/log(1.0E1)-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9 = pow(1.0E1,log(K_FeN3H9)/log(1.0E1)-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12 = pow(1.0E1,log(K_FeN4H12)/log(1.0E1)-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1)-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2 = pow(1.0E1,log(K_FeOH2)/log(1.0E1)-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3 = pow(1.0E1,log(K_FeOH3)/log(1.0E1)-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1)-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4 = pow(1.0E1,log(K_FeOH4)/log(1.0E1)-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8 = pow(1.0E1,log(K_FeS2O8)/log(1.0E1)-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2 = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1)-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4 = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1)-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc = pow(1.0E1,log(K_FeAc)/log(1.0E1)-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2 = pow(1.0E1,log(K_FeAc_2)/log(1.0E1)-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2 = pow(1.0E1,log(K_FeBu_2)/log(1.0E1)-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHCO3 = pow(1.0E1,log(K_FeHCO3)/log(1.0E1)-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS = pow(1.0E1,log(K_FeHS)/log(1.0E1)-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3 = pow(1.0E1,log(K_FeNH3)/log(1.0E1)-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH = pow(1.0E1,log(K_FeOH)/log(1.0E1)-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2 = pow(1.0E1,log(K_FeOH_2)/log(1.0E1)-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr = pow(1.0E1,log(K_FePr)/log(1.0E1)-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1)-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4 = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-

```

```

3)*4.342162396873643E-1)/R);
KT_FeVa_2 = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2S_aq = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

```

```

d = exp((-3.743E2)/Temp_C)*1.1445E-2-((Temp_C+2.889414E2)*pow(Temp_C-
3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0;
eps = Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-5.200087E4/Temp_K+2.727586E3;
A = 1.0/pow(Temp_K*(Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-
5.200087E4/Temp_K+2.727586E3),3.0/2.0)*sqrt(exp((-3.743E2)/Temp_C)*1.1445E-2-
((Temp_C+2.889414E2)*pow(Temp_C-3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0)*1.82483E6;

```

```

Tot_NH4 = u[10];
Tot_Ac = u[6]/64.0;
Tot_Pr = u[5]/112.0;
Tot_CO3 = u[9];
Tot_SO4 = u[26];
Tot_Fe2 = u[33];
Tot_Fe3 = u[34];
Tot_HS = u[27]/64.0;
Tot_Bu = u[4]/160.0;
Tot_Va = u[3]/208.0;
Tot_NO3 = 0.0;
Tot_NO2 = 0.0;

```

```

S_act_H = x[0]; // Activity of hydrogen ion
S_act_NH4 = x[1]; // Activity of ammonium ion

```

```

S_act_Ac = x[2];    // Activity of acetate ion
S_act_Pr = x[3];    // Activity of propionate ion
S_act_CO3 = x[4];   // Activity of carbonate ion
S_act_SO4 = x[5];   // Activity of sulfate ion
S_act_Fe2 = x[6];
S_act_Fe3 = x[7];
S_act_HS = x[8];
S_act_Bu = x[9];
S_act_Va = x[10];
S_act_NO3 = x[11];
S_act_NO2 = x[12];
mu      = x[13];    // Ionic strength of the solution
gama    = x[14];    // Activity coefficient

```

```

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

```

```

Equ[0] = S_act_H/gama-(S_act_CO3*2.0)/gama2+S_act_NH4/gama-S_act_HS/gama-S_act_NO2/gama-
S_act_NO3/gama-S_act_Ac/gama-(S_act_SO4*2.0)/gama2+(S_act_Fe2*2.0)/gama2+(S_act_Fe3*3.0)/gama3-
S_act_Bu/gama-S_act_Va/gama-S_act_Pr/gama-KT_H2O/(S_act_H*gama)-(KT_HCO3*S_act_H*S_act_CO3)/gama-
(KT_HSO4*S_act_H*S_act_SO4)/gama+(KT_FeHS*S_act_HS*S_act_Fe2)/gama+(KT_FeNO2*S_act_NO2*S_act_Fe3*
2.0)/gama2+(KT_FeAc*S_act_Ac*S_act_Fe2)/gama-
(KT_NH4SO4*S_act_NH4*S_act_SO4)/gama+(KT_FeSO4*S_act_SO4*S_act_Fe3)/gama+(KT_FePr*S_act_Fe3*S_act_
Pr*2.0)/gama2+(KT_FeAc_2*S_act_Ac*S_act_Fe3*2.0)/gama2+(KT_FeBu_2*S_act_Fe3*S_act_Bu*2.0)/gama2+(KT_Fe
Va_2*S_act_Fe3*S_act_Va*2.0)/gama2-
(KT_S*S_act_HS*2.0)/(S_act_H*gama2)+(KT_FeOH*S_act_Fe2)/(S_act_H*gama)+(KT_FeOH2*1.0/(S_act_H*S_act_H)
*S_act_Fe3)/gama-(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama+(KT_FeOH_2*S_act_Fe3*2.0)/(S_act_H*gama
2)+(KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3)/gama-
(KT_FeSO4*(S_act_SO4*S_act_SO4)*S_act_Fe3)/gama+(KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3)/gama+
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*4.0)/gama4+(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_a
ct_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*5.0)/gama5+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_a
ct_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*
S_act_Fe2*2.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*
S_act_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama+(KT_FeNH3*S_act_NH4*S
_act_Fe2*2.0)/(S_act_H*gama2);

```

```

Equ[1] = -
Tot_NH4+S_act_NH4/gama+(KT_NH3_aq*S_act_NH4)/S_act_H+(KT_NH4SO4*S_act_NH4*S_act_SO4)/gama+(KT_Fe
N2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S
_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*3.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*
S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2+(KT_FeNH3*S_act_NH4*S_act_Fe
2)/(S_act_H*gama2);

```

```

Equ[2] = -
Tot_Ac+S_act_Ac/gama+KT_HAc_aq*S_act_H*S_act_Ac+KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*S_act_Fe3*3.
0+(KT_FeAc*S_act_Ac*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Ac*S_act_Fe3)/gama2+(KT_FeAc2*(S_act_Ac*S_act_Ac
)*S_act_Fe3*2.0)/gama;

```

```

Equ[3] = -Tot_Pr+S_act_Pr/gama+KT_HPr_aq*S_act_H*S_act_Pr+(KT_FePr*S_act_Fe3*S_act_Pr)/gama2;

```

```

Equ[4] = -
Tot_CO3+S_act_CO3/gama2+KT_H2CO3_aq*(S_act_H*S_act_H)*S_act_CO3+(KT_HCO3*S_act_H*S_act_CO3)/gama
+(KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama;

```

```

Equ[5] = -

```

$$\text{Tot_Fe2+S_act_Fe2/gama2+KT_FeSO4_aq*S_act_SO4*S_act_Fe2+KT_FeOH2_aq*1.0/(S_act_H*S_act_H)*S_act_Fe2} \\ +(\text{KT_FeHS*S_act_HS*S_act_Fe2)/gama}+(\text{KT_FeAc*S_act_Ac*S_act_Fe2)/gama}+(\text{KT_FeOH*S_act_Fe2)/(S_act_H*gama)} \\ +(\text{KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2)/gama}+(\text{KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)} \\ *S_act_Fe2)/gama2+(\text{KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)} \\ *S_act_Fe2)/gama2+(\text{KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)} \\ *S_act_Fe2)/gama2+(\text{KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama}+(\text{KT_FeNH3*S_act_NH4*S_act_Fe2)/(S_act_H*gama2});$$
$$\text{Equ}[8] = -\frac{\text{Tot_HS} + \text{S_act_HS}/\gamma + K_T \text{H}_2\text{S}_{\text{aq}} \text{S_act_H}^+ \text{S_act_HS} + (K_T \text{FeHS} \text{S_act_HS} \text{S_act_Fe}^{2+})/\gamma + (K_T \text{S} \text{S_act_HS})/(\text{S_act_H}^+ \gamma^2)}{}$$
$$\text{Equ}[10] = -\text{Tot_Va} + S_act_Va / \gamma + K T_H V a_{aq} S_act_H S_act_Va + (K T_Fe V a_2 S_act_Fe^3 S_act_Va) / \gamma_2;$$
$$\text{Equ}[12] = - \frac{\text{Tot_NO2} + \text{S_act_NO2} / \text{gama} + \text{KT_HNO2_aq} * \text{S_act_H} * \text{S_act_NO2} + \text{KT_FeNO2_aq} * (\text{S_act_NO2} * \text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3}^{3.0} + (\text{KT_FeNO2} * \text{S_act_NO2} * \text{S_act_Fe3}) / \text{gama}^2 + (\text{KT_FeNO2_2} * (\text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3}^{2.0}) / \text{gama}}{}$$

129

```

ct_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeHC
O3*S_act_H*S_act_CO3*S_act_Fe2*(1.0/2.0))/gama+(KT_FeNH3*S_act_NH4*S_act_Fe2*2.0)/(S_act_H*gama2);

```

```

Equ[14] = -gama+pow(1.0E1,A*(mu*(3.0/1.0E1)-sqrt(mu)/(sqrt(mu)+1.0)));

```

```

    return 1;
}

```

```

bool JacAna(const double x[], double *jac)
{

```

```

    SimStruct *S;
    double *u;

```

```

    double K_FeAc2,K_FeAc3_aq,K_FeN2H6,K_FeN3H9,K_FeN4H12,K_FeOH2_aq,K_FeOH2;
    double

```

```

K_FeOH3,K_FeOH3_aq,K_FeOH4,K_FeS2O8,K_Fe2O2H2,K_Fe3O4H4,K_FeAc,K_FeAc_2,K_FeBu_2,K_FeHCO3,K_
FeHS,K_FeNH3,K_FeOH,K_FeOH_2,K_FePr,K_FeSO4_aq,K_FeSO4,K_FeVa_2,K_H2CO3_aq,K_H2S_aq,K_HAc_aq,
K_HBu_aq,K_HCO3,K_HPr_aq,K_HSO4,K_HVa_aq;
    double K_NH3_aq,K_NH4SO4,K_H2O,K_S,K_CO2;
    double K_FeNO2_2,K_FeNO2_aq,K_FeNO2,K_HNO2_aq;

```

```

    double DH_FeAc2,DH_FeAc3_aq,DH_FeN2H6,DH_FeN3H9,DH_FeN4H12,DH_FeOH2_aq,DH_FeOH2;
    double

```

```

DH_FeOH3,DH_FeOH3_aq,DH_FeOH4,DH_FeS2O8,DH_Fe2O2H2,DH_Fe3O4H4,DH_FeAc,DH_FeAc_2,DH_FeBu_2,
DH_FeHCO3,DH_FeHS,DH_FeNH3,DH_FeOH,DH_FeOH_2,DH_FePr,DH_FeSO4_aq,DH_FeSO4,DH_FeVa_2,DH_H2
CO3_aq,DH_H2S_aq,DH_HAc_aq,DH_HBu_aq,DH_HCO3,DH_HPr_aq,DH_HSO4,DH_HVa_aq;
    double DH_NH3_aq,DH_NH4SO4,DH_H2O,DH_S,DH_CO2;
    double DH_FeNO2_2,DH_FeNO2_aq,DH_FeNO2,DH_HNO2_aq;

```

```

    double KT_FeAc2,KT_FeAc3_aq,KT_FeN2H6,KT_FeN3H9,KT_FeN4H12,KT_FeOH2_aq,KT_FeOH2;
    double

```

```

KT_FeOH3,KT_FeOH3_aq,KT_FeOH4,KT_FeS2O8,KT_Fe2O2H2,KT_Fe3O4H4,KT_FeAc,KT_FeAc_2,KT_FeBu_2,KT
_FeHCO3,KT_FeHS,KT_FeNH3,KT_FeOH,KT_FeOH_2,KT_FePr,KT_FeSO4_aq,KT_FeSO4,KT_FeVa_2,KT_H2CO3_
aq,KT_H2S_aq,KT_HAc_aq,KT_HBu_aq,KT_HCO3,KT_HPr_aq,KT_HSO4,KT_HVa_aq;
    double KT_NH3_aq,KT_NH4SO4,KT_H2O,KT_S,KT_CO2;
    double KT_FeNO2_2,KT_FeNO2_aq,KT_FeNO2,KT_HNO2_aq;

```

```

    double R, Temp_K, Temp_C;

```

```

    double d, eps, A;

```

```

    double S_act_H, S_act_NH4,S_act_Ac,S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;

```

```

    double Tot_H,Tot_NH4, Tot_Ac, Tot_Pr, Tot_CO3, Tot_SO4, Tot_Fe2, Tot_Fe3, Tot_HS, Tot_Bu, Tot_Va, Tot_NO3,
Tot_NO2;

```

```

    double gama2, gama3, gama4, gama5;

```

```

    double gradEqu[231]; /* This will need to change */

```

```

    unsigned int i,j;
    bool status = 1;

```

```

    S=Smain;
    u=uemain;

```

```

K_FeAc2   = mxGetPr(CHEM_PAR)[0];
K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];
K_FeN2H6   = mxGetPr(CHEM_PAR)[2];
K_FeN3H9   = mxGetPr(CHEM_PAR)[3];
K_FeN4H12  = mxGetPr(CHEM_PAR)[4];
K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];
K_FeOH2    = mxGetPr(CHEM_PAR)[6];

```

```

K_FeOH3   = mxGetPr(CHEM_PAR)[7];
K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];
K_FeOH4   = mxGetPr(CHEM_PAR)[9];
K_FeS2O8  = mxGetPr(CHEM_PAR)[10];
K_Fe2O2H2 = mxGetPr(CHEM_PAR)[11];
K_Fe3O4H4 = mxGetPr(CHEM_PAR)[12];
K_FeAc    = mxGetPr(CHEM_PAR)[13];
K_FeAc_2  = mxGetPr(CHEM_PAR)[14];
K_FeBu_2  = mxGetPr(CHEM_PAR)[15];
K_FeHCO3  = mxGetPr(CHEM_PAR)[16];
K_FeHS    = mxGetPr(CHEM_PAR)[17];
K_FeNH3   = mxGetPr(CHEM_PAR)[18];
K_FeOH    = mxGetPr(CHEM_PAR)[19];
K_FeOH_2  = mxGetPr(CHEM_PAR)[20];
K_FePr    = mxGetPr(CHEM_PAR)[21];
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];
K_FeSO4    = mxGetPr(CHEM_PAR)[23];
K_FeVa_2  = mxGetPr(CHEM_PAR)[24];
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];
K_H2S_aq   = mxGetPr(CHEM_PAR)[26];
K_HAc_aq   = mxGetPr(CHEM_PAR)[27];
K_HBu_aq   = mxGetPr(CHEM_PAR)[28];
K_HCO3     = mxGetPr(CHEM_PAR)[29];
K_HPr_aq   = mxGetPr(CHEM_PAR)[30];
K_HSO4     = mxGetPr(CHEM_PAR)[31];
K_HVa_aq   = mxGetPr(CHEM_PAR)[32];
K_NH3_aq   = mxGetPr(CHEM_PAR)[33];
K_NH4SO4   = mxGetPr(CHEM_PAR)[34];
K_H2O      = mxGetPr(CHEM_PAR)[35];
K_S        = mxGetPr(CHEM_PAR)[36];
K_CO2      = mxGetPr(CHEM_PAR)[37];
K_FeNO2_2  = mxGetPr(CHEM_PAR)[38];
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];
K_FeNO2    = mxGetPr(CHEM_PAR)[40];
K_HNO2_aq  = mxGetPr(CHEM_PAR)[41];

DH_FeAc2   = mxGetPr(CHEM_PAR1)[0];
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];
DH_FeN2H6   = mxGetPr(CHEM_PAR1)[2];
DH_FeN3H9   = mxGetPr(CHEM_PAR1)[3];
DH_FeN4H12  = mxGetPr(CHEM_PAR1)[4];
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];
DH_FeOH2    = mxGetPr(CHEM_PAR1)[6];
DH_FeOH3    = mxGetPr(CHEM_PAR1)[7];
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];
DH_FeOH4    = mxGetPr(CHEM_PAR1)[9];
DH_FeS2O8   = mxGetPr(CHEM_PAR1)[10];
DH_Fe2O2H2  = mxGetPr(CHEM_PAR1)[11];
DH_Fe3O4H4  = mxGetPr(CHEM_PAR1)[12];
DH_FeAc     = mxGetPr(CHEM_PAR1)[13];
DH_FeAc_2   = mxGetPr(CHEM_PAR1)[14];
DH_FeBu_2   = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3   = mxGetPr(CHEM_PAR1)[16];
DH_FeHS     = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3    = mxGetPr(CHEM_PAR1)[18];
DH_FeOH     = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2   = mxGetPr(CHEM_PAR1)[20];
DH_FePr     = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4    = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2   = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq   = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq   = mxGetPr(CHEM_PAR1)[27];

```

```

DH_HBu_aq = mxGetPr(CHEM_PAR1)[28];
DH_HCO3 = mxGetPr(CHEM_PAR1)[29];
DH_HPr_aq = mxGetPr(CHEM_PAR1)[30];
DH_HSO4 = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq = mxGetPr(CHEM_PAR1)[33];
DH_NH4SO4 = mxGetPr(CHEM_PAR1)[34];
DH_H2O = mxGetPr(CHEM_PAR1)[35];
DH_S = mxGetPr(CHEM_PAR1)[36];
DH_CO2 = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2 = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq = mxGetPr(CHEM_PAR1)[41];

R = 8.314; // J/mole/K
Temp_C = 25.0; // 15.0; Why is the temp 15 and not 35
Temp_K = Temp_C + 273.15;

KT_FeAc2 = pow(1.0E1,log(K_FeAc2)/log(1.0E1))-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1))-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6 = pow(1.0E1,log(K_FeN2H6)/log(1.0E1))-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9 = pow(1.0E1,log(K_FeN3H9)/log(1.0E1))-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12 = pow(1.0E1,log(K_FeN4H12)/log(1.0E1))-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1))-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2 = pow(1.0E1,log(K_FeOH2)/log(1.0E1))-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3 = pow(1.0E1,log(K_FeOH3)/log(1.0E1))-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1))-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4 = pow(1.0E1,log(K_FeOH4)/log(1.0E1))-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8 = pow(1.0E1,log(K_FeS2O8)/log(1.0E1))-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2 = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1))-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4 = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1))-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc = pow(1.0E1,log(K_FeAc)/log(1.0E1))-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2 = pow(1.0E1,log(K_FeAc_2)/log(1.0E1))-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2 = pow(1.0E1,log(K_FeBu_2)/log(1.0E1))-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHCO3 = pow(1.0E1,log(K_FeHCO3)/log(1.0E1))-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS = pow(1.0E1,log(K_FeHS)/log(1.0E1))-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3 = pow(1.0E1,log(K_FeNH3)/log(1.0E1))-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH = pow(1.0E1,log(K_FeOH)/log(1.0E1))-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2 = pow(1.0E1,log(K_FeOH_2)/log(1.0E1))-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr = pow(1.0E1,log(K_FePr)/log(1.0E1))-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1))-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-

```

```

3)*4.342162396873643E-1)/R);
KT_FeSO4 = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeVa_2 = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2S_aq = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

```

```

d = exp((-3.743E2)/Temp_C)*1.1445E-2-((Temp_C+2.889414E2)*pow(Temp_C-
3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0;
eps = Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-5.200087E4/Temp_K+2.727586E3;
A = 1.0/pow(Temp_K*(Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-
5.200087E4/Temp_K+2.727586E3),3.0/2.0)*sqrt(exp((-3.743E2)/Temp_C)*1.1445E-2-
((Temp_C+2.889414E2)*pow(Temp_C-3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0)*1.82483E6;

```

/ Find the order of these numbers*/ /*need to work out what is going on here*/*

```

Tot_NH4 = u[10];
Tot_Ac = u[6]/64.0; /* Why is this dived by 64*/
Tot_Pr = u[5]/112.0; /* Why is this dived by 112*/
Tot_CO3 = u[9];
Tot_SO4 = u[26];
Tot_Fe2 = u[33];
Tot_Fe3 = u[34];
Tot_HS = u[27]/64.0; /* Why is this dived by 64*/
Tot_Bu = u[4]/160.0; /* Why is this dived by 160*/
Tot_Va = u[3]/208.0; /* Why is this dived by 208*/
Tot_NO3 = 0.0;
Tot_NO2 = 0.0;

```

/ Tot_PO4 = u[26] removed */*

```

S_act_H = x[0]; // Activity of hydrogen ion
S_act_NH4 = x[1]; // Activity of ammonium ion
S_act_Ac = x[2]; // Activity of acetate ion
S_act_Pr = x[3]; // Activity of propionate ion
S_act_CO3 = x[4]; // Activity of carbonate ion
S_act_SO4 = x[5]; // Activity of sulfate ion
S_act_Fe2 = x[6];
S_act_Fe3 = x[7];
S_act_HS = x[8];
S_act_Bu = x[9];
S_act_Va = x[10];
S_act_NO3 = x[11];
S_act_NO2 = x[12];
mu = x[13]; // Ionic strength of the solution
gama = x[14]; // Activity coefficient

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

// Eq1: Charge balance

gradEqu[0] = 1.0/gama+(KT_H2O*1.0/(S_act_H*S_act_H))/gama-(KT_HCO3*S_act_CO3)/gama-
(KT_HSO4*S_act_SO4)/gama+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama+(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS*
2.0)/gama2-(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3*2.0)/gama+(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*
S_act_Fe2*3.0)/gama+(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*S_act_H*
S_act_Fe3*4.0)/gama-
(KT_FeOH*2*1.0/(S_act_H*S_act_H)*S_act_Fe3*2.0)/gama2-
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3*S_act_Fe3)*8.0)/gama4-
(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*S_act_Fe3*S_act_Fe3)*2.0E1)/gama5-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*8.0)/gama2-
(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0)/gama2;

gradEqu[1] = 1.0/gama-
(KT_NH4SO4*S_act_SO4)/gama+(KT_FeNH3*S_act_Fe2*2.0)/(S_act_H*gama2)+(KT_FeN3H9*1.0/(S_act_H*S_act_H*
S_act_H)*S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*
S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*8.0)/gama2+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_
Fe2*4.0)/gama2;

gradEqu[2] = -
1.0/gama+(KT_FeAc*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Fe3*2.0)/gama2+(KT_FeAc2*S_act_Ac*S_act_Fe3*2.0)/ga
ma;

gradEqu[3] = -1.0/gama+(KT_FePr*S_act_Fe3*2.0)/gama2;

gradEqu[4] = -2.0/gama2-(KT_HCO3*S_act_H)/gama+(KT_FeHCO3*S_act_H*S_act_Fe2)/gama;

gradEqu[5] = -2.0/gama2-(KT_HSO4*S_act_H)/gama-
(KT_NH4SO4*S_act_NH4)/gama+(KT_FeSO4*S_act_Fe3)/gama-(KT_FeS2O8*S_act_SO4*S_act_Fe3*2.0)/gama;

gradEqu[6] = 2.0/gama2+KT_FeOH/(S_act_H*gama)-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H))/gama+(KT_FeHS*S_act_HS)/gama+(KT_FeAc*S_act_Ac)/gama+(KT_Fe

```



```

gradEqu[24] = 0.0;

gradEqu[25] = 0.0;

gradEqu[26] = (KT_NH4SO4*S_act_NH4)/gama;

gradEqu[27] =
(KT_FeNH3*S_act_NH4)/(S_act_H*gama2)+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*2.0)/gama
2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*3.0)/gama2+(KT_FeN4H12*1.0
/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*4.0)/gama2;

gradEqu[28] = 0.0;

gradEqu[29] = 0.0;

gradEqu[30] = 0.0;

gradEqu[31] = 0.0;

gradEqu[32] = 0.0;

gradEqu[33] = 0.0;

gradEqu[34] = 0.0;

gradEqu[35] = -S_act_NH4*1.0/(gama*gama)-KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama);

// Eq3: Ac
gradEqu[36] = KT_HAc_aq*S_act_Ac;

gradEqu[37] = 0.0;

gradEqu[38] =
1.0/gama+KT_HAc_aq*S_act_H+(KT_FeAc*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Fe3)/gama2+(KT_FeAc2*S_act_Ac*
S_act_Fe3*4.0)/gama;

gradEqu[39] = 0.0;

gradEqu[40] = 0.0;

gradEqu[41] = 0.0;

gradEqu[42] = (KT_FeAc*S_act_Ac)/gama;

```

```
gradEqu[43] =
KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*3.0+(KT_FeAc2*(S_act_Ac*S_act_Ac)*2.0)/gama+(KT_FeAc_2*S_act_A
c)/gama2;
```

```
gradEqu[44] = 0.0;
```

```
gradEqu[45] = 0.0;
```

```
gradEqu[46] = 0.0;
```

```
gradEqu[47] = 0.0;
```

```
gradEqu[48] = 0.0;
```

```
gradEqu[49] = 0.0;
```

```
gradEqu[50] = -S_act_Ac*1.0/(gama*gama)-KT_FeAc*S_act_Ac*S_act_Fe2*1.0/(gama*gama)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)*2.0;
```

```
// Eq4: Pr
```

```
gradEqu[51] = KT_HPr_aq*S_act_Pr;
```

```
gradEqu[52] = 0.0;
```

```
gradEqu[53] = 0.0;
```

```
gradEqu[54] = 1.0/gama+KT_HPr_aq*S_act_H+(KT_FePr*S_act_Fe3)/gama2;
```

```
gradEqu[55] = 0.0;
```

```
gradEqu[56] = 0.0;
```

```
gradEqu[57] = 0.0;
```

```
gradEqu[58] = (KT_FePr*S_act_Pr)/gama2;
```

```
gradEqu[59] = 0.0;
```

```
gradEqu[60] = 0.0;
```

```
gradEqu[61] = 0.0;
```

```

gradEqu[62] = 0.0;

gradEqu[63] = 0.0;

gradEqu[64] = 0.0;

gradEqu[65] = -S_act_Pr*1.0/(gama*gama);

// Eq5: CO3

gradEqu[66] = KT_H2CO3_aq*S_act_H*S_act_CO3*2.0+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama;

gradEqu[67] = 0.0;

gradEqu[68] = 0.0;

gradEqu[69] = 0.0;

gradEqu[70] =
KT_H2CO3_aq*(S_act_H*S_act_H)+1.0/gama2+(KT_HCO3*S_act_H)/gama+(KT_FeHCO3*S_act_H*S_act_Fe2)/gama;

gradEqu[71] = 0.0;

gradEqu[72] = (KT_FeHCO3*S_act_H*S_act_CO3)/gama;

gradEqu[73] = 0.0;

gradEqu[74] = 0.0;

gradEqu[75] = 0.0;

gradEqu[76] = 0.0;

gradEqu[77] = 0.0;

gradEqu[78] = 0.0;

gradEqu[79] = 0.0;

gradEqu[80] = -KT_HCO3*S_act_H*S_act_CO3*1.0/(gama*gama)-
KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama);

// Eq6: SO4

```

```

gradEqu[81] = (KT_HSO4*S_act_SO4)/gama;

gradEqu[82] = (KT_NH4SO4*S_act_SO4)/gama;

gradEqu[83] = 0.0;

gradEqu[84] = 0.0;

gradEqu[85] = 0.0;

gradEqu[86] =
1.0/gama2+KT_FeSO4_aq*S_act_Fe2+(KT_HSO4*S_act_H)/gama+(KT_NH4SO4*S_act_NH4)/gama+(KT_FeSO4*S_a
ct_Fe3)/gama+(KT_FeS2O8*S_act_SO4*S_act_Fe3*4.0)/gama;

gradEqu[87] = KT_FeSO4_aq*S_act_SO4;

gradEqu[88] = (KT_FeS2O8*(S_act_SO4*S_act_SO4)*2.0)/gama+(KT_FeSO4*S_act_SO4)/gama;

gradEqu[89] = 0.0;

gradEqu[90] = 0.0;

gradEqu[91] = 0.0;

gradEqu[92] = 0.0;

gradEqu[93] = 0.0;

gradEqu[94] = 0.0;

gradEqu[95] = -KT_HSO4*S_act_H*S_act_SO4*1.0/(gama*gama)-
KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama)-KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)*2.0;

// Eq7: Fe2+

gradEqu[96] = -
KT_FeOH2_aq*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*2.0+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama-
(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe2*3.0)/gama-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*2.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*3.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S
_act_Fe2*4.0)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2)/gama2;

gradEqu[97] =

```

$$\frac{(KT_FeNH3*S_act_Fe2)}{(S_act_H*gama2)} + \frac{(KT_FeN3H9*1.0)}{(S_act_H*S_act_H*S_act_H)} * \frac{(S_act_NH4*S_act_NH4)*S_act_Fe2*3.0}{gama2} + \frac{(KT_FeN4H12*1.0)}{(S_act_H*S_act_H*S_act_H*S_act_H)} * \frac{(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*4.0}{gama2} + \frac{(KT_FeN2H6*1.0)}{(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0} / gama2;$$

$$gradEqu[98] = (KT_FeAc*S_act_Fe2)/gama;$$

$$gradEqu[99] = 0.0;$$

$$gradEqu[100] = (KT_FeHCO3*S_act_H*S_act_Fe2)/gama;$$

$$gradEqu[101] = KT_FeSO4_aq*S_act_Fe2;$$

$$gradEqu[102] = \frac{KT_FeOH2_aq*1.0}{(S_act_H*S_act_H)} + \frac{1.0}{gama2} + \frac{KT_FeSO4_aq*S_act_SO4 + KT_FeOH}{(S_act_H*gama)} + \frac{(KT_FeOH3*1.0)}{(S_act_H*S_act_H*S_act_H)} / gama + \frac{(KT_FeHS*S_act_HS)}{gama} + \frac{(KT_FeAc*S_act_Ac)}{gama} + \frac{(KT_FeHCO3*S_act_H*S_act_CO3)}{gama} + \frac{(KT_FeNH3*S_act_NH4)}{(S_act_H*gama2)} + \frac{(KT_FeN2H6*1.0)}{(S_act_H*S_act_H)} * \frac{(S_act_NH4*S_act_NH4)}{gama2} + \frac{(KT_FeN3H9*1.0)}{(S_act_H*S_act_H*S_act_H)} * \frac{(S_act_NH4*S_act_NH4*S_act_NH4)}{gama2} + \frac{(KT_FeN4H12*1.0)}{(S_act_H*S_act_H*S_act_H*S_act_H)} * \frac{(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)}{gama2};$$

$$gradEqu[103] = 0.0;$$

$$gradEqu[104] = (KT_FeHS*S_act_Fe2)/gama;$$

$$gradEqu[105] = 0.0;$$

$$gradEqu[106] = 0.0;$$

$$gradEqu[107] = 0.0;$$

$$gradEqu[108] = 0.0;$$

$$gradEqu[109] = 0.0;$$

$$gradEqu[110] = -KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama) - \frac{KT_FeAc*S_act_Ac*S_act_Fe2*1.0}{(gama*gama)} - \frac{(KT_FeOH*S_act_Fe2*1.0)}{(gama*gama)} / S_act_H - \frac{KT_FeOH3*1.0}{(S_act_H*S_act_H*S_act_H)} * S_act_Fe2*1.0/(gama*gama) - \frac{KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0}{(gama*gama)};$$

// Eq8: Fe3+

$$gradEqu[111] = \frac{KT_FeOH3_aq*1.0}{(S_act_H*S_act_H*S_act_H*S_act_H)} * S_act_Fe3*3.0 - \frac{(KT_FeOH2*1.0)}{(S_act_H*S_act_H*S_act_H)} * S_act_Fe3*2.0 / gama - \frac{(KT_FeOH4*1.0)}{(S_act_H*S_act_H*S_act_H*S_act_H)} * S_act_Fe3*4.0 / gama - \frac{(KT_FeOH_2*1.0)}{(S_act_H*S_act_H)} * S_act_Fe3 / gama2 - \frac{(KT_Fe2O2H2*1.0)}{(S_act_H*S_act_H*S_act_H)} * \frac{(S_act_Fe3*S_act_Fe3)*4.0}{gama4} - \frac{(KT_Fe3O4H4*1.0)}{(S_act_H*S_act_H*S_act_H*S_act_H)} * \frac{(S_act_Fe3*S_act_Fe3*S_act_Fe3)*1.2E1}{gama5};$$

```

gradEqu[112] = 0.0;

gradEqu[113] =
KT_FeAc3_aq*(S_act_Ac*S_act_Ac)*S_act_Fe3*3.0+(KT_FeAc_2*S_act_Fe3)/gama2+(KT_FeAc2*S_act_Ac*S_act_Fe
3*2.0)/gama;

gradEqu[114] = (KT_FePr*S_act_Fe3)/gama2;

gradEqu[115] = 0.0;

gradEqu[116] = (KT_FeSO4*S_act_Fe3)/gama+(KT_FeS2O8*S_act_SO4*S_act_Fe3*2.0)/gama;

gradEqu[117] = 0.0;

gradEqu[118] =
KT_FeOH3_aq*1.0/(S_act_H*S_act_H*S_act_H)+KT_FeNO2_aq*(S_act_NO2*S_act_NO2*S_act_NO2)+KT_FeAc3_aq*
(S_act_Ac*S_act_Ac*S_act_Ac)+1.0/gama3+(KT_FeOH2*1.0/(S_act_H*S_act_H))/gama+(KT_FeOH4*1.0/(S_act_H*S_
act_H*S_act_H*S_act_H))/gama+KT_FeOH_2/(S_act_H*gama2)+(KT_FeAc2*(S_act_Ac*S_act_Ac))/gama+(KT_FeS2O
8*(S_act_SO4*S_act_SO4))/gama+(KT_FeNO2_2*(S_act_NO2*S_act_NO2))/gama+(KT_FeNO2*S_act_NO2)/gama2+(
KT_FeSO4*S_act_SO4)/gama+(KT_FePr*S_act_Pr)/gama2+(KT_FeAc_2*S_act_Ac)/gama2+(KT_FeBu_2*S_act_Bu)/g
ama2+(KT_FeVa_2*S_act_Va)/gama2+(KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*S_act_Fe3*4.0)/gama4+(KT_Fe3O4H4*
1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*9.0)/gama5;

gradEqu[119] = 0.0;

gradEqu[120] = (KT_FeBu_2*S_act_Fe3)/gama2;

gradEqu[121] = (KT_FeVa_2*S_act_Fe3)/gama2;

gradEqu[122] = 0.0;

gradEqu[123] =
KT_FeNO2_aq*(S_act_NO2*S_act_NO2)*S_act_Fe3*3.0+(KT_FeNO2*S_act_Fe3)/gama2+(KT_FeNO2_2*S_act_NO2*
S_act_Fe3*2.0)/gama;

gradEqu[124] = 0.0;

gradEqu[125] = -KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)-
KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)-
KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama);

// Eq9:HS

gradEqu[126] = KT_H2S_aq*S_act_HS-(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS)/gama2;

```

```

gradEqu[127] = 0.0;

gradEqu[128] = 0.0;

gradEqu[129] = 0.0;

gradEqu[130] = 0.0;

gradEqu[131] = 0.0;

gradEqu[132] = (KT_FeHS*S_act_HS)/gama;

gradEqu[133] = 0.0;

gradEqu[134] = 1.0/gama+KT_H2S_aq*S_act_H+KT_S/(S_act_H*gama2)+(KT_FeHS*S_act_Fe2)/gama;

gradEqu[135] = 0.0;

gradEqu[136] = 0.0;

gradEqu[137] = 0.0;

gradEqu[138] = 0.0;

gradEqu[139] = 0.0;

gradEqu[140] = -S_act_HS*1.0/(gama*gama)-KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama);

// Eq10:Bu

gradEqu[141] = KT_HBu_aq*S_act_Bu;

gradEqu[142] = 0.0;

gradEqu[143] = 0.0;

gradEqu[144] = 0.0;

gradEqu[145] = 0.0;

gradEqu[146] = 0.0;

gradEqu[147] = 0.0;

```

```

gradEqu[148] = (KT_FeBu_2*S_act_Bu)/gama2;

gradEqu[149] = 0.0;

gradEqu[150] = 1.0/gama+KT_HBu_aq*S_act_H+(KT_FeBu_2*S_act_Fe3)/gama2;

gradEqu[151] = 0.0;

gradEqu[152] = 0.0;

gradEqu[153] = 0.0;

gradEqu[154] = 0.0;

gradEqu[155] = -S_act_Bu*1.0/(gama*gama);

// Eq11:Va

gradEqu[156] = KT_HVa_aq*S_act_Va;

gradEqu[157] = 0.0;

gradEqu[158] = 0.0;

gradEqu[159] = 0.0;

gradEqu[160] = 0.0;

gradEqu[161] = 0.0;

gradEqu[162] = 0.0;

gradEqu[163] = (KT_FeVa_2*S_act_Va)/gama2;

gradEqu[164] = 0.0;

gradEqu[165] = 0.0;

gradEqu[166] = 1.0/gama+KT_HVa_aq*S_act_H+(KT_FeVa_2*S_act_Fe3)/gama2;

gradEqu[167] = 0.0;

```

```

gradEqu[168] = 0.0;

gradEqu[169] = 0.0;

gradEqu[170] = -S_act_Va*1.0/(gama*gama);

// Eq12:NO3

gradEqu[171] = 0.0;

gradEqu[172] = 0.0;

gradEqu[173] = 0.0;

gradEqu[174] = 0.0;

gradEqu[175] = 0.0;

gradEqu[176] = 0.0;

gradEqu[177] = 0.0;

gradEqu[178] = 0.0;

gradEqu[179] = 0.0;

gradEqu[180] = 0.0;

gradEqu[181] = 0.0;

gradEqu[182] = 1.0/gama;

gradEqu[183] = 0.0;

gradEqu[184] = 0.0;

gradEqu[185] = -S_act_NO3*1.0/(gama*gama);

// Eq13:NO2

gradEqu[186] = KT_HNO2_aq*S_act_NO2;

```

```

gradEqu[187] = 0.0;

gradEqu[188] = 0.0;

gradEqu[189] = 0.0;

gradEqu[190] = 0.0;

gradEqu[191] = 0.0;

gradEqu[192] = 0.0;

gradEqu[193] =
KT_FeNO2_aq*(S_act_NO2*S_act_NO2*S_act_NO2)*3.0+(KT_FeNO2_2*(S_act_NO2*S_act_NO2)*2.0)/gama+(KT_Fe
NO2*S_act_NO2)/gama2;

gradEqu[194] = 0.0;

gradEqu[195] = 0.0;

gradEqu[196] = 0.0;

gradEqu[197] = 0.0;

gradEqu[198] =
1.0/gama+KT_HNO2_aq*S_act_H+KT_FeNO2_aq*(S_act_NO2*S_act_NO2)*S_act_Fe3*9.0+(KT_FeNO2*S_act_Fe3)/g
ama2+(KT_FeNO2_2*S_act_NO2*S_act_Fe3*4.0)/gama;

gradEqu[199] = 0.0;

gradEqu[200] = -S_act_NO2*1.0/(gama*gama)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama)*2.0;

// Eq14: mu

gradEqu[201] = (1.0/2.0)/gama-
(KT_H2O*1.0/(S_act_H*S_act_H)*(1.0/2.0))/gama+(KT_HCO3*S_act_CO3*(1.0/2.0))/gama+(KT_HSO4*S_act_SO4*(1.
0/2.0))/gama+(KT_FeHCO3*S_act_CO3*S_act_Fe2*(1.0/2.0))/gama-
(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS*2.0)/gama2-(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2*(1.0/2.0))/gama-
(KT_FeOH2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe2*(3.0/2.0))/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*2.0)/gama-
(KT_FeOH_2*1.0/(S_act_H*S_act_H)*S_act_Fe3*2.0)/gama2-
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*1.6E1)/gama4-
(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3)*5.0E1)/gama5-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S
_act_Fe2*8.0)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0)/gama2;

```

$$\text{gradEqu}[202] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_NH4SO4} * \text{S_act_SO4} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeNH3} * \text{S_act_Fe2} * 2.0)}{(\text{S_act_H} * \text{gama}^2)} + \frac{(\text{KT_FeN3H9} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2} * 6.0}{\text{gama}^2} + \frac{(\text{KT_FeN4H12} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2} * 8.0}{\text{gama}^2} + \frac{(\text{KT_FeN2H6} * 1.0)}{(\text{S_act_H} * \text{S_act_H})} * \text{S_act_NH4} * \text{S_act_Fe2} * 4.0}{\text{gama}^2};$$

$$\text{gradEqu}[203] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeAc} * \text{S_act_Fe2} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeAc_2} * \text{S_act_Fe3} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_FeAc2} * \text{S_act_Ac} * \text{S_act_Fe3})}{\text{gama}};$$

$$\text{gradEqu}[204] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FePr} * \text{S_act_Fe3} * 2.0)}{\text{gama}^2};$$

$$\text{gradEqu}[205] = \frac{2.0}{\text{gama}^2} + \frac{(\text{KT_HCO3} * \text{S_act_H} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_Fe2} * (1.0/2.0))}{\text{gama}};$$

$$\text{gradEqu}[206] = \frac{2.0}{\text{gama}^2} + \frac{(\text{KT_HSO4} * \text{S_act_H} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_NH4SO4} * \text{S_act_NH4} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeSO4} * \text{S_act_Fe3} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeS2O8} * \text{S_act_SO4} * \text{S_act_Fe3})}{\text{gama}};$$

$$\text{gradEqu}[207] = \frac{2.0}{\text{gama}^2} + \frac{(\text{KT_FeOH} * (1.0/2.0))}{(\text{S_act_H} * \text{gama})} + \frac{(\text{KT_FeOH3} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeHS} * \text{S_act_HS} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeAc} * \text{S_act_Ac} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_CO3} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeNH3} * \text{S_act_NH4} * 2.0)}{(\text{S_act_H} * \text{gama}^2)} + \frac{(\text{KT_FeN2H6} * 1.0)}{(\text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_NH4} * \text{S_act_NH4}) * 2.0}{\text{gama}^2} + \frac{(\text{KT_FeN3H9} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * 2.0}{\text{gama}^2} + \frac{(\text{KT_FeN4H12} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * 2.0}{\text{gama}^2};$$

$$\text{gradEqu}[208] = \frac{(9.0/2.0)}{\text{gama}^3} + \frac{(\text{KT_FeOH2} * 1.0)}{(\text{S_act_H} * \text{S_act_H})} * \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeOH4} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeOH_2} * 2.0)}{(\text{S_act_H} * \text{gama}^2)} + \frac{(\text{KT_FeAc2} * (\text{S_act_Ac} * \text{S_act_Ac}) * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeS2O8} * (\text{S_act_SO4} * \text{S_act_SO4}) * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeNO2_2} * (\text{S_act_NO2} * \text{S_act_NO2}) * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FeNO2_2} * \text{S_act_NO2} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_FeSO4} * \text{S_act_SO4} * (1.0/2.0))}{\text{gama}} + \frac{(\text{KT_FePr} * \text{S_act_Pr} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_FeAc_2} * \text{S_act_Ac} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_FeVa_2} * \text{S_act_Va} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_Fe2O2H2} * 1.0)}{(\text{S_act_H} * \text{S_act_H})} * \text{S_act_Fe3} * 1.6\text{E}1}{\text{gama}} + \frac{(\text{KT_Fe3O4H4} * 1.0)}{(\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H})} * \frac{(\text{S_act_Fe3} * \text{S_act_Fe3}) * (7.5\text{E}1/2.0))}{\text{gama}^5};$$

$$\text{gradEqu}[209] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_S} * 2.0)}{(\text{S_act_H} * \text{gama}^2)} + \frac{(\text{KT_FeHS} * \text{S_act_Fe2} * (1.0/2.0))}{\text{gama}};$$

$$\text{gradEqu}[210] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeBu_2} * \text{S_act_Fe3} * 2.0)}{\text{gama}^2};$$

$$\text{gradEqu}[211] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeVa_2} * \text{S_act_Fe3} * 2.0)}{\text{gama}^2};$$

$$\text{gradEqu}[212] = \frac{(1.0/2.0)}{\text{gama}};$$

$$\text{gradEqu}[213] = \frac{(1.0/2.0)}{\text{gama}} + \frac{(\text{KT_FeNO2} * \text{S_act_Fe3} * 2.0)}{\text{gama}^2} + \frac{(\text{KT_FeNO2_2} * \text{S_act_NO2} * \text{S_act_Fe3})}{\text{gama}};$$

$$\text{gradEqu}[214] = -1.0;$$

$$\text{gradEqu}[215] = \text{S_act_H} * 1.0 / (\text{gama} * \text{gama}) * (-1.0/2.0) - \text{S_act_NH4} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_HS} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_NO2} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_NO3} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Ac} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Bu} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Va} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) -$$

```

S_act_Pr*1.0/(gama*gama)*(1.0/2.0)-(KT_H2O*1.0/(gama*gama)*(1.0/2.0))/S_act_H-
KT_HCO3*S_act_H*S_act_CO3*1.0/(gama*gama)*(1.0/2.0)-
KT_HSO4*S_act_H*S_act_SO4*1.0/(gama*gama)*(1.0/2.0)-
KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0)-
KT_FeAc*S_act_Ac*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0)-
KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama)*(1.0/2.0)-
KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
(KT_FeOH*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0))/S_act_H-
KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0)-
KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0);

```

// Eq15: gamma

```
gradEqu[216] = 0.0;
```

```
gradEqu[217] = 0.0;
```

```
gradEqu[218] = 0.0;
```

```
gradEqu[219] = 0.0;
```

```
gradEqu[220] = 0.0;
```

```
gradEqu[221] = 0.0;
```

```
gradEqu[222] = 0.0;
```

```
gradEqu[223] = 0.0;
```

```
gradEqu[224] = 0.0;
```

```
gradEqu[225] = 0.0;
```

```
gradEqu[226] = 0.0;
```

```
gradEqu[227] = 0.0;
```

```
gradEqu[228] = 0.0;
```

```

gradEqu[229] = pow(1.0E1,A*(mu*(3.0/1.0E1)-
sqrt(mu)/(sqrt(mu)+1.0)))*A*log(1.0E1)*(1.0/pow(sqrt(mu)+1.0,2.0)*(1.0/2.0)-
(1.0/sqrt(mu)*(1.0/2.0))/(sqrt(mu)+1.0)+3.0/1.0E1);

```

```

gradEqu[230] = -1.0;

for (i=0;i<15;i++){
    for (j=0;j<15;j++){
        jac[15*i+j]=gradEqu[i+15*j]; //transpose
    }
}
return status;
}

/*
*calculate initial states
*/
double CalcStaAnneal(double *x)
{
    int i, j, k, n, m, na, accept;
    double fun[15], MeanState[15], MaxState[15], xi[15], xc[15], xinorm[15], xcnorm[15];
    double F, Fc, p1, p50, T, T1, T50, frac, DeltaEavg, DeltaE, p, xrand, RandRange, range;
    bool status;

    //define seed for a random number generator
    srand((unsigned int)time(NULL));

    //random generator period
    RandRange=RAND_MAX;

    //mean values of states
    MeanState[0]=2.781e-9;    //x_H
    MeanState[1]=0.03815;    //x_NH4
    MeanState[2]=0.0056282;   //x_Ac
    MeanState[3]=0.0046166;   //x_Pr
    MeanState[4]=0.0010237;   //x_CO3
    MeanState[5]=0.00026465;  //x_SO4
    MeanState[6]=3.1584e-5;   //x_Fe2
    MeanState[7]=9.3764e-15;  //x_Fe3
    MeanState[8]=0.000106;    //x_HS
    MeanState[9]=0.0003839;   //x_Bu
    MeanState[10]=0.00037056;  //x_Va
    MeanState[11]=0.00079054;  //x_NO3
    MeanState[12]=0.00079472;  //x_NO2
    MeanState[13]=0.10822;    // ionic strength
    MeanState[14]=0.77753;    // activity coefficient

    //maximum values of states divided by mean
    MaxState[0]=5.3969;    //x_H
    MaxState[1]=1.7769;    //x_NH4
    MaxState[2]=1.8509;    //x_Ac
    MaxState[3]=1.8944;    //x_Pr
    MaxState[4]=2.9678;    //x_CO3
    MaxState[5]=1.8566;    //x_SO4
    MaxState[6]=4.3015;    //x_Fe2
    MaxState[7]=11.54;     //x_Fe3
    MaxState[8]=5.4733;    //x_HS
    MaxState[9]=1.9311;    //x_Bu
    MaxState[10]=1.7687;   //x_Va
    MaxState[11]=1.9566;   //x_NO3
    MaxState[12]=1.8227;   //x_NO2
    MaxState[13]=1.3637;   // ionic strength
    MaxState[14]=1.0256;   // activity coefficient

    //range of trial ponts

```

```

range=1e-3;

//Simulated annealing algorithm
//number of cycles
n = 50;
//number of trials per cycle
m = 200;
//number of accepted solutions
na = 0;
// probability of accepting worse solution at the start
p1 = 0.7;
// probability of accepting worse solution at the end
p50 = 0.001;
// initial temperature
T1 = -1/log(p1);
// final temperature
T50 = -1/log(p50);
// fractional reduction every cycle
frac = pow((T50/T1),(1.0/(n-1.0)));

//read initial states
for (i = 0; i < 15; i++) {
    xc[i]=x[i];
    xcnorm[i]=xc[i]/MeanState[i]; //normalize states
}

//initialize counter of accepted solutions
na = na + 1;

//evaluate initial states
F=0;
status=FunEv(xc,15,fun);
for (i = 0; i < 15; i++) {
    F = F+fun[i]*fun[i];
}
Fc=F; //initial cost function

//Current temperature
T = T1;
//DeltaE Average
DeltaEavg = 0.0;

for (j = 0; j < n; j++) {
    for (k = 0; k < m; k++) {
        //Generate new trial points
        for (i = 0; i < 15; i++) {
            xrand=rand();
            xinorm[i]=xcnorm[i]+(xrand/RandRange-0.5)*range;

            //clip to upper and lower bounds
            if (xinorm[i] > MaxState[i]){
                xinorm[i] = MaxState[i];
            }
            if (xinorm[i] < 0){
                xinorm[i] = 0;
            }
        }
    }

    //denormalize states
    for (i = 0; i < 15; i++) {
        xi[i]=xinorm[i]*MeanState[i];
    }
}

```

```

}

//evaluate new states
F=0;
status=FunEv(xi,15,fun);
for (i = 0; i < 15; i++) {
    F = F+fun[i]*fun[i];
}

DeltaE = fabs(F-Fc); ////calculate new DeltaE

if (F>Fc){
    // initialize DeltaE_avg if a worse solution was found
    // on the first iteration
    if ((j==0)&&(k==0)){
        DeltaEavg = DeltaE;
    }

    // objective function is worse
    // generate probability of acceptance
    p = exp(-DeltaE/(DeltaEavg*T));

    // determine whether to accept worse point
    xrand = rand();
    if ((xrand/RAND_MAX) < p){
        //accept the worse solution
        accept = 1;
    }
    else{
        //don't accept the worse solution
        accept = 0;
    }
}
else{
    //objective function is lower, automatically accept
    accept = 1;
}

if (accept==1){
    //update currently accepted solution
    for (i = 0; i < 15; i++) {
        xc[i]=xi[i];
        xcnorm[i]=xc[i]/MeanState[i]; //normalize states
    }
    //calculate the new best cost function
    F=0;
    status=FunEv(xc,15,fun);
    for (i = 0; i < 15; i++) {
        F = F+fun[i]*fun[i];
    }
    Fc=F;
    //increment number of accepted solutions
    na = na + 1;
    //update DeltaEavg
    DeltaEavg = (DeltaEavg*(na-1)+DeltaE)/na;
}
}

//lower the temperature for next cycle
T = frac * T;
}

for (i = 0; i < 15; i++) {
    x[i]=xc[i]; //return the the best states
}

```

```

    return Fc; //return the minimum cost function
}

/*
 * mdlUpdate - perform action at major integration time step
 */
static void mdlUpdate(double *x, double *u, SimStruct *S, int tid)
{
    /*need to fix with malloc*/
    double gres[15], gresold[15], jaca[231], xold[15], xnew[15], deltax[15]; //, testmat[100], testvec[10], testout[10];
    double acc[15];
    double Fmin;
    double TOL;
    int MaxSteps;
    bool status, conv;
    int i,j;

    TOL = 1e-12;
    MaxSteps= 10000;
    j=1;
    //This is function root position
    //set globals so subfunctions can use them
    Smain=S;
    umain=u;

    //transfer x & tol
    for (i = 0; i < 15; i++) xnew[i]=x[i];
    for (i = 0; i < 15; i++) acc[i]=TOL;
    status=FunEv(xnew,15,gres);
    conv=1;
    for (i=0;i<15;i++) if (fabs(gres[i])>acc[i]) conv=0;
    j=0;
    //main loop
    while ((!conv) && (j <= MaxSteps)){
        //Calculate the Jacobian
        status=JacAna(xnew,jaca);
        //Calculate residual
        status=FunEv(xnew,15,gres);
        //Calculate deltax
        MatDiv(15, jaca, gres, deltax);

        //Update x and store old values
        for (i = 0; i < 15; i++){
            xold[i]=xnew[i];
            gresold[i]=gres[i];
            xnew[i]=xold[i]-deltax[i]; //weighting should be changed.
        }

        conv=1;
        for (i=0;i<15;i++) {
            if (fabs(gres[i])>acc[i]) conv=0;
        }
    }

    //check if unstable results are reached
    if (xnew[0] != xnew[0]){ //is xnew[0] NaN
        Fmin=CalcStaAnneal(x); //calculate states by using simulated annealing algorithm
        for (i = 0; i < 15; i++) xnew[i]=x[i];
    }
    else{
        if (conv)
        {
            for (i = 0; i < 15; i++){

```

```

        if (xnew[i] < 0){
            xnew[i] = 0; //set negative states to zero
        }
        x[i]=xnew[i];
    }
}
else status=0;
}
}

/*
 * mdlDerivatives - compute the derivatives
 */
static void mdlDerivatives(double *dx, double *x, double *u, SimStruct *S, int tid)
{
}

/*
 * mdlTerminate - called when the simulation is terminated.
 */
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE /* Is this file being compiled as a MEX-file? */
#include "simulink.c" /* MEX-file interface mechanism */
#else
#include "cg_sfun.h" /* Code generation registration function */
#endif

```

```

/*
 * UQ_precipitation_DAEs model block
 * * C file – S function UQ_speciation2_DAE_synAMD
 * Copyright: Xavier Flores-Alsina, Ulf Jeppsson, IEA, Lund University, Lund, Sweden
 *      Krist V. Gernaey, DTU, Denmark
 *      November 2013
 *
 * Edited: Natalie Greenway
 *      University of the Witwatersrand
 *      July 2017
 */

#define S_FUNCTION_NAME UQ_speciation2_DAE_synAMD
#include "simstruc.h"
#include "mex.h"
#include "lapack.h"
#include <math.h>
#include "matrix.h"

#define CHEM_INIT    ssGetArg(S,0)    /* initial values */
#define CHEM_PAR     ssGetArg(S,1)    /* chemical parameters */
#define CHEM_PAR1    ssGetArg(S,2)    /* chemical parameters */
#define ACTIVITY     ssGetArg(S,3)    /* activity */
#define PRECRX_PAR   ssGetArg(S,4)    /* chemical parameters */

double *umain;
SimStruct *Smain;

/*
 * mdlInitializeSizes - initialize the sizes array
 */

static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumContStates( S, 0); /* number of continuous states */
    ssSetNumDiscStates( S, 15); /* number of discrete states */
    ssSetNumInputs(     S, 49); /* number of inputs */
    ssSetNumOutputs(    S, 73); /* number of outputs */
    ssSetDirectFeedThrough(S, 0); /* direct feed through flag */
    ssSetNumSampleTimes( S, 1); /* number of sample times */
    ssSetNumSFcnParams(  S, 5); /* number of input arguments */
    ssSetNumRWork(       S, 0); /* number of real work vector elements */
    ssSetNumIWork(       S, 0); /* number of integer work vector elements*/
    ssSetNumPWork(       S, 0); /* number of pointer work vector elements*/
}

/*
 * mdlInitializeSampleTimes - initialize the sample times array
 */
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, INHERITED_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0.0);
}

/*
 * mdlInitializeConditions - initialize the states

```

```

*/
static void mdlInitializeConditions(double *x0, SimStruct *S)
{
    int i;

    for (i = 0; i < 15; i++) {
        x0[i] = mxGetPr(CHEM_INIT)[i];
    }
}

/*
 * mdlOutputs - compute the outputs
 */
static void mdlOutputs(double *y, double *x, double *u, SimStruct *S, int tid)
{
    double S_act_H, S_act_NH4, S_act_Ac, S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
    S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;
    double gama2, gama3, gama4, gama5;
    double S_act_S;

    double K_FeAc2, K_FeAc3_aq, K_FeN2H6, K_FeN3H9, K_FeN4H12, K_FeOH2_aq, K_FeOH2;
    double
    K_FeOH3, K_FeOH3_aq, K_FeOH4, K_FeS2O8, K_Fe2O2H2, K_Fe3O4H4, K_FeAc, K_FeAc_2, K_FeBu_2, K_FeHCO3, K_
    FeHS, K_FeNH3, K_FeOH, K_FeOH_2, K_FePr, K_FeSO4_aq, K_FeSO4, K_FeVa_2, K_H2CO3_aq, K_H2S_aq, K_HAc_aq,
    K_HBu_aq, K_HCO3, K_HPr_aq, K_HSO4, K_HVa_aq;
    double K_NH3_aq, K_NH4SO4, K_H2O, K_S, K_CO2;
    double K_FeNO2_2, K_FeNO2_aq, K_FeNO2, K_HNO2_aq;

    double DH_FeAc2, DH_FeAc3_aq, DH_FeN2H6, DH_FeN3H9, DH_FeN4H12, DH_FeOH2_aq, DH_FeOH2;
    double
    DH_FeOH3, DH_FeOH3_aq, DH_FeOH4, DH_FeS2O8, DH_Fe2O2H2, DH_Fe3O4H4, DH_FeAc, DH_FeAc_2, DH_FeBu_2,
    DH_FeHCO3, DH_FeHS, DH_FeNH3, DH_FeOH, DH_FeOH_2, DH_FePr, DH_FeSO4_aq, DH_FeSO4, DH_FeVa_2, DH_H2
    CO3_aq, DH_H2S_aq, DH_HAc_aq, DH_HBu_aq, DH_HCO3, DH_HPr_aq, DH_HSO4, DH_HVa_aq;
    double DH_NH3_aq, DH_NH4SO4, DH_H2O, DH_S, DH_CO2;
    double DH_FeNO2_2, DH_FeNO2_aq, DH_FeNO2, DH_HNO2_aq;

    double KT_FeAc2, KT_FeAc3_aq, KT_FeN2H6, KT_FeN3H9, KT_FeN4H12, KT_FeOH2_aq, KT_FeOH2;
    double
    KT_FeOH3, KT_FeOH3_aq, KT_FeOH4, KT_FeS2O8, KT_Fe2O2H2, KT_Fe3O4H4, KT_FeAc, KT_FeAc_2, KT_FeBu_2, KT
    _FeHCO3, KT_FeHS, KT_FeNH3, KT_FeOH, KT_FeOH_2, KT_FePr, KT_FeSO4_aq, KT_FeSO4, KT_FeVa_2, KT_H2CO3_
    aq, KT_H2S_aq, KT_HAc_aq, KT_HBu_aq, KT_HCO3, KT_HPr_aq, KT_HSO4, KT_HVa_aq;
    double KT_NH3_aq, KT_NH4SO4, KT_H2O, KT_S, KT_CO2;
    double KT_FeNO2_2, KT_FeNO2_aq, KT_FeNO2, KT_HNO2_aq;

    double KT_FeSs;

    double SI_fes;

    double R, Temp_K, Temp_C;

    double xtemp[15];
    int i;

    K_FeAc2   = mxGetPr(CHEM_PAR)[0];
    K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];
    K_FeN2H6   = mxGetPr(CHEM_PAR)[2];
    K_FeN3H9   = mxGetPr(CHEM_PAR)[3];
    K_FeN4H12  = mxGetPr(CHEM_PAR)[4];
    K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];
    K_FeOH2    = mxGetPr(CHEM_PAR)[6];
    K_FeOH3    = mxGetPr(CHEM_PAR)[7];

```

```

K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];
K_FeOH4    = mxGetPr(CHEM_PAR)[9];
K_FeS2O8   = mxGetPr(CHEM_PAR)[10];
K_Fe2O2H2  = mxGetPr(CHEM_PAR)[11];
K_Fe3O4H4  = mxGetPr(CHEM_PAR)[12];
K_FeAc     = mxGetPr(CHEM_PAR)[13];
K_FeAc_2   = mxGetPr(CHEM_PAR)[14];
K_FeBu_2   = mxGetPr(CHEM_PAR)[15];
K_FeHCO3   = mxGetPr(CHEM_PAR)[16];
K_FeHS     = mxGetPr(CHEM_PAR)[17];
K_FeNH3    = mxGetPr(CHEM_PAR)[18];
K_FeOH     = mxGetPr(CHEM_PAR)[19];
K_FeOH_2   = mxGetPr(CHEM_PAR)[20];
K_FePr     = mxGetPr(CHEM_PAR)[21];
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];
K_FeSO4    = mxGetPr(CHEM_PAR)[23];
K_FeVa_2   = mxGetPr(CHEM_PAR)[24];
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];
K_H2S_aq   = mxGetPr(CHEM_PAR)[26];
K_HAc_aq   = mxGetPr(CHEM_PAR)[27];
K_HBu_aq   = mxGetPr(CHEM_PAR)[28];
K_HCO3     = mxGetPr(CHEM_PAR)[29];
K_HPr_aq   = mxGetPr(CHEM_PAR)[30];
K_HSO4     = mxGetPr(CHEM_PAR)[31];
K_HVa_aq   = mxGetPr(CHEM_PAR)[32];
K_NH3_aq   = mxGetPr(CHEM_PAR)[33];
K_NH4SO4   = mxGetPr(CHEM_PAR)[34];
K_H2O      = mxGetPr(CHEM_PAR)[35];
K_S        = mxGetPr(CHEM_PAR)[36];
K_CO2      = mxGetPr(CHEM_PAR)[37];
K_FeNO2_2  = mxGetPr(CHEM_PAR)[38];
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];
K_FeNO2    = mxGetPr(CHEM_PAR)[40];
K_HNO2_aq  = mxGetPr(CHEM_PAR)[41];

DH_FeAc2   = mxGetPr(CHEM_PAR1)[0];
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];
DH_FeN2H6  = mxGetPr(CHEM_PAR1)[2];
DH_FeN3H9  = mxGetPr(CHEM_PAR1)[3];
DH_FeN4H12 = mxGetPr(CHEM_PAR1)[4];
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];
DH_FeOH2    = mxGetPr(CHEM_PAR1)[6];
DH_FeOH3    = mxGetPr(CHEM_PAR1)[7];
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];
DH_FeOH4    = mxGetPr(CHEM_PAR1)[9];
DH_FeS2O8   = mxGetPr(CHEM_PAR1)[10];
DH_Fe2O2H2  = mxGetPr(CHEM_PAR1)[11];
DH_Fe3O4H4  = mxGetPr(CHEM_PAR1)[12];
DH_FeAc     = mxGetPr(CHEM_PAR1)[13];
DH_FeAc_2   = mxGetPr(CHEM_PAR1)[14];
DH_FeBu_2   = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3   = mxGetPr(CHEM_PAR1)[16];
DH_FeHS     = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3    = mxGetPr(CHEM_PAR1)[18];
DH_FeOH     = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2   = mxGetPr(CHEM_PAR1)[20];
DH_FePr     = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4    = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2   = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq   = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq   = mxGetPr(CHEM_PAR1)[27];
DH_HBu_aq   = mxGetPr(CHEM_PAR1)[28];

```

```

DH_HCO3   = mxGetPr(CHEM_PAR1)[29];
DH_HPr_aq  = mxGetPr(CHEM_PAR1)[30];
DH_HSO4    = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq  = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq  = mxGetPr(CHEM_PAR1)[33];
DH_NH4SO4  = mxGetPr(CHEM_PAR1)[34];
DH_H2O     = mxGetPr(CHEM_PAR1)[35];
DH_S       = mxGetPr(CHEM_PAR1)[36];
DH_CO2     = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2    = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq  = mxGetPr(CHEM_PAR1)[41];

R   = 8.314;           // J/mole/K
Temp_C = 25.0;         // 15.0
Temp_K = Temp_C + 273.15;

KT_FeAc2   = pow(1.0E1,log(K_FeAc2)/log(1.0E1)-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1)-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6   = pow(1.0E1,log(K_FeN2H6)/log(1.0E1)-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9   = pow(1.0E1,log(K_FeN3H9)/log(1.0E1)-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12  = pow(1.0E1,log(K_FeN4H12)/log(1.0E1)-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1)-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2    = pow(1.0E1,log(K_FeOH2)/log(1.0E1)-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3    = pow(1.0E1,log(K_FeOH3)/log(1.0E1)-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1)-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4    = pow(1.0E1,log(K_FeOH4)/log(1.0E1)-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8   = pow(1.0E1,log(K_FeS2O8)/log(1.0E1)-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2  = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1)-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4  = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1)-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc     = pow(1.0E1,log(K_FeAc)/log(1.0E1)-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2   = pow(1.0E1,log(K_FeAc_2)/log(1.0E1)-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2   = pow(1.0E1,log(K_FeBu_2)/log(1.0E1)-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHCO3   = pow(1.0E1,log(K_FeHCO3)/log(1.0E1)-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS     = pow(1.0E1,log(K_FeHS)/log(1.0E1)-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3    = pow(1.0E1,log(K_FeNH3)/log(1.0E1)-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH     = pow(1.0E1,log(K_FeOH)/log(1.0E1)-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2   = pow(1.0E1,log(K_FeOH_2)/log(1.0E1)-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr     = pow(1.0E1,log(K_FePr)/log(1.0E1)-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1)-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

```

```

KT_FeSO4 = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeVa_2 = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2S_aq = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

KT_FeSs = mxGetPr(PRECRX_PAR)[1];

for (i = 0; i < 15; i++) {
    if (x[i] < 1e-15)
        xtemp[i] = 0.0;
    else
        xtemp[i] = x[i];
}

S_act_H = xtemp[0]; // Activity of hydrogen ion
S_act_NH4 = xtemp[1]; // Activity of ammonium ion
S_act_Ac = xtemp[2]; // Activity of acetate ion
S_act_Pr = xtemp[3]; // Activity of propionate ion
S_act_CO3 = xtemp[4]; // Activity of carbonate ion
S_act_SO4 = xtemp[5]; // Activity of sulfate ion
S_act_Fe2 = xtemp[6];
S_act_Fe3 = xtemp[7];
S_act_HS = xtemp[8];
S_act_Bu = xtemp[9];
S_act_Va = xtemp[10];
S_act_NO3 = xtemp[11];
S_act_NO2 = xtemp[12];
mu = xtemp[13]; // Ionic strength of the solution
gama = xtemp[14]; // Activity coefficient

```

```

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

// output activities

y[0] = S_act_H;
y[1] = S_act_NH4;
y[2] = S_act_Ac;
y[3] = S_act_Pr;
y[4] = S_act_CO3;
y[5] = S_act_SO4;
y[6] = S_act_Fe2;
y[7] = S_act_Fe3;
y[8] = S_act_HS;
y[9] = S_act_Bu;
y[10] = S_act_Va;
y[11] = S_act_NO3;
y[12] = S_act_NO2;
y[13] = mu;
y[14] = gama;

// output concentrations

y[15] = S_act_H/gama; // H
y[16] = S_act_NH4/gama; // NH4
y[17] = S_act_Ac/gama; // Ac
y[18] = S_act_Pr/gama; // Prop
y[19] = S_act_CO3/pow(gama,4); // CO3-2
y[20] = S_act_SO4/pow(gama,4); // SO4-2
y[21] = S_act_Fe2/pow(gama,4); //Fe+2
y[22] = S_act_Fe3/pow(gama,9); //Fe+3
y[23] = S_act_HS/gama; //Fe+3
y[24] = S_act_Bu/gama; //Bu
y[25] = S_act_Va/gama; //Va
y[26] = S_act_NO3/gama; //NO3
y[27] = S_act_NO2/gama; //NO2

// Ph values
y[28] = -log10(S_act_H); // based on activity
y[29] = -log10(S_act_H/gama); // based on concentration

//Species composition

y[30] = (KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3)/gama;
y[31] = (KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*S_act_Fe3);
y[32] = (KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2)/gama2;
y[33] = (KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2)/gama2;
y[34] =
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2)
/gama2;
y[35] = (KT_FeOH2_aq*1.0/(S_act_H*S_act_H)*S_act_Fe2);
y[36] = (KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3)/gama;
y[37] = (KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2)/gama;
y[38] = (KT_FeOH3_aq*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3);
y[39] = (KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama;
y[40] = (KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3)/gama;
y[41] = (KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3))/gama4;
y[42] = (KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3))/gama5;
y[43] = (KT_FeAc*S_act_Ac*S_act_Fe2)/gama;
y[44] = (KT_FeAc_2*S_act_Ac*S_act_Fe3)/gama2;
y[45] = (KT_FeBu_2*S_act_Fe3*S_act_Bu)/gama2;
y[46] = (KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama;

```

```

y[47] = (KT_FeHS*S_act_HS*S_act_Fe2)/gama;
y[48] = (KT_FeNH3*S_act_NH4*S_act_Fe2)/(S_act_H*gama2);
y[49] = (KT_FeOH*S_act_Fe2)/(S_act_H*gama);
y[50] = (KT_FeOH_2*S_act_Fe3)/(S_act_H*gama2);
y[51] = (KT_FePr*S_act_Fe3*S_act_Pr)/gama2;
y[52] = KT_FeSO4_aq*S_act_SO4*S_act_Fe2;
y[53] = (KT_FeSO4*S_act_SO4*S_act_Fe3)/gama;
y[54] = (KT_FeVa_2*S_act_Fe3*S_act_Va)/gama2;
y[55] = KT_H2CO3_aq*(S_act_H*S_act_H)*S_act_CO3;
y[56] = KT_H2S_aq*S_act_H*S_act_HS;
y[57] = KT_HAc_aq*S_act_H*S_act_Ac;
y[58] = KT_HBu_aq*S_act_H*S_act_Bu;
y[59] = (KT_HCO3*S_act_H*S_act_CO3)/gama;
y[60] = KT_HPr_aq*S_act_H*S_act_Pr;
y[61] = (KT_HSO4*S_act_H*S_act_SO4)/gama;
y[62] = KT_HVa_aq*S_act_H*S_act_Va;
y[63] = (KT_NH3_aq*S_act_NH4)/S_act_H;
y[64] = (KT_NH4SO4*S_act_NH4*S_act_SO4)/gama;
y[65] = KT_H2O/(S_act_H*gama);
y[66] = (KT_S*S_act_HS)/(S_act_H*gama2);
y[67] = 0.0;
y[68] = KT_FeNO2_2*S_act_Fe3*pow((S_act_NO2),2)/gama;
y[69] = KT_FeNO2_aq*S_act_Fe3*pow((S_act_NO2),3);
y[70] = KT_FeNO2*S_act_Fe3*S_act_NO2/gama2;
y[71] = KT_HNO2_aq*S_act_H*S_act_NO2;

// Compute the saturation index of potential minerals

S_act_S = (S_act_HS)/(S_act_H);

SI_fes          = log10(S_act_Fe2*S_act_S/KT_FeSs);

y[72] = SI_fes;
}

void MatDiv(int dim, double *A, double *B, double *Bout) //nlhs, mxArray *plhs[], int nrhs, const mxArray *prhs[]
{

    mwSignedIndex m,n,p;    /* matrix dimensions */
    mwSignedIndex *iPivot; /* inputs to DGESV */
    mxArray *mxPivot;
    mwSignedIndex info, dims[2];
    unsigned int i;
    double *Awork=malloc(dim*dim*sizeof(*Awork));//not allowed to use new

    /* dimensions of input matrices */
    m = (mwSignedIndex)dim;
    p = (mwSignedIndex)dim;
    n = (mwSignedIndex)1;

    /* DGESV works in-place, so we copy the inputs first. */
    for (i=0;i<dim;i++) Bout[i]=B[i];
    for (i=0;i<(dim*dim);i++) Awork[i]=A[i];

    /* Create pivot for DGESV */
    dims[0] = m;
    dims[1] = p;
    mxPivot = mxCreateNumericArray(2, dims, mxINT32_CLASS, mxREAL);
    iPivot = (mwSignedIndex*)mxGetData(mxPivot);

    /* Call LAPACK */
    dgesv(&m,&n,Awork,&m,iPivot,Bout,&p,&info);

```

```

/* Bout now holds X */
//todo recover status from info

mxDestroyArray(mxPivot);
free(Awork);
}

bool FunEv(const double x[], unsigned int neq, double Equ[])
{
    SimStruct *S;
    double *u;

    double K_FeAc2,K_FeAc3_aq,K_FeN2H6,K_FeN3H9,K_FeN4H12,K_FeOH2_aq,K_FeOH2;
    double
    K_FeOH3,K_FeOH3_aq,K_FeOH4,K_FeS2O8,K_Fe2O2H2,K_Fe3O4H4,K_FeAc,K_FeAc_2,K_FeBu_2,K_FeHCO3,K_
    FeHS,K_FeNH3,K_FeOH,K_FeOH_2,K_FePr,K_FeSO4_aq,K_FeSO4,K_FeVa_2,K_H2CO3_aq,K_H2S_aq,K_HAc_aq,
    K_HBu_aq,K_HCO3,K_HPr_aq,K_HSO4,K_HVa_aq;
    double K_NH3_aq,K_NH4SO4,K_H2O,K_S,K_CO2;
    double K_FeNO2_2,K_FeNO2_aq,K_FeNO2,K_HNO2_aq;

    double DH_FeAc2,DH_FeAc3_aq,DH_FeN2H6,DH_FeN3H9,DH_FeN4H12,DH_FeOH2_aq,DH_FeOH2;
    double
    DH_FeOH3,DH_FeOH3_aq,DH_FeOH4,DH_FeS2O8,DH_Fe2O2H2,DH_Fe3O4H4,DH_FeAc,DH_FeAc_2,DH_FeBu_2,
    DH_FeHCO3,DH_FeHS,DH_FeNH3,DH_FeOH,DH_FeOH_2,DH_FePr,DH_FeSO4_aq,DH_FeSO4,DH_FeVa_2,DH_H2
    CO3_aq,DH_H2S_aq,DH_HAc_aq,DH_HBu_aq,DH_HCO3,DH_HPr_aq,DH_HSO4,DH_HVa_aq;
    double DH_NH3_aq,DH_NH4SO4,DH_H2O,DH_S,DH_CO2;
    double DH_FeNO2_2,DH_FeNO2_aq,DH_FeNO2,DH_HNO2_aq;

    double KT_FeAc2,KT_FeAc3_aq,KT_FeN2H6,KT_FeN3H9,KT_FeN4H12,KT_FeOH2_aq,KT_FeOH2;
    double
    KT_FeOH3,KT_FeOH3_aq,KT_FeOH4,KT_FeS2O8,KT_Fe2O2H2,KT_Fe3O4H4,KT_FeAc,KT_FeAc_2,KT_FeBu_2,KT
    _FeHCO3,KT_FeHS,KT_FeNH3,KT_FeOH,KT_FeOH_2,KT_FePr,KT_FeSO4_aq,KT_FeSO4,KT_FeVa_2,KT_H2CO3_
    aq,KT_H2S_aq,KT_HAc_aq,KT_HBu_aq,KT_HCO3,KT_HPr_aq,KT_HSO4,KT_HVa_aq;
    double KT_NH3_aq,KT_NH4SO4,KT_H2O,KT_S,KT_CO2;
    double KT_FeNO2_2,KT_FeNO2_aq,KT_FeNO2,KT_HNO2_aq;

    double R, Temp_K, Temp_C;

    double d, eps, A;

    double S_act_H, S_act_NH4, S_act_Ac, S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
    S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;
    double Tot_H,Tot_NH4, Tot_Ac, Tot_Pr, Tot_CO3, Tot_SO4, Tot_Fe2, Tot_Fe3, Tot_HS, Tot_Bu, Tot_Va, Tot_NO3,
    Tot_NO2;
    double gama2, gama3, gama4, gama5;

    S=Smain;
    u=umain;

    K_FeAc2   = mxGetPr(CHEM_PAR)[0];
    K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];
    K_FeN2H6   = mxGetPr(CHEM_PAR)[2];
    K_FeN3H9   = mxGetPr(CHEM_PAR)[3];
    K_FeN4H12  = mxGetPr(CHEM_PAR)[4];
    K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];
    K_FeOH2    = mxGetPr(CHEM_PAR)[6];
    K_FeOH3    = mxGetPr(CHEM_PAR)[7];
    K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];
    K_FeOH4    = mxGetPr(CHEM_PAR)[9];
    K_FeS2O8   = mxGetPr(CHEM_PAR)[10];
    K_Fe2O2H2  = mxGetPr(CHEM_PAR)[11];
    K_Fe3O4H4  = mxGetPr(CHEM_PAR)[12];
    K_FeAc     = mxGetPr(CHEM_PAR)[13];
    K_FeAc_2   = mxGetPr(CHEM_PAR)[14];

```

```

K_FeBu_2 = mxGetPr(CHEM_PAR)[15];
K_FeHCO3 = mxGetPr(CHEM_PAR)[16];
K_FeHS = mxGetPr(CHEM_PAR)[17];
K_FeNH3 = mxGetPr(CHEM_PAR)[18];
K_FeOH = mxGetPr(CHEM_PAR)[19];
K_FeOH_2 = mxGetPr(CHEM_PAR)[20];
K_FePr = mxGetPr(CHEM_PAR)[21];
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];
K_FeSO4 = mxGetPr(CHEM_PAR)[23];
K_FeVa_2 = mxGetPr(CHEM_PAR)[24];
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];
K_H2S_aq = mxGetPr(CHEM_PAR)[26];
K_HAc_aq = mxGetPr(CHEM_PAR)[27];
K_HBu_aq = mxGetPr(CHEM_PAR)[28];
K_HCO3 = mxGetPr(CHEM_PAR)[29];
K_HPr_aq = mxGetPr(CHEM_PAR)[30];
K_HSO4 = mxGetPr(CHEM_PAR)[31];
K_HVa_aq = mxGetPr(CHEM_PAR)[32];
K_NH3_aq = mxGetPr(CHEM_PAR)[33];
K_NH4SO4 = mxGetPr(CHEM_PAR)[34];
K_H2O = mxGetPr(CHEM_PAR)[35];
K_S = mxGetPr(CHEM_PAR)[36];
K_CO2 = mxGetPr(CHEM_PAR)[37];
K_FeNO2_2 = mxGetPr(CHEM_PAR)[38];
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];
K_FeNO2 = mxGetPr(CHEM_PAR)[40];
K_HNO2_aq = mxGetPr(CHEM_PAR)[41];

DH_FeAc2 = mxGetPr(CHEM_PAR1)[0];
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];
DH_FeN2H6 = mxGetPr(CHEM_PAR1)[2];
DH_FeN3H9 = mxGetPr(CHEM_PAR1)[3];
DH_FeN4H12 = mxGetPr(CHEM_PAR1)[4];
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];
DH_FeOH2 = mxGetPr(CHEM_PAR1)[6];
DH_FeOH3 = mxGetPr(CHEM_PAR1)[7];
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];
DH_FeOH4 = mxGetPr(CHEM_PAR1)[9];
DH_FeS2O8 = mxGetPr(CHEM_PAR1)[10];
DH_Fe2O2H2 = mxGetPr(CHEM_PAR1)[11];
DH_Fe3O4H4 = mxGetPr(CHEM_PAR1)[12];
DH_FeAc = mxGetPr(CHEM_PAR1)[13];
DH_FeAc_2 = mxGetPr(CHEM_PAR1)[14];
DH_FeBu_2 = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3 = mxGetPr(CHEM_PAR1)[16];
DH_FeHS = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3 = mxGetPr(CHEM_PAR1)[18];
DH_FeOH = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2 = mxGetPr(CHEM_PAR1)[20];
DH_FePr = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4 = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2 = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq = mxGetPr(CHEM_PAR1)[27];
DH_HBu_aq = mxGetPr(CHEM_PAR1)[28];
DH_HCO3 = mxGetPr(CHEM_PAR1)[29];
DH_HPr_aq = mxGetPr(CHEM_PAR1)[30];
DH_HSO4 = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq = mxGetPr(CHEM_PAR1)[33];
DH_NH4SO4 = mxGetPr(CHEM_PAR1)[34];
DH_H2O = mxGetPr(CHEM_PAR1)[35];

```

```

DH_S      = mxGetPr(CHEM_PAR1)[36];
DH_CO2    = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2   = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq = mxGetPr(CHEM_PAR1)[41];

R  = 8.314;      // J/mole/K
Temp_C = 25.0;   // 15.0
Temp_K = Temp_C + 273.15;

KT_FeAc2   = pow(1.0E1,log(K_FeAc2)/log(1.0E1)-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1)-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6   = pow(1.0E1,log(K_FeN2H6)/log(1.0E1)-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9   = pow(1.0E1,log(K_FeN3H9)/log(1.0E1)-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12  = pow(1.0E1,log(K_FeN4H12)/log(1.0E1)-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1)-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2    = pow(1.0E1,log(K_FeOH2)/log(1.0E1)-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3    = pow(1.0E1,log(K_FeOH3)/log(1.0E1)-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1)-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4    = pow(1.0E1,log(K_FeOH4)/log(1.0E1)-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8   = pow(1.0E1,log(K_FeS2O8)/log(1.0E1)-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2  = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1)-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4  = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1)-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc     = pow(1.0E1,log(K_FeAc)/log(1.0E1)-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2   = pow(1.0E1,log(K_FeAc_2)/log(1.0E1)-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2   = pow(1.0E1,log(K_FeBu_2)/log(1.0E1)-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHCO3   = pow(1.0E1,log(K_FeHCO3)/log(1.0E1)-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS     = pow(1.0E1,log(K_FeHS)/log(1.0E1)-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3    = pow(1.0E1,log(K_FeNH3)/log(1.0E1)-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH     = pow(1.0E1,log(K_FeOH)/log(1.0E1)-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2   = pow(1.0E1,log(K_FeOH_2)/log(1.0E1)-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr     = pow(1.0E1,log(K_FePr)/log(1.0E1)-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1)-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4    = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeVa_2   = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2S_aq   = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-

```

```

3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

d = exp((-3.743E2)/Temp_C)*1.1445E-2-((Temp_C+2.889414E2)*pow(Temp_C-
3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0;
eps = Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-5.200087E4/Temp_K+2.727586E3;
A = 1.0/pow(Temp_K*(Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-
5.200087E4/Temp_K+2.727586E3),3.0/2.0)*sqrt(exp((-3.743E2)/Temp_C)*1.1445E-2-
((Temp_C+2.889414E2)*pow(Temp_C-3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0)*1.82483E6;

```

/ Find the order of these numbers*/ /*need to work out what is going on here*/*

```

Tot_NH4 = u[10];
Tot_Ac = u[6]/64.0;          /* Why is this divided by 64*/
Tot_Pr = u[5]/112.0;        /* Why is this divided by 112*/
Tot_CO3 = u[9];
Tot_SO4 = u[39];
Tot_Fe2 = u[46];
Tot_Fe3 = u[47];
Tot_HS = u[40]/64.0;        /* Why is this divided by 64*/
Tot_Bu = u[4]/160.0;        /* Why is this divided by 160*/
Tot_Va = u[3]/208.0;        /* Why is this divided by 208*/
Tot_NO3 = 0.0;
Tot_NO2 = 0.0;

```

/ Tot_PO4 = u[26] removed */*

```

S_act_H = x[0];      // Activity of hydrogen ion
S_act_NH4 = x[1];     // Activity of ammonium ion
S_act_Ac = x[2];      // Activity of acetate ion
S_act_Pr = x[3];      // Activity of propionate ion
S_act_CO3 = x[4];     // Activity of carbonate ion
S_act_SO4 = x[5];     // Activity of sulfate ion
S_act_Fe2 = x[6];

```

```

S_act_Fe3 = x[7];
S_act_HS = x[8];
S_act_Bu = x[9];
S_act_Va = x[10];
S_act_NO3 = x[11];
S_act_NO2 = x[12];
mu      = x[13]; // Ionic strength of the solution
gama    = x[14]; // Activity coefficient

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

Equi[0] = S_act_H/gama-(S_act_CO3*2.0)/gama2+S_act_NH4/gama-S_act_HS/gama-S_act_NO2/gama-
S_act_NO3/gama-S_act_Ac/gama-(S_act_SO4*2.0)/gama2+(S_act_Fe2*2.0)/gama2+(S_act_Fe3*3.0)/gama3-
S_act_Bu/gama-S_act_Va/gama-S_act_Pr/gama-KT_H2O/(S_act_H*gama)-(KT_HCO3*S_act_H*S_act_CO3)/gama-
(KT_HSO4*S_act_H*S_act_SO4)/gama+(KT_FeHS*S_act_HS*S_act_Fe2)/gama+(KT_FeNO2*S_act_NO2*S_act_Fe3*
2.0)/gama2+(KT_FeAc*S_act_Ac*S_act_Fe2)/gama-
(KT_NH4SO4*S_act_NH4*S_act_SO4)/gama+(KT_FeSO4*S_act_SO4*S_act_Fe3)/gama+(KT_FePr*S_act_Fe3*S_act_
Pr*2.0)/gama2+(KT_FeAc_2*S_act_Ac*S_act_Fe3*2.0)/gama2+(KT_FeBu_2*S_act_Fe3*S_act_Bu*2.0)/gama2+(KT_Fe
Va_2*S_act_Fe3*S_act_Va*2.0)/gama2-
(KT_S*S_act_HS*2.0)/(S_act_H*gama2)+(KT_FeOH*S_act_Fe2)/(S_act_H*gama)+(KT_FeOH2*1.0/(S_act_H*S_act_H)
*S_act_Fe3)/gama-(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama+(KT_FeOH_2*S_act_Fe3*2.0)/(S_act_H*gama
2)+(KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3)/gama-
(KT_FeSO8*(S_act_SO4*S_act_SO4)*S_act_Fe3)/gama+(KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3)/gama+
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*4.0)/gama4+(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_a
ct_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3)*5.0)/gama5+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_a
ct_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*
S_act_Fe2*2.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*
S_act_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama+(KT_FeNH3*S_act_NH4*S
_act_Fe2*2.0)/(S_act_H*gama2);

Equi[1] = -
Tot_NH4+S_act_NH4/gama+(KT_NH3_aq*S_act_NH4)/S_act_H+(KT_NH4SO4*S_act_NH4*S_act_SO4)/gama+(KT_Fe
N2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*2.0)/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S
_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*3.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*
S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2+(KT_FeNH3*S_act_NH4*S_act_Fe
2)/(S_act_H*gama2);

Equi[2] = -
Tot_Ac+S_act_Ac/gama+KT_HAc_aq*S_act_H*S_act_Ac+KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*S_act_Fe3*3.
0+(KT_FeAc*S_act_Ac*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Ac*S_act_Fe3)/gama2+(KT_FeAc2*(S_act_Ac*S_act_Ac
)*S_act_Fe3*2.0)/gama;

Equi[3] = -Tot_Pr+S_act_Pr/gama+KT_HPr_aq*S_act_H*S_act_Pr+(KT_FePr*S_act_Fe3*S_act_Pr)/gama2;

Equi[4] = -
Tot_CO3+S_act_CO3/gama2+KT_H2CO3_aq*(S_act_H*S_act_H)*S_act_CO3+(KT_HCO3*S_act_H*S_act_CO3)/gama
+(KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2)/gama;

Equi[5] = -
Tot_SO4+S_act_SO4/gama2+KT_FeSO4_aq*S_act_SO4*S_act_Fe2+(KT_HSO4*S_act_H*S_act_SO4)/gama+(KT_NH
4SO4*S_act_NH4*S_act_SO4)/gama+(KT_FeSO4*S_act_SO4*S_act_Fe3)/gama+(KT_FeSO8*(S_act_SO4*S_act_SO
4)*S_act_Fe3*2.0)/gama;

```

$$\text{Equ}[6] = - \text{Tot_Fe2} + \text{S_act_Fe2}/\text{gama2} + \text{KT_FeSO4_aq} * \text{S_act_SO4} * \text{S_act_Fe2} + \text{KT_FeOH2_aq} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe2} + (\text{KT_FeHS} * \text{S_act_HS} * \text{S_act_Fe2}) / \text{gama} + (\text{KT_FeAc} * \text{S_act_Ac} * \text{S_act_Fe2}) / \text{gama} + (\text{KT_FeOH} * \text{S_act_Fe2}) / (\text{S_act_H} * \text{gama}) + (\text{KT_FeOH3} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe2}) / \text{gama} + (\text{KT_FeN2H6} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2}) / \text{gama2} + (\text{KT_FeN3H9} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2}) / \text{gama2} + (\text{KT_FeN4H12} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2}) / \text{gama2} + (\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_CO3} * \text{S_act_Fe2}) / \text{gama} + (\text{KT_FeNH3} * \text{S_act_NH4} * \text{S_act_Fe2}) / (\text{S_act_H} * \text{gama2});$$

$$\text{Equ}[7] = - \text{Tot_Fe3} + \text{S_act_Fe3}/\text{gama3} + \text{KT_FeOH3_aq} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3} + \text{KT_FeNO2_aq} * (\text{S_act_NO2} * \text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3} + \text{KT_FeAc3_aq} * (\text{S_act_Ac} * \text{S_act_Ac} * \text{S_act_Ac}) * \text{S_act_Fe3} + (\text{KT_FeNO2} * \text{S_act_NO2} * \text{S_act_Fe3}) / \text{gama2} + (\text{KT_FeSO4} * \text{S_act_SO4} * \text{S_act_Fe3}) / \text{gama} + (\text{KT_FePr} * \text{S_act_Fe3} * \text{S_act_Pr}) / \text{gama2} + (\text{KT_FeAc2} * \text{S_act_Ac} * \text{S_act_Fe3}) / \text{gama2} + (\text{KT_FeBu2} * \text{S_act_Fe3} * \text{S_act_Bu}) / \text{gama2} + (\text{KT_FeVa2} * \text{S_act_Fe3} * \text{S_act_Va}) / \text{gama2} + (\text{KT_FeOH2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3}) / \text{gama} + (\text{KT_FeOH4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3}) / \text{gama} + (\text{KT_FeOH2} * \text{S_act_Fe3}) / (\text{S_act_H} * \text{gama2}) + (\text{KT_FeAc2} * (\text{S_act_Ac} * \text{S_act_Ac}) * \text{S_act_Fe3}) / \text{gama} + (\text{KT_FeS2O8} * (\text{S_act_SO4} * \text{S_act_SO4}) * \text{S_act_Fe3}) / \text{gama} + (\text{KT_FeNO22} * (\text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3}) / \text{gama} + (\text{KT_Fe2O2H2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (\text{S_act_Fe3} * \text{S_act_Fe3}) * 2.0) / \text{gama4} + (\text{KT_Fe3O4H4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_Fe3} * \text{S_act_Fe3} * \text{S_act_Fe3}) * 3.0) / \text{gama5};$$

$$\text{Equ}[8] = - \text{Tot_HS} + \text{S_act_HS}/\text{gama} + \text{KT_H2S_aq} * \text{S_act_H} * \text{S_act_HS} + (\text{KT_FeHS} * \text{S_act_HS} * \text{S_act_Fe2}) / \text{gama} + (\text{KT_S} * \text{S_act_HS}) / (\text{S_act_H} * \text{gama2});$$

$$\text{Equ}[9] = -\text{Tot_Bu} + \text{S_act_Bu}/\text{gama} + \text{KT_HBu_aq} * \text{S_act_H} * \text{S_act_Bu} + (\text{KT_FeBu2} * \text{S_act_Fe3} * \text{S_act_Bu}) / \text{gama2};$$

$$\text{Equ}[10] = -\text{Tot_Va} + \text{S_act_Va}/\text{gama} + \text{KT_HV a_aq} * \text{S_act_H} * \text{S_act_Va} + (\text{KT_FeVa2} * \text{S_act_Fe3} * \text{S_act_Va}) / \text{gama2};$$

$$\text{Equ}[11] = -\text{Tot_NO3} + \text{S_act_NO3}/\text{gama};$$

$$\text{Equ}[12] = - \text{Tot_NO2} + \text{S_act_NO2}/\text{gama} + \text{KT_HNO2_aq} * \text{S_act_H} * \text{S_act_NO2} + \text{KT_FeNO2_aq} * (\text{S_act_NO2} * \text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3} * 3.0 + (\text{KT_FeNO2} * \text{S_act_NO2} * \text{S_act_Fe3}) / \text{gama2} + (\text{KT_FeNO22} * (\text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3} * 2.0) / \text{gama};$$

$$\text{Equ}[13] = - \text{mu} + (\text{S_act_H} * (1.0/2.0)) / \text{gama} + (\text{S_act_NH4} * (1.0/2.0)) / \text{gama} + (\text{S_act_HS} * (1.0/2.0)) / \text{gama} + (\text{S_act_NO2} * (1.0/2.0)) / \text{gama} + (\text{S_act_NO3} * (1.0/2.0)) / \text{gama} + (\text{S_act_Ac} * (1.0/2.0)) / \text{gama} + (\text{S_act_SO4} * 2.0) / \text{gama2} + (\text{S_act_Fe2} * 2.0) / \text{gama2} + (\text{S_act_Fe3} * (9.0/2.0)) / \text{gama3} + (\text{S_act_Bu} * (1.0/2.0)) / \text{gama} + (\text{S_act_Va} * (1.0/2.0)) / \text{gama} + (\text{S_act_Pr} * (1.0/2.0)) / \text{gama} + (\text{KT_H2O} * (1.0/2.0)) / (\text{S_act_H} * \text{gama}) + (\text{KT_HCO3} * \text{S_act_H} * \text{S_act_CO3} * (1.0/2.0)) / \text{gama} + (\text{KT_HSO4} * \text{S_act_H} * \text{S_act_SO4} * (1.0/2.0)) / \text{gama} + (\text{KT_FeHS} * \text{S_act_HS} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_FeNO2} * \text{S_act_NO2} * \text{S_act_Fe3} * 2.0) / \text{gama2} + (\text{KT_FeAc} * \text{S_act_Ac} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_NH4SO4} * \text{S_act_NH4} * \text{S_act_SO4} * (1.0/2.0)) / \text{gama} + (\text{KT_FeSO4} * \text{S_act_SO4} * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FePr} * \text{S_act_Fe3} * \text{S_act_Pr} * 2.0) / \text{gama2} + (\text{KT_FeAc2} * \text{S_act_Ac} * \text{S_act_Fe3} * 2.0) / \text{gama2} + (\text{KT_FeBu2} * \text{S_act_Fe3} * \text{S_act_Bu} * 2.0) / \text{gama2} + (\text{KT_FeVa2} * \text{S_act_Fe3} * \text{S_act_Va} * 2.0) / \text{gama2} + (\text{KT_S} * \text{S_act_HS} * 2.0) / (\text{S_act_H} * \text{gama2}) + (\text{KT_FeOH} * \text{S_act_Fe2} * (1.0/2.0)) / (\text{S_act_H} * \text{gama}) + (\text{KT_FeOH2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH3} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH2} * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH2} * \text{S_act_Fe3} * 2.0) / (\text{S_act_H} * \text{gama2}) + (\text{KT_FeAc2} * (\text{S_act_Ac} * \text{S_act_Ac}) * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeS2O8} * (\text{S_act_SO4} * \text{S_act_SO4}) * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeNO22} * (\text{S_act_NO2} * \text{S_act_NO2}) * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_Fe2O2H2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (\text{S_act_Fe3} * \text{S_act_Fe3}) * 8.0) / \text{gama4} + (\text{KT_Fe3O4H4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_Fe3} * \text{S_act_Fe3} * \text{S_act_Fe3}) * (2.5 \text{E} 1/2.0)) / \text{gama5} + (\text{KT_FeN2H6} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2} * 2.0) / \text{gama2} + (\text{KT_FeN3H9} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2} * 2.0) / \text{gama2} + (\text{KT_FeN4H12} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * \text{S_act_Fe2} * 2.0) / \text{gama2} + (\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_CO3} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_FeNH3} * \text{S_act_NH4} * \text{S_act_Fe2} * 2.0) / (\text{S_act_H} * \text{gama2});$$

$$\text{Equ}[14] = -\text{gama} + \text{pow}(1.0\text{E}1, \text{A} * (\text{mu} * (3.0/1.0\text{E}1) - \text{sqrt}(\text{mu}) / (\text{sqrt}(\text{mu}) + 1.0)));$$

```

    return 1;
}

bool JacAna(const double x[], double *jac)
{
    SimStruct *S;
    double *u;

    double K_FeAc2,K_FeAc3_aq,K_FeN2H6,K_FeN3H9,K_FeN4H12,K_FeOH2_aq,K_FeOH2;
    double
    K_FeOH3,K_FeOH3_aq,K_FeOH4,K_FeS2O8,K_Fe2O2H2,K_Fe3O4H4,K_FeAc,K_FeAc_2,K_FeBu_2,K_FeHCO3,K_
    FeHS,K_FeNH3,K_FeOH,K_FeOH_2,K_FePr,K_FeSO4_aq,K_FeSO4,K_FeVa_2,K_H2CO3_aq,K_H2S_aq,K_HAc_aq,
    K_HBu_aq,K_HCO3,K_HPr_aq,K_HSO4,K_HVa_aq;
    double K_NH3_aq,K_NH4SO4,K_H2O,K_S,K_CO2;
    double K_FeNO2_2,K_FeNO2_aq,K_FeNO2,K_HNO2_aq;

    double DH_FeAc2,DH_FeAc3_aq,DH_FeN2H6,DH_FeN3H9,DH_FeN4H12,DH_FeOH2_aq,DH_FeOH2;
    double
    DH_FeOH3,DH_FeOH3_aq,DH_FeOH4,DH_FeS2O8,DH_Fe2O2H2,DH_Fe3O4H4,DH_FeAc,DH_FeAc_2,DH_FeBu_2,
    DH_FeHCO3,DH_FeHS,DH_FeNH3,DH_FeOH,DH_FeOH_2,DH_FePr,DH_FeSO4_aq,DH_FeSO4,DH_FeVa_2,DH_H2
    CO3_aq,DH_H2S_aq,DH_HAc_aq,DH_HBu_aq,DH_HCO3,DH_HPr_aq,DH_HSO4,DH_HVa_aq;
    double DH_NH3_aq,DH_NH4SO4,DH_H2O,DH_S,DH_CO2;
    double DH_FeNO2_2,DH_FeNO2_aq,DH_FeNO2,DH_HNO2_aq;

    double KT_FeAc2,KT_FeAc3_aq,KT_FeN2H6,KT_FeN3H9,KT_FeN4H12,KT_FeOH2_aq,KT_FeOH2;
    double
    KT_FeOH3,KT_FeOH3_aq,KT_FeOH4,KT_FeS2O8,KT_Fe2O2H2,KT_Fe3O4H4,KT_FeAc,KT_FeAc_2,KT_FeBu_2,KT
    _FeHCO3,KT_FeHS,KT_FeNH3,KT_FeOH,KT_FeOH_2,KT_FePr,KT_FeSO4_aq,KT_FeSO4,KT_FeVa_2,KT_H2CO3_
    aq,KT_H2S_aq,KT_HAc_aq,KT_HBu_aq,KT_HCO3,KT_HPr_aq,KT_HSO4,KT_HVa_aq;
    double KT_NH3_aq,KT_NH4SO4,KT_H2O,KT_S,KT_CO2;
    double KT_FeNO2_2,KT_FeNO2_aq,KT_FeNO2,KT_HNO2_aq;

    double R, Temp_K, Temp_C;

    double d, eps, A;

    double S_act_H, S_act_NH4,S_act_Ac,S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
    S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, mu, gama;
    double Tot_H,Tot_NH4, Tot_Ac, Tot_Pr, Tot_CO3, Tot_SO4, Tot_Fe2, Tot_Fe3, Tot_HS, Tot_Bu, Tot_Va, Tot_NO3,
    Tot_NO2;
    double gama2, gama3, gama4, gama5;

    double gradEqu[231];

    unsigned int i,j;
    bool status = 1;

    S=Smain;
    u=uemain;

    K_FeAc2   = mxGetPr(CHEM_PAR)[0];
    K_FeAc3_aq = mxGetPr(CHEM_PAR)[1];
    K_FeN2H6   = mxGetPr(CHEM_PAR)[2];
    K_FeN3H9   = mxGetPr(CHEM_PAR)[3];
    K_FeN4H12  = mxGetPr(CHEM_PAR)[4];
    K_FeOH2_aq = mxGetPr(CHEM_PAR)[5];
    K_FeOH2    = mxGetPr(CHEM_PAR)[6];
    K_FeOH3    = mxGetPr(CHEM_PAR)[7];
    K_FeOH3_aq = mxGetPr(CHEM_PAR)[8];
    K_FeOH4    = mxGetPr(CHEM_PAR)[9];
    K_FeS2O8   = mxGetPr(CHEM_PAR)[10];
    K_Fe2O2H2  = mxGetPr(CHEM_PAR)[11];
    K_Fe3O4H4  = mxGetPr(CHEM_PAR)[12];

```

```

K_FeAc   = mxGetPr(CHEM_PAR)[13];
K_FeAc_2 = mxGetPr(CHEM_PAR)[14];
K_FeBu_2 = mxGetPr(CHEM_PAR)[15];
K_FeHCO3 = mxGetPr(CHEM_PAR)[16];
K_FeHS    = mxGetPr(CHEM_PAR)[17];
K_FeNH3   = mxGetPr(CHEM_PAR)[18];
K_FeOH    = mxGetPr(CHEM_PAR)[19];
K_FeOH_2  = mxGetPr(CHEM_PAR)[20];
K_FePr    = mxGetPr(CHEM_PAR)[21];
K_FeSO4_aq = mxGetPr(CHEM_PAR)[22];
K_FeSO4    = mxGetPr(CHEM_PAR)[23];
K_FeVa_2   = mxGetPr(CHEM_PAR)[24];
K_H2CO3_aq = mxGetPr(CHEM_PAR)[25];
K_H2S_aq   = mxGetPr(CHEM_PAR)[26];
K_HAc_aq   = mxGetPr(CHEM_PAR)[27];
K_HBu_aq   = mxGetPr(CHEM_PAR)[28];
K_HCO3     = mxGetPr(CHEM_PAR)[29];
K_HPr_aq   = mxGetPr(CHEM_PAR)[30];
K_HSO4     = mxGetPr(CHEM_PAR)[31];
K_HVa_aq   = mxGetPr(CHEM_PAR)[32];
K_NH3_aq   = mxGetPr(CHEM_PAR)[33];
K_NH4SO4   = mxGetPr(CHEM_PAR)[34];
K_H2O      = mxGetPr(CHEM_PAR)[35];
K_S        = mxGetPr(CHEM_PAR)[36];
K_CO2      = mxGetPr(CHEM_PAR)[37];
K_FeNO2_2  = mxGetPr(CHEM_PAR)[38];
K_FeNO2_aq = mxGetPr(CHEM_PAR)[39];
K_FeNO2    = mxGetPr(CHEM_PAR)[40];
K_HNO2_aq  = mxGetPr(CHEM_PAR)[41];

DH_FeAc2   = mxGetPr(CHEM_PAR1)[0];
DH_FeAc3_aq = mxGetPr(CHEM_PAR1)[1];
DH_FeN2H6  = mxGetPr(CHEM_PAR1)[2];
DH_FeN3H9  = mxGetPr(CHEM_PAR1)[3];
DH_FeN4H12 = mxGetPr(CHEM_PAR1)[4];
DH_FeOH2_aq = mxGetPr(CHEM_PAR1)[5];
DH_FeOH2    = mxGetPr(CHEM_PAR1)[6];
DH_FeOH3    = mxGetPr(CHEM_PAR1)[7];
DH_FeOH3_aq = mxGetPr(CHEM_PAR1)[8];
DH_FeOH4    = mxGetPr(CHEM_PAR1)[9];
DH_FeS2O8   = mxGetPr(CHEM_PAR1)[10];
DH_Fe2O2H2  = mxGetPr(CHEM_PAR1)[11];
DH_Fe3O4H4  = mxGetPr(CHEM_PAR1)[12];
DH_FeAc     = mxGetPr(CHEM_PAR1)[13];
DH_FeAc_2   = mxGetPr(CHEM_PAR1)[14];
DH_FeBu_2   = mxGetPr(CHEM_PAR1)[15];
DH_FeHCO3   = mxGetPr(CHEM_PAR1)[16];
DH_FeHS     = mxGetPr(CHEM_PAR1)[17];
DH_FeNH3    = mxGetPr(CHEM_PAR1)[18];
DH_FeOH     = mxGetPr(CHEM_PAR1)[19];
DH_FeOH_2   = mxGetPr(CHEM_PAR1)[20];
DH_FePr     = mxGetPr(CHEM_PAR1)[21];
DH_FeSO4_aq = mxGetPr(CHEM_PAR1)[22];
DH_FeSO4    = mxGetPr(CHEM_PAR1)[23];
DH_FeVa_2   = mxGetPr(CHEM_PAR1)[24];
DH_H2CO3_aq = mxGetPr(CHEM_PAR1)[25];
DH_H2S_aq   = mxGetPr(CHEM_PAR1)[26];
DH_HAc_aq   = mxGetPr(CHEM_PAR1)[27];
DH_HBu_aq   = mxGetPr(CHEM_PAR1)[28];
DH_HCO3     = mxGetPr(CHEM_PAR1)[29];
DH_HPr_aq   = mxGetPr(CHEM_PAR1)[30];
DH_HSO4     = mxGetPr(CHEM_PAR1)[31];
DH_HVa_aq   = mxGetPr(CHEM_PAR1)[32];
DH_NH3_aq   = mxGetPr(CHEM_PAR1)[33];

```

```

DH_NH4SO4 = mxGetPr(CHEM_PAR1)[34];
DH_H2O    = mxGetPr(CHEM_PAR1)[35];
DH_S      = mxGetPr(CHEM_PAR1)[36];
DH_CO2    = mxGetPr(CHEM_PAR1)[37];
DH_FeNO2_2 = mxGetPr(CHEM_PAR1)[38];
DH_FeNO2_aq = mxGetPr(CHEM_PAR1)[39];
DH_FeNO2   = mxGetPr(CHEM_PAR1)[40];
DH_HNO2_aq = mxGetPr(CHEM_PAR1)[41];

R = 8.314;           // J/mole/K
Temp_C = 25.0;       // 15.0; Why is the temp 15 and not 35
Temp_K = Temp_C + 273.15;

KT_FeAc2 = pow(1.0E1,log(K_FeAc2)/log(1.0E1)-(DH_FeAc2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc3_aq = pow(1.0E1,log(K_FeAc3_aq)/log(1.0E1)-(DH_FeAc3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN2H6 = pow(1.0E1,log(K_FeN2H6)/log(1.0E1)-(DH_FeN2H6*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN3H9 = pow(1.0E1,log(K_FeN3H9)/log(1.0E1)-(DH_FeN3H9*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeN4H12 = pow(1.0E1,log(K_FeN4H12)/log(1.0E1)-(DH_FeN4H12*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2_aq = pow(1.0E1,log(K_FeOH2_aq)/log(1.0E1)-(DH_FeOH2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH2 = pow(1.0E1,log(K_FeOH2)/log(1.0E1)-(DH_FeOH2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3 = pow(1.0E1,log(K_FeOH3)/log(1.0E1)-(DH_FeOH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH3_aq = pow(1.0E1,log(K_FeOH3_aq)/log(1.0E1)-(DH_FeOH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH4 = pow(1.0E1,log(K_FeOH4)/log(1.0E1)-(DH_FeOH4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeS2O8 = pow(1.0E1,log(K_FeS2O8)/log(1.0E1)-(DH_FeS2O8*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe2O2H2 = pow(1.0E1,log(K_Fe2O2H2)/log(1.0E1)-(DH_Fe2O2H2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_Fe3O4H4 = pow(1.0E1,log(K_Fe3O4H4)/log(1.0E1)-(DH_Fe3O4H4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc = pow(1.0E1,log(K_FeAc)/log(1.0E1)-(DH_FeAc*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeAc_2 = pow(1.0E1,log(K_FeAc_2)/log(1.0E1)-(DH_FeAc_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeBu_2 = pow(1.0E1,log(K_FeBu_2)/log(1.0E1)-(DH_FeBu_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHCO3 = pow(1.0E1,log(K_FeHCO3)/log(1.0E1)-(DH_FeHCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeHS = pow(1.0E1,log(K_FeHS)/log(1.0E1)-(DH_FeHS*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNH3 = pow(1.0E1,log(K_FeNH3)/log(1.0E1)-(DH_FeNH3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH = pow(1.0E1,log(K_FeOH)/log(1.0E1)-(DH_FeOH*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeOH_2 = pow(1.0E1,log(K_FeOH_2)/log(1.0E1)-(DH_FeOH_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FePr = pow(1.0E1,log(K_FePr)/log(1.0E1)-(DH_FePr*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4_aq = pow(1.0E1,log(K_FeSO4_aq)/log(1.0E1)-(DH_FeSO4_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeSO4 = pow(1.0E1,log(K_FeSO4)/log(1.0E1)-(DH_FeSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeVa_2 = pow(1.0E1,log(K_FeVa_2)/log(1.0E1)-(DH_FeVa_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2CO3_aq = pow(1.0E1,log(K_H2CO3_aq)/log(1.0E1)-(DH_H2CO3_aq*(1.0/Temp_K-3.35401643468053E-

```

```

3)*4.342162396873643E-1)/R);
KT_H2S_aq = pow(1.0E1,log(K_H2S_aq)/log(1.0E1)-(DH_H2S_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HAc_aq = pow(1.0E1,log(K_HAc_aq)/log(1.0E1)-(DH_HAc_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HBu_aq = pow(1.0E1,log(K_HBu_aq)/log(1.0E1)-(DH_HBu_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HCO3 = pow(1.0E1,log(K_HCO3)/log(1.0E1)-(DH_HCO3*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HPr_aq = pow(1.0E1,log(K_HPr_aq)/log(1.0E1)-(DH_HPr_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HSO4 = pow(1.0E1,log(K_HSO4)/log(1.0E1)-(DH_HSO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HVa_aq = pow(1.0E1,log(K_HVa_aq)/log(1.0E1)-(DH_HVa_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH3_aq = pow(1.0E1,log(K_NH3_aq)/log(1.0E1)-(DH_NH3_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_NH4SO4 = pow(1.0E1,log(K_NH4SO4)/log(1.0E1)-(DH_NH4SO4*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_H2O = pow(1.0E1,log(K_H2O)/log(1.0E1)-(DH_H2O*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_S = pow(1.0E1,log(K_S)/log(1.0E1)-(DH_S*(1.0/Temp_K-3.35401643468053E-3)*4.342162396873643E-
1)/R);
KT_CO2 = pow(1.0E1,log(K_CO2)/log(1.0E1)-(DH_CO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_2 = pow(1.0E1,log(K_FeNO2_2)/log(1.0E1)-(DH_FeNO2_2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2_aq = pow(1.0E1,log(K_FeNO2_aq)/log(1.0E1)-(DH_FeNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_FeNO2 = pow(1.0E1,log(K_FeNO2)/log(1.0E1)-(DH_FeNO2*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);
KT_HNO2_aq = pow(1.0E1,log(K_HNO2_aq)/log(1.0E1)-(DH_HNO2_aq*(1.0/Temp_K-3.35401643468053E-
3)*4.342162396873643E-1)/R);

```

```

d = exp((-3.743E2)/Temp_C)*1.1445E-2-((Temp_C+2.889414E2)*pow(Temp_C-
3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0;
eps = Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-5.200087E4/Temp_K+2.727586E3;
A = 1.0/pow(Temp_K*(Temp_K*6.224107E-1-log(Temp_K)*4.669151E2-
5.200087E4/Temp_K+2.727586E3),3.0/2.0)*sqrt(exp((-3.743E2)/Temp_C)*1.1445E-2-
((Temp_C+2.889414E2)*pow(Temp_C-3.9863,2.0))/(Temp_C*5.089292E5+3.4673158092196E7)+1.0)*1.82483E6;

```

```

Tot_NH4 = u[10];
Tot_Ac = u[6]/64.0;
Tot_Pr = u[5]/112.0;
Tot_CO3 = u[9];
Tot_SO4 = u[39];
Tot_Fe2 = u[46];
Tot_Fe3 = u[47];
Tot_HS = u[40]/64.0;
Tot_Bu = u[4]/160.0;
Tot_Va = u[3]/208.0;
Tot_NO3 = 0.0;
Tot_NO2 = 0.0;

```

```

S_act_H = x[0]; // Activity of hydrogen ion
S_act_NH4 = x[1]; // Activity of ammonium ion
S_act_Ac = x[2]; // Activity of acetate ion
S_act_Pr = x[3]; // Activity of propionate ion
S_act_CO3 = x[4]; // Activity of carbonate ion
S_act_SO4 = x[5]; // Activity of sulfate ion
S_act_Fe2 = x[6];

```

```

S_act_Fe3 = x[7];
S_act_HS = x[8];
S_act_Bu = x[9];
S_act_Va = x[10];
S_act_NO3 = x[11];
S_act_NO2 = x[12];
mu      = x[13]; // Ionic strength of the solution
gama    = x[14]; // Activity coefficient

gama2 = pow(gama,4); // For divalent ions
gama3 = pow(gama,9); // For trivalent ions
gama4 = pow(gama,16); // 4 times charged ions
gama5 = pow(gama,25); // 5 times charged ions

// Eq1: Charge balance

gradEqu[0] = 1.0/gama+(KT_H2O*1.0/(S_act_H*S_act_H))/gama-(KT_HCO3*S_act_CO3)/gama-
(KT_HSO4*S_act_SO4)/gama+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama+(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS*
2.0)/gama2-(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3*2.0)/gama+(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe2*3.0)/gama+(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*4.0)/gama-
(KT_FeOH_2*1.0/(S_act_H*S_act_H)*S_act_Fe3*2.0)/gama2-
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*8.0)/gama4-
(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3)*2.0E1)/gama5-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*8.0)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0)/gama2;

gradEqu[1] = 1.0/gama-
(KT_NH4SO4*S_act_SO4)/gama+(KT_FeNH3*S_act_Fe2*2.0)/(S_act_H*gama2)+(KT_FeN3H9*1.0/(S_act_H*S_act_H*
S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(
S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*8.0)/gama2+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*4.0)/gama2;

gradEqu[2] = -
1.0/gama+(KT_FeAc*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Fe3*2.0)/gama2+(KT_FeAc2*S_act_Ac*S_act_Fe3*2.0)/gama;

gradEqu[3] = -1.0/gama+(KT_FePr*S_act_Fe3*2.0)/gama2;

gradEqu[4] = -2.0/gama2-(KT_HCO3*S_act_H)/gama+(KT_FeHCO3*S_act_H*S_act_Fe2)/gama;

gradEqu[5] = -2.0/gama2-(KT_HSO4*S_act_H)/gama-
(KT_NH4SO4*S_act_NH4)/gama+(KT_FeSO4*S_act_Fe3)/gama-(KT_FeS2O8*S_act_SO4*S_act_Fe3*2.0)/gama;

gradEqu[6] = 2.0/gama2+KT_FeOH/(S_act_H*gama)-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H))/gama+(KT_FeHS*S_act_HS)/gama+(KT_FeAc*S_act_Ac)/gama+(KT_FeHCO3*S_act_H*S_act_CO3)/gama+(KT_FeNH3*S_act_NH4*2.0)/(S_act_H*gama2)+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*2.0)/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*2.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*2.0)/gama2;

gradEqu[7] = 3.0/gama3+(KT_FeOH2*1.0/(S_act_H*S_act_H))/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H))/gama+(KT_FeOH_2*2.0)/(S_act_H*gama2)+(KT_FeAc2*(S_act_Ac*S_act_Ac))/gama-

```

$$\frac{(KT_FeS2O8*(S_act_SO4*S_act_SO4))/gama+(KT_FeNO2_2*(S_act_NO2*S_act_NO2))/gama+(KT_FeNO2*S_act_NO2*2.0)/gama2+(KT_FeSO4*S_act_SO4)/gama+(KT_FePr*S_act_Pr*2.0)/gama2+(KT_FeAc_2*S_act_Ac*2.0)/gama2+(KT_FeBu_2*S_act_Bu*2.0)/gama2+(KT_FeVa_2*S_act_Va*2.0)/gama2+(KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*S_act_Fe3*8.0)/gama4+(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*1.5E1)/gama5;$$

$$gradEqu[8] = -1.0/gama-(KT_S*2.0)/(S_act_H*gama2)+(KT_FeHS*S_act_Fe2)/gama;$$

$$gradEqu[9] = -1.0/gama+(KT_FeBu_2*S_act_Fe3*2.0)/gama2;$$

$$gradEqu[10] = -1.0/gama+(KT_FeVa_2*S_act_Fe3*2.0)/gama2;$$

$$gradEqu[17] = -1.0/gama;$$

$$gradEqu[18] = -1.0/gama+(KT_FeNO2*S_act_Fe3*2.0)/gama2+(KT_FeNO2_2*S_act_NO2*S_act_Fe3*2.0)/gama;$$

$$gradEqu[19] = 0.0;$$

$$gradEqu[20] = -S_act_H*1.0/(gama*gama)-S_act_NH4*1.0/(gama*gama)+S_act_HS*1.0/(gama*gama)+S_act_NO2*1.0/(gama*gama)+S_act_NO3*1.0/(gama*gama)+S_act_Ac*1.0/(gama*gama)+S_act_Bu*1.0/(gama*gama)+S_act_Va*1.0/(gama*gama)+S_act_Pr*1.0/(gama*gama)+(KT_H2O*1.0/(gama*gama))/S_act_H+KT_HCO3*S_act_H*S_act_CO3*1.0/(gama*gama)+KT_HSO4*S_act_H*S_act_SO4*1.0/(gama*gama)-KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama)-KT_FeAc*S_act_Ac*S_act_Fe2*1.0/(gama*gama)+KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama)-KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)-(KT_FeOH*S_act_Fe2*1.0/(gama*gama))/S_act_H-KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)+KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*1.0/(gama*gama)+KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)-KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)+KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)-KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama)-KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama);$$

// Eq2: NH4

$$gradEqu[21] = -KT_NH3_aq*1.0/(S_act_H*S_act_H)*S_act_NH4-(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2-(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*9.0)/gama2-(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*1.6E1)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2)/gama2;$$

$$gradEqu[22] = KT_NH3_aq/S_act_H+1.0/gama+(KT_NH4SO4*S_act_SO4)/gama+(KT_FeNH3*S_act_Fe2)/(S_act_H*gama2)+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*9.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*1.6E1)/gama2+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*4.0)/gama2;$$

$$gradEqu[23] = 0.0;$$

$$gradEqu[24] = 0.0;$$

$$gradEqu[25] = 0.0;$$

$$gradEqu[26] = (KT_NH4SO4*S_act_NH4)/gama;$$

```

gradEqu[27] =
(KT_FeNH3*S_act_NH4)/(S_act_H*gama2)+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*2.0)/gama
2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*3.0)/gama2+(KT_FeN4H12*1.0
/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*4.0)/gama2;

```

```

gradEqu[28] = 0.0;

```

```

gradEqu[29] = 0.0;

```

```

gradEqu[30] = 0.0;

```

```

gradEqu[31] = 0.0;

```

```

gradEqu[32] = 0.0;

```

```

gradEqu[33] = 0.0;

```

```

gradEqu[34] = 0.0;

```

```

gradEqu[35] = -S_act_NH4*1.0/(gama*gama)-KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama);

```

```

// Eq3: Ac

```

```

gradEqu[36] = KT_HAc_aq*S_act_Ac;

```

```

gradEqu[37] = 0.0;

```

```

gradEqu[38] =
1.0/gama+KT_HAc_aq*S_act_H+(KT_FeAc*S_act_Fe2)/gama+(KT_FeAc_2*S_act_Fe3)/gama2+(KT_FeAc2*S_act_Ac*
S_act_Fe3*4.0)/gama;

```

```

gradEqu[39] = 0.0;

```

```

gradEqu[40] = 0.0;

```

```

gradEqu[41] = 0.0;

```

```

gradEqu[42] = (KT_FeAc*S_act_Ac)/gama;

```

```

gradEqu[43] =
KT_FeAc3_aq*(S_act_Ac*S_act_Ac*S_act_Ac)*3.0+(KT_FeAc2*(S_act_Ac*S_act_Ac)*2.0)/gama+(KT_FeAc_2*S_act_A
c)/gama2;

```

```

gradEqu[44] = 0.0;

```

```

gradEqu[45] = 0.0;

gradEqu[46] = 0.0;

gradEqu[47] = 0.0;

gradEqu[48] = 0.0;

gradEqu[49] = 0.0;

gradEqu[50] = -S_act_Ac*1.0/(gama*gama)-KT_FeAc*S_act_Ac*S_act_Fe2*1.0/(gama*gama)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)*2.0;

// Eq4: Pr

gradEqu[51] = KT_HPr_aq*S_act_Pr;

gradEqu[52] = 0.0;

gradEqu[53] = 0.0;

gradEqu[54] = 1.0/gama+KT_HPr_aq*S_act_H+(KT_FePr*S_act_Fe3)/gama2;

gradEqu[55] = 0.0;

gradEqu[56] = 0.0;

gradEqu[57] = 0.0;

gradEqu[58] = (KT_FePr*S_act_Pr)/gama2;

gradEqu[59] = 0.0;

gradEqu[60] = 0.0;

gradEqu[61] = 0.0;

gradEqu[62] = 0.0;

gradEqu[63] = 0.0;

gradEqu[64] = 0.0;

```

```

gradEqu[65] = -S_act_Pr*1.0/(gama*gama);

// Eq5: CO3

gradEqu[66] = KT_H2CO3_aq*S_act_H*S_act_CO3*2.0+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama;

gradEqu[67] = 0.0;

gradEqu[68] = 0.0;

gradEqu[69] = 0.0;

gradEqu[70] =
KT_H2CO3_aq*(S_act_H*S_act_H)+1.0/gama2+(KT_HCO3*S_act_H)/gama+(KT_FeHCO3*S_act_H*S_act_Fe2)/gama;

gradEqu[71] = 0.0;

gradEqu[72] = (KT_FeHCO3*S_act_H*S_act_CO3)/gama;

gradEqu[73] = 0.0;

gradEqu[74] = 0.0;

gradEqu[75] = 0.0;

gradEqu[76] = 0.0;

gradEqu[77] = 0.0;

gradEqu[78] = 0.0;

gradEqu[79] = 0.0;

gradEqu[80] = -KT_HCO3*S_act_H*S_act_CO3*1.0/(gama*gama)-
KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama);

// Eq6: SO4

gradEqu[81] = (KT_HSO4*S_act_SO4)/gama;

gradEqu[82] = (KT_NH4SO4*S_act_SO4)/gama;

gradEqu[83] = 0.0;

```

```

gradEqu[84] = 0.0;

gradEqu[85] = 0.0;

gradEqu[86] =
1.0/gama2+KT_FeSO4_aq*S_act_Fe2+(KT_HSO4*S_act_H)/gama+(KT_NH4SO4*S_act_NH4)/gama+(KT_FeSO4*S_a
ct_Fe3)/gama+(KT_FeS2O8*S_act_SO4*S_act_Fe3*4.0)/gama;

gradEqu[87] = KT_FeSO4_aq*S_act_SO4;

gradEqu[88] = (KT_FeS2O8*(S_act_SO4*S_act_SO4)*2.0)/gama+(KT_FeSO4*S_act_SO4)/gama;

gradEqu[89] = 0.0;

gradEqu[90] = 0.0;

gradEqu[91] = 0.0;

gradEqu[92] = 0.0;

gradEqu[93] = 0.0;

gradEqu[94] = 0.0;

gradEqu[95] = -KT_HSO4*S_act_H*S_act_SO4*1.0/(gama*gama)-
KT_NH4SO4*S_act_NH4*S_act_SO4*1.0/(gama*gama)-KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)*2.0;

// Eq7: Fe2+

gradEqu[96] = -
KT_FeOH2_aq*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*2.0+(KT_FeHCO3*S_act_CO3*S_act_Fe2)/gama-
(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2)/gama-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe2*3.0)/gama-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*2.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*3.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S
_act_Fe2*4.0)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2)/gama2;

gradEqu[97] =
(KT_FeNH3*S_act_Fe2)/(S_act_H*gama2)+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_
act_Fe2*3.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S
_act_Fe2*4.0)/gama2+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0)/gama2;

gradEqu[98] = (KT_FeAc*S_act_Fe2)/gama;

gradEqu[99] = 0.0;

```

```

gradEqu[100] = (KT_FeHCO3*S_act_H*S_act_Fe2)/gama;

gradEqu[101] = KT_FeSO4_aq*S_act_Fe2;

gradEqu[102] =
KT_FeOH2_aq*1.0/(S_act_H*S_act_H)+1.0/gama2+KT_FeSO4_aq*S_act_SO4+KT_FeOH/(S_act_H*gama)+(KT_FeO
H3*1.0/(S_act_H*S_act_H*S_act_H))/gama+(KT_FeHS*S_act_HS)/gama+(KT_FeAc*S_act_Ac)/gama+(KT_FeHCO3*S
_act_H*S_act_CO3)/gama+(KT_FeNH3*S_act_NH4)/(S_act_H*gama2)+(KT_FeN2H6*1.0/(S_act_H*S_act_H)*(S_act_N
H4*S_act_NH4))/gama2+(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4))/gama2+
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4))/gama2;

gradEqu[103] = 0.0;

gradEqu[104] = (KT_FeHS*S_act_Fe2)/gama;

gradEqu[105] = 0.0;

gradEqu[106] = 0.0;

gradEqu[107] = 0.0;

gradEqu[108] = 0.0;

gradEqu[109] = 0.0;

gradEqu[110] = -KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama)-
KT_FeAc*S_act_Ac*S_act_Fe2*1.0/(gama*gama)-(KT_FeOH*S_act_Fe2*1.0/(gama*gama))/S_act_H-
KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*1.0/(gama*gama)-
KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama);

// Eq8: Fe3+

gradEqu[111] = KT_FeOH3_aq*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*-3.0-
(KT_FeOH2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3*2.0)/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*4.0)/gama-
(KT_FeOH_2*1.0/(S_act_H*S_act_H)*S_act_Fe3)/gama2-
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*4.0)/gama4-
(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3)*1.2E1)/gama5;

gradEqu[112] = 0.0;

gradEqu[113] =
KT_FeAc3_aq*(S_act_Ac*S_act_Ac)*S_act_Fe3*3.0+(KT_FeAc_2*S_act_Fe3)/gama2+(KT_FeAc2*S_act_Ac*S_act_Fe
3*2.0)/gama;

gradEqu[114] = (KT_FePr*S_act_Fe3)/gama2;

```

```

gradEqu[115] = 0.0;

gradEqu[116] = (KT_FeSO4*S_act_Fe3)/gama+(KT_FeS2O8*S_act_SO4*S_act_Fe3*2.0)/gama;

gradEqu[117] = 0.0;

gradEqu[118] =
KT_FeOH3_aq*1.0/(S_act_H*S_act_H*S_act_H)+KT_FeNO2_aq*(S_act_NO2*S_act_NO2*S_act_NO2)+KT_FeAc3_aq*
(S_act_Ac*S_act_Ac*S_act_Ac)+1.0/gama3+(KT_FeOH2*1.0/(S_act_H*S_act_H))/gama+(KT_FeOH4*1.0/(S_act_H*S_
act_H*S_act_H*S_act_H))/gama+KT_FeOH_2/(S_act_H*gama2)+(KT_FeAc2*(S_act_Ac*S_act_Ac))/gama+(KT_FeS2O
8*(S_act_SO4*S_act_SO4))/gama+(KT_FeNO2_2*(S_act_NO2*S_act_NO2))/gama+(KT_FeNO2*S_act_NO2)/gama2+(
KT_FeSO4*S_act_SO4)/gama+(KT_FePr*S_act_Pr)/gama2+(KT_FeAc_2*S_act_Ac)/gama2+(KT_FeBu_2*S_act_Bu)/g
ama2+(KT_FeVa_2*S_act_Va)/gama2+(KT_Fe2O2H2*1.0/(S_act_H*S_act_H)*S_act_Fe3*4.0)/gama4+(KT_Fe3O4H4*
1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*9.0)/gama5;

gradEqu[119] = 0.0;

gradEqu[120] = (KT_FeBu_2*S_act_Fe3)/gama2;

gradEqu[121] = (KT_FeVa_2*S_act_Fe3)/gama2;

gradEqu[122] = 0.0;

gradEqu[123] =
KT_FeNO2_aq*(S_act_NO2*S_act_NO2)*S_act_Fe3*3.0+(KT_FeNO2*S_act_Fe3)/gama2+(KT_FeNO2_2*S_act_NO2*
S_act_Fe3*2.0)/gama;

gradEqu[124] = 0.0;

gradEqu[125] = -KT_FeSO4*S_act_SO4*S_act_Fe3*1.0/(gama*gama)-
KT_FeOH2*1.0/(S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)-
KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama);

// Eq9:HS

gradEqu[126] = KT_H2S_aq*S_act_HS-(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS)/gama2;

gradEqu[127] = 0.0;

gradEqu[128] = 0.0;

gradEqu[129] = 0.0;

```

```

gradEqu[130] = 0.0;

gradEqu[131] = 0.0;

gradEqu[132] = (KT_FeHS*S_act_HS)/gama;

gradEqu[133] = 0.0;

gradEqu[134] = 1.0/gama+KT_H2S_aq*S_act_H+KT_S/(S_act_H*gama2)+(KT_FeHS*S_act_Fe2)/gama;

gradEqu[135] = 0.0;

gradEqu[136] = 0.0;

gradEqu[137] = 0.0;

gradEqu[138] = 0.0;

gradEqu[139] = 0.0;

gradEqu[140] = -S_act_HS*1.0/(gama*gama)-KT_FeHS*S_act_HS*S_act_Fe2*1.0/(gama*gama);

// Eq10:Bu

gradEqu[141] = KT_HBu_aq*S_act_Bu;

gradEqu[142] = 0.0;

gradEqu[143] = 0.0;

gradEqu[144] = 0.0;

gradEqu[145] = 0.0;

gradEqu[146] = 0.0;

gradEqu[147] = 0.0;

gradEqu[148] = (KT_FeBu_2*S_act_Bu)/gama2;

gradEqu[149] = 0.0;

gradEqu[150] = 1.0/gama+KT_HBu_aq*S_act_H+(KT_FeBu_2*S_act_Fe3)/gama2;

```

```

gradEqu[151] = 0.0;

gradEqu[152] = 0.0;

gradEqu[153] = 0.0;

gradEqu[154] = 0.0;

gradEqu[155] = -S_act_Bu*1.0/(gama*gama);

// Eq11:Va

gradEqu[156] = KT_HVa_aq*S_act_Va;

gradEqu[157] = 0.0;

gradEqu[158] = 0.0;

gradEqu[159] = 0.0;

gradEqu[160] = 0.0;

gradEqu[161] = 0.0;

gradEqu[162] = 0.0;

gradEqu[163] = (KT_FeVa_2*S_act_Va)/gama2;

gradEqu[164] = 0.0;

gradEqu[165] = 0.0;

gradEqu[166] = 1.0/gama+KT_HVa_aq*S_act_H+(KT_FeVa_2*S_act_Fe3)/gama2;

gradEqu[167] = 0.0;

gradEqu[168] = 0.0;

gradEqu[169] = 0.0;

gradEqu[170] = -S_act_Va*1.0/(gama*gama);

```

// Eq12:NO3

gradEqu[171] = 0.0;

gradEqu[172] = 0.0;

gradEqu[173] = 0.0;

gradEqu[174] = 0.0;

gradEqu[175] = 0.0;

gradEqu[176] = 0.0;

gradEqu[177] = 0.0;

gradEqu[178] = 0.0;

gradEqu[179] = 0.0;

gradEqu[180] = 0.0;

gradEqu[181] = 0.0;

gradEqu[182] = 1.0/gama;

gradEqu[183] = 0.0;

gradEqu[184] = 0.0;

gradEqu[185] = -S_act_NO3*1.0/(gama*gama);

// Eq13:NO2

gradEqu[186] = KT_HNO2_aq*S_act_NO2;

gradEqu[187] = 0.0;

gradEqu[188] = 0.0;

gradEqu[189] = 0.0;

```

gradEqu[190] = 0.0;

gradEqu[191] = 0.0;

gradEqu[192] = 0.0;

gradEqu[193] =
KT_FeNO2_aq*(S_act_NO2*S_act_NO2*S_act_NO2)*3.0+(KT_FeNO2_2*(S_act_NO2*S_act_NO2)*2.0)/gama+(KT_Fe
NO2*S_act_NO2)/gama2;

gradEqu[194] = 0.0;

gradEqu[195] = 0.0;

gradEqu[196] = 0.0;

gradEqu[197] = 0.0;

gradEqu[198] =
1.0/gama+KT_HNO2_aq*S_act_H+KT_FeNO2_aq*(S_act_NO2*S_act_NO2)*S_act_Fe3*9.0+(KT_FeNO2*S_act_Fe3)/g
ama2+(KT_FeNO2_2*S_act_NO2*S_act_Fe3*4.0)/gama;

gradEqu[199] = 0.0;

gradEqu[200] = -S_act_NO2*1.0/(gama*gama)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama)*2.0;

// Eq14: mu

gradEqu[201] = (1.0/2.0)/gama-
(KT_H2O*1.0/(S_act_H*S_act_H)*(1.0/2.0))/gama+(KT_HCO3*S_act_CO3*(1.0/2.0))/gama+(KT_HSO4*S_act_SO4*(1.
0/2.0))/gama+(KT_FeHCO3*S_act_CO3*S_act_Fe2*(1.0/2.0))/gama-
(KT_S*1.0/(S_act_H*S_act_H)*S_act_HS*2.0)/gama2-(KT_FeOH*1.0/(S_act_H*S_act_H)*S_act_Fe2*(1.0/2.0))/gama-
(KT_FeOH2*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe3)/gama-
(KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe2*(3.0/2.0))/gama-
(KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*2.0)/gama-
(KT_FeOH_2*1.0/(S_act_H*S_act_H)*S_act_Fe3*2.0)/gama2-
(KT_Fe2O2H2*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3)*1.6E1)/gama4-
(KT_Fe3O4H4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_Fe3*S_act_Fe3*S_act_Fe3)*5.0E1)/gama5-
(KT_FeN2H6*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*4.0)/gama2-
(KT_FeN3H9*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2-
(KT_FeN4H12*1.0/(S_act_H*S_act_H*S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4*S_act_NH4)*S
_act_Fe2*8.0)/gama2-(KT_FeNH3*1.0/(S_act_H*S_act_H)*S_act_NH4*S_act_Fe2*2.0)/gama2;

gradEqu[202] =
(1.0/2.0)/gama+(KT_NH4SO4*S_act_SO4*(1.0/2.0))/gama+(KT_FeNH3*S_act_Fe2*2.0)/(S_act_H*gama2)+(KT_FeN3H
9*1.0/(S_act_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4)*S_act_Fe2*6.0)/gama2+(KT_FeN4H12*1.0/(S_act_H*S_ac
t_H*S_act_H*S_act_H)*(S_act_NH4*S_act_NH4*S_act_NH4)*S_act_Fe2*8.0)/gama2+(KT_FeN2H6*1.0/(S_act_H*S_act
_H)*S_act_NH4*S_act_Fe2*4.0)/gama2;

```

$$\text{gradEqu}[203] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_FeAc} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama} + (\text{KT_FeAc_2} * \text{S_act_Fe3} * 2.0) / \text{gama}^2 + (\text{KT_FeAc2} * \text{S_act_Ac} * \text{S_act_Fe3}) / \text{gama};$$

$$\text{gradEqu}[204] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_FePr} * \text{S_act_Fe3} * 2.0) / \text{gama}^2};$$

$$\text{gradEqu}[205] = \frac{2.0}{\text{gama}^2 + (\text{KT_HCO3} * \text{S_act_H} * (1.0/2.0)) / \text{gama} + (\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama};$$

$$\text{gradEqu}[206] = \frac{2.0}{\text{gama}^2 + (\text{KT_HSO4} * \text{S_act_H} * (1.0/2.0)) / \text{gama} + (\text{KT_NH4SO4} * \text{S_act_NH4} * (1.0/2.0)) / \text{gama} + (\text{KT_FeSO4} * \text{S_act_Fe3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeS2O8} * \text{S_act_SO4} * \text{S_act_Fe3}) / \text{gama};$$

$$\text{gradEqu}[207] = \frac{2.0}{\text{gama}^2 + (\text{KT_FeOH} * (1.0/2.0)) / (\text{S_act_H} * \text{gama}) + (\text{KT_FeOH3} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeHS} * \text{S_act_HS} * (1.0/2.0)) / \text{gama} + (\text{KT_FeAc} * \text{S_act_Ac} * (1.0/2.0)) / \text{gama} + (\text{KT_FeHCO3} * \text{S_act_H} * \text{S_act_CO3} * (1.0/2.0)) / \text{gama} + (\text{KT_FeNH3} * \text{S_act_NH4} * 2.0) / (\text{S_act_H} * \text{gama}^2) + (\text{KT_FeN2H6} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4}) * 2.0) / \text{gama}^2 + (\text{KT_FeN3H9} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * 2.0) / \text{gama}^2 + (\text{KT_FeN4H12} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4} * \text{S_act_NH4}) * 2.0) / \text{gama}^2};$$

$$\text{gradEqu}[208] = \frac{(9.0/2.0)}{\text{gama}^3 + (\text{KT_FeOH2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeOH_2} * 2.0) / (\text{S_act_H} * \text{gama}^2) + (\text{KT_FeAc2} * (\text{S_act_Ac} * \text{S_act_Ac}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeS2O8} * (\text{S_act_SO4} * \text{S_act_SO4}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeNO2_2} * (\text{S_act_NO2} * \text{S_act_NO2}) * (1.0/2.0)) / \text{gama} + (\text{KT_FeNO2} * \text{S_act_NO2} * 2.0) / \text{gama}^2 + (\text{KT_FeSO4} * \text{S_act_SO4} * (1.0/2.0)) / \text{gama} + (\text{KT_FePr} * \text{S_act_Pr} * 2.0) / \text{gama}^2 + (\text{KT_FeAc_2} * \text{S_act_Ac} * 2.0) / \text{gama}^2 + (\text{KT_FeVa_2} * \text{S_act_Va} * 2.0) / \text{gama}^2 + (\text{KT_Fe2O2H2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3} * 1.6E1) / \text{gama}^4 + (\text{KT_Fe3O4H4} * 1.0 / (\text{S_act_H} * \text{S_act_H} * \text{S_act_H} * \text{S_act_H}) * (\text{S_act_Fe3} * \text{S_act_Fe3}) * (7.5E1/2.0)) / \text{gama}^5};$$

$$\text{gradEqu}[209] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_S} * 2.0) / (\text{S_act_H} * \text{gama}^2) + (\text{KT_FeHS} * \text{S_act_Fe2} * (1.0/2.0)) / \text{gama};$$

$$\text{gradEqu}[210] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_FeBu_2} * \text{S_act_Fe3} * 2.0) / \text{gama}^2};$$

$$\text{gradEqu}[211] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_FeVa_2} * \text{S_act_Fe3} * 2.0) / \text{gama}^2};$$

$$\text{gradEqu}[212] = \frac{(1.0/2.0)}{\text{gama}};$$

$$\text{gradEqu}[213] = \frac{(1.0/2.0)}{\text{gama} + (\text{KT_FeNO2} * \text{S_act_Fe3} * 2.0) / \text{gama}^2 + (\text{KT_FeNO2_2} * \text{S_act_NO2} * \text{S_act_Fe3}) / \text{gama};$$

$$\text{gradEqu}[214] = -1.0;$$

$$\text{gradEqu}[215] = \text{S_act_H} * 1.0 / (\text{gama} * \text{gama}) * (-1.0/2.0) - \text{S_act_NH4} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_HS} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_NO2} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_NO3} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Ac} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Bu} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Va} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{S_act_Pr} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - (\text{KT_H2O} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0)) / \text{S_act_H} - \text{KT_HCO3} * \text{S_act_H} * \text{S_act_CO3} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{KT_HSO4} * \text{S_act_H} * \text{S_act_SO4} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{KT_FeHS} * \text{S_act_HS} * \text{S_act_Fe2} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{KT_FeAc} * \text{S_act_Ac} * \text{S_act_Fe2} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{KT_NH4SO4} * \text{S_act_NH4} * \text{S_act_SO4} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - \text{KT_FeSO4} * \text{S_act_SO4} * \text{S_act_Fe3} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) - (\text{KT_FeOH} * \text{S_act_Fe2} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0)) / \text{S_act_H} - \text{KT_FeOH2} * 1.0 / (\text{S_act_H} * \text{S_act_H}) * \text{S_act_Fe3} * 1.0 / (\text{gama} * \text{gama}) * (1.0/2.0) -$$

```

KT_FeOH3*1.0/(S_act_H*S_act_H*S_act_H)*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0)-
KT_FeOH4*1.0/(S_act_H*S_act_H*S_act_H*S_act_H)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeAc2*(S_act_Ac*S_act_Ac)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeS2O8*(S_act_SO4*S_act_SO4)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeNO2_2*(S_act_NO2*S_act_NO2)*S_act_Fe3*1.0/(gama*gama)*(1.0/2.0)-
KT_FeHCO3*S_act_H*S_act_CO3*S_act_Fe2*1.0/(gama*gama)*(1.0/2.0);

```

```
// Eq15: gamma
```

```
gradEqu[216] = 0.0;
```

```
gradEqu[217] = 0.0;
```

```
gradEqu[218] = 0.0;
```

```
gradEqu[219] = 0.0;
```

```
gradEqu[220] = 0.0;
```

```
gradEqu[221] = 0.0;
```

```
gradEqu[222] = 0.0;
```

```
gradEqu[223] = 0.0;
```

```
gradEqu[224] = 0.0;
```

```
gradEqu[225] = 0.0;
```

```
gradEqu[226] = 0.0;
```

```
gradEqu[227] = 0.0;
```

```
gradEqu[228] = 0.0;
```

```

gradEqu[229] = pow(1.0E1,A*(mu*(3.0/1.0E1)-
sqrt(mu)/(sqrt(mu)+1.0)))*A*log(1.0E1)*(1.0/pow(sqrt(mu)+1.0,2.0)*(1.0/2.0)-
(1.0/sqrt(mu)*(1.0/2.0))/(sqrt(mu)+1.0)+3.0/1.0E1);

```

```
gradEqu[230] = -1.0;
```

```

for (i=0;i<15;i++){
    for (j=0;j<15;j++){
        jac[15*i+j]=gradEqu[i+15*j];transpose
    }
}
return status;

```

```

}

/*
*calculate initial states
*/
double CalcStaAnneal(double *x)
{
    int i, j, k, n, m, na, accept;
    double fun[15], MeanState[15], MaxState[15], xi[15], xc[15], xinorm[15], xcnorm[15];
    double F, Fc, p1, p50, T, T1, T50, frac, DeltaEavg, DeltaE, p, xrand, RandRange, range;
    bool status;

    //define seed for a random number generator
    srand((unsigned int)time(NULL));

    //random generator period
    RandRange=RAND_MAX;

    //mean values of states
    MeanState[0]=2.781e-9;    //x_H
    MeanState[1]=0.03815;    //x_NH4
    MeanState[2]=0.0056282;  //x_Ac
    MeanState[3]=0.0046166;  //x_Pr
    MeanState[4]=0.0010237;  //x_CO3
    MeanState[5]=0.00026465; //x_SO4
    MeanState[6]=3.1584e-5;  //x_Fe2
    MeanState[7]=9.3764e-15; //x_Fe3
    MeanState[8]=0.000106;   //x_HS
    MeanState[9]=0.0003839;  //x_Bu
    MeanState[10]=0.00037056; //x_Va
    MeanState[11]=0.00079054; //x_NO3
    MeanState[12]=0.00079472; //x_NO2
    MeanState[13]=0.10822;   // ionic strength
    MeanState[14]=0.77753;   // activity coefficient

    //maximum values of states divided by mean
    MaxState[0]=5.3969;    //x_H
    MaxState[1]=1.7769;    //x_NH4
    MaxState[2]=1.8509;    //x_Ac
    MaxState[3]=1.8944;    //x_Pr
    MaxState[4]=2.9678;    //x_CO3
    MaxState[5]=1.8566;    //x_SO4
    MaxState[6]=4.3015;    //x_Fe2
    MaxState[7]=11.54;     //x_Fe3
    MaxState[8]=5.4733;    //x_HS
    MaxState[9]=1.9311;    //x_Bu
    MaxState[10]=1.7687;   //x_Va
    MaxState[11]=1.9566;   //x_NO3
    MaxState[12]=1.8227;   //x_NO2
    MaxState[13]=1.3637;   // ionic strength
    MaxState[14]=1.0256;   // activity coefficient

    //range of trial ponts
    range=1e-3;

    //Simulated annealing algorithm
    //number of cycles
    n = 50;
    //number of trials per cycle
    m = 200;
    //number of accepted solutions
    na = 0;

```

```

// probability of accepting worse solution at the start
p1 = 0.7;
// probability of accepting worse solution at the end
p50 = 0.001;
// initial temperature
T1 = -1/log(p1);
// final temperature
T50 = -1/log(p50);
// fractional reduction every cycle
frac = pow((T50/T1),(1.0/(n-1.0)));

//read initial states
for (i = 0; i < 15; i++) {
    xc[i]=x[i];
    xcnorm[i]=xc[i]/MeanState[i]; //normalize states
}

//initialize counter of accepted solutions
na = na + 1;

//evaluate initial states
F=0;
status=FunEv(xc,15,fun);
for (i = 0; i < 15; i++) {
    F = F+fun[i]*fun[i];
}
Fc=F; //initial cost function

//Current temperature
T = T1;
//DeltaE Average
DeltaEavg = 0.0;

for (j = 0; j < n; j++) {
    for (k = 0; k < m; k++) {
        //Generate new trial points
        for (i = 0; i < 15; i++) {
            xrand=rand();
            xinorm[i]=xcnorm[i]+(xrand/RandRange-0.5)*range;

            //clip to upper and lower bounds
            if (xinorm[i] > MaxState[i]){
                xinorm[i] = MaxState[i];
            }
            if (xinorm[i] < 0){
                xinorm[i] = 0;
            }
        }
    }

    //denormalize states
    for (i = 0; i < 15; i++) {
        xi[i]=xinorm[i]*MeanState[i];
    }

    //evaluate new states
    F=0;
    status=FunEv(xi,15,fun);
    for (i = 0; i < 15; i++) {
        F = F+fun[i]*fun[i];
    }
}

```

```

DeltaE = fabs(F-Fc); ////calculate new DeltaE

if (F>Fc){
    // initialize DeltaE_avg if a worse solution was found
    // on the first iteration
    if ((j==0)&&(k==0)){
        DeltaEavg = DeltaE;
    }

    // objective function is worse
    // generate probability of acceptance
    p = exp(-DeltaE/(DeltaEavg*T));

    // determine whether to accept worse point
    xrand = rand();
    if ((xrand/RAND_MAX) < p){
        //accept the worse solution
        accept = 1;
    }
    else{
        //don't accept the worse solution
        accept = 0;
    }
}
else{
    //objective function is lower, automatically accept
    accept = 1;
}

if (accept==1){
    //update currently accepted solution
    for (i = 0; i < 15; i++) {
        xc[i]=xi[i];
        xcnorm[i]=xc[i]/MeanState[i]; //normalize states
    }
    //calculate the new best cost function
    F=0;
    status=FunEv(xc,15,fun);
    for (i = 0; i < 15; i++) {
        F = F+fun[i]*fun[i];
    }
    Fc=F;
    //increment number of accepted solutions
    na = na + 1;
    //update DeltaEavg
    DeltaEavg = (DeltaEavg*(na-1)+DeltaE)/na;
}
}
//lower the temperature for next cycle
T = frac * T;
}

for (i = 0; i < 15; i++) {
    x[i]=xc[i]; //return the the best states
}
return Fc; //return the minimum cost function
}

/*
 * mdlUpdate - perform action at major integration time step
 */
static void mdlUpdate(double *x, double *u, SimStruct *S, int tid)
{
    /*need to fix with malloc*/

```

```

double gres[15], gresold[15], jaca[231], xold[15], xnew[15], deltax[15]; //, testmat[100], testvec[10], testout[10];
double acc[15];
double Fmin;
double TOL;
int MaxSteps;
bool status, conv;
int i,j;

TOL = 1e-12;
MaxSteps= 10000;
j=1;
//This is function root position
//set globals so subfunctions can use them
Smain=S;
umain=u;

//transfer x & tol
for (i = 0; i < 15; i++) xnew[i]=x[i];
for (i = 0; i < 15; i++) acc[i]=TOL;
status=FunEv(xnew,15,gres);
conv=1;
for (i=0;i<15;i++) if (fabs(gres[i])>acc[i]) conv=0;
j=0;
//main loop
while ((!conv) && (j <= MaxSteps)){
    //Calculate the Jacobian
    status=JacAna(xnew,jaca);
    //Calculate residual
    status=FunEv(xnew,15,gres);
    //Calculate deltax
    MatDiv(15, jaca, gres, deltax);

    //Update x and store old values
    for (i = 0; i < 15; i++){
        xold[i]=xnew[i];
        gresold[i]=gres[i];
        xnew[i]=xold[i]-deltax[i]; //weighting should be changed.
    }

    conv=1;
    for (i=0;i<15;i++) {
        if (fabs(gres[i])>acc[i]) conv=0;
    }
}

//check if unstable results are reached
if (xnew[0] != xnew[0]){ //is xnew[0] NaN
    Fmin=CalcStaAnneal(x); //calculate states by using simulated annealing algorithm
    for (i = 0; i < 15; i++) xnew[i]=x[i];
}
else{
    if (conv)
    {
        for (i = 0; i < 15; i++){
            if (xnew[i] < 0){
                xnew[i] = 0; //set negative states to zero
            }
            x[i]=xnew[i];
        }
    }
    else status=0;
}
}

```

```

/*
 * mdlDerivatives - compute the derivatives
 */
static void mdlDerivatives(double *dx, double *x, double *u, SimStruct *S, int tid)
{
}

/*
 * mdlTerminate - called when the simulation is terminated.
 */
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE /* Is this file being compiled as a MEX-file? */
#include "simulink.c" /* MEX-file interface mechanism */
#else
#include "cg_sfun.h" /* Code generation registration function */
#endif

```

```

% This file initiates parameter values for Fe precipitation.
%
% Copyright 2006:
% Dr Christian Rosen, Dr Darko Vrecko and Dr Ulf Jeppsson
% Dept. Industrial Electrical Engineering and Automation
% Lund University, Sweden
% http://www.iea.lth.se
%
% Modified 2017:
% Ms. Natalie Greenway
% Dept. Chemical Engineering
% University of the Witwatersrand, South Africa
% m file

```

```

K_precipitation = 0.0; % no process information

```

```

=====
%
% Input measured values of total concentrations (mole/L)

```

```

%=====
%
Temp_C = 25;
%-----
%
% Input Crystallization kinetic data
%-----
%

% Iron Sulfide precipitation rate constant and effective order of reaction
k_fes = 1e3;
n_fes = 2;

KIN_PAR2 = [k_fes n_fes];

%=====
%
% Equilibrium constant at 25 oC (MINTEQ database in PHREEQC)
%=====
%

% -logK                Equilibrium constant K                Reaction
%=====
pK_FeSs = 2.95;        K_FeSs = 10^(-pK_FeSs);                % Iron Sulfide
%=====
%
% Heat of reaction (Joule/mole) at 25 oC (MINTEQ database in PHREEQC)
%=====
%

DH_FeSs = -11000;

%-----
%
% Gas constant
%-----
%
R = 8.314; % J/mole/K

%-----
%
% Calculate the temperature in Kelvin
%-----
%
Temp_K = Temp_C + 273.15;

%-----
%
% Calcualte the temperature dependence of the equilibrium constant using the Van't Hoff
expression.
%-----
%

if Temp_C == 25
    KT_FeSs = K_FeSs;
else
    KT_FeSs = 10^(log10(K_FeSs) - DH_FeSs/2.303/R*(1/(Temp_K) - 1/298.15));
end

PRECRX_PAR2 = [KT_FeSs];

/*
* adm1_ODE_bsm2.c is a C-file S-function for IAWQ AD Model No 1.
* In addition to the ADM1, temperature dependency and dummy states are added.
* Some details are adjusted for BSM2 (pH inhibition, gas flow output etc).
*
* C file – S function Sh2solv_synAMD
*
* Copyright (2006):

```

```

*
* Dr Christian Rosen, Dr Darko Vrecko and Dr Ulf Jeppsson
* Dept. Industrial Electrical Engineering and Automation (IEA)
* Lund University, Sweden
* http://www.iea.lth.se/
*
* Modified by:
* Natalie Greenway,, August 2017
* Dept. Chemical Engineering
* University of the Witwatersrand
*/

#define S_FUNCTION_NAME Sh2solv_synAMD

#include "simstruc.h"
#include <math.h>

#define XINIT ssGetArg(S,0)
#define PAR ssGetArg(S,1)
#define V ssGetArg(S,2)

/*
* mdlInitializeSizes - initialize the sizes array
*/
static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumContStates( S, 0); /* number of continuous states */
    ssSetNumDiscStates( S, 1); /* number of discrete states */
    ssSetNumInputs( S, 122); /* number of inputs */
    ssSetNumOutputs( S, 1); /* number of outputs */
    ssSetDirectFeedThrough(S, 0); /* direct feedthrough flag */
    ssSetNumSampleTimes( S, 1); /* number of sample times */
    ssSetNumSFcnParams( S, 3); /* number of input arguments */
    ssSetNumRWork( S, 0); /* number of real work vector elements */
    ssSetNumIWork( S, 0); /* number of integer work vector elements*/
    ssSetNumPWork( S, 0); /* number of pointer work vector elements*/
}

/*
* mdlInitializeSampleTimes - initialize the sample times array
*/
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, INHERITED_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0.0);
}

/*
* mdlInitializeConditions - initialize the states
*/
static void mdlInitializeConditions(double *x0, SimStruct *S)
{
    x0[0] = mxGetPr(XINIT)[0];
}

/*
* mdlOutputs - compute the outputs
*/
static void mdlOutputs(double *y, double *x, double *u, SimStruct *S, int tid)
{

```

```

y[0] = x[0]; /* state variable is passed on as output variable */
}

/*
 * mdlUpdate - perform action at major integration time step
 */
static void mdlUpdate(double *x, double *u, SimStruct *S, int tid)
{
    /* Newton-Raphson algorithm */

    double eps, f_h2_su, Y_su, f_h2_aa, Y_aa, Y_fa, Y_c4, Y_pro, K_S_IN, k_m_su, K_S_su, pH_UL_aa, pH_LL_aa,
    k_m_aa;
    double K_S_aa, k_m_fa, K_S_fa, K_lh2_fa, k_m_c4, K_S_c4, K_lh2_c4, k_m_pro, K_S_pro, K_lh2_pro;
    double pH_UL_ac, pH_LL_ac, k_m_h2, K_S_h2, pH_UL_h2, pH_LL_h2, R, T_base, T_op, kLa, K_H_h2,
    K_H_h2_base, V_liq, pH_op, l_pH_aa;
    double l_pH_h2, l_IN_lim, l_h2_fa, l_h2_c4, l_h2_pro, inhib[6];
    double proc5, proc6, proc7, proc8, proc9, proc10, proc12, proc20, proc34, proc30, procT8, reac8;
    double pHLim_aa, pHLim_h2, a_aa, a_h2, S_H_ion, n_aa, n_h2;
    double Equ;
    double gradEqu;

    // Additional states to deal with S //

    double S_su, S_aa, S_fa, S_va, S_bu, S_pro, S_ac, S_h2, S_ch4, S_IC, S_IN, S_I, X_xc, X_ch, X_pr, X_li, X_su,
    X_aa, X_fa, X_c4, X_pro, X_ac, X_h2, X_I, S_co2, S_nh3, S_h2s, S_so4, S_IS, X_IS, S_Fe2, S_Fe3;
    double S_gas_h2, S_gas_ch4, S_gas_co2, S_gas_h2s;
    double p_gas_h2, p_gas_ch4, p_gas_co2, p_gas_h2s, q_gas;

    double f_ac_bu, f_h2_bu, f_ac_pro, f_h2_pro, f_so4_pSRB, f_so4_aSRB, f_so4_hSRB;
    double Y_pSRB, Y_aSRB, Y_hSRB, k_m_pSRB, k_m_aSRB, k_m_hSRB, K_S_pSRB, K_S_aSRB, K_S_hSRB,
    K_S_so4_pSRB, K_S_so4_aSRB, K_S_so4_hSRB;
    double Y_c4SRB, k_m_c4SRB, K_S_c4SRB, K_S_so4_c4SRB;

    double K_l_h2s_c4, K_l_h2s_pro, K_l_h2s_ac, K_l_h2s_h2, K_l_h2s_pSRB, K_l_h2s_aSRB, K_l_h2s_hSRB;
    double pH_UL_pSRB, pH_LL_pSRB, pH_UL_aSRB, pH_LL_aSRB, pH_UL_hSRB, pH_LL_hSRB;
    double k_dec_XpSRB, k_dec_XaSRB, k_dec_XhSRB;
    double pH_UL_c4SRB, pH_LL_c4SRB, K_l_h2s_c4SRB, k_dec_Xc4SRB;

    double pK_a_so4_base, k_A_so4, k_A_h2s;
    double SRT, kLa_h2, kLa_ch4, kLa_h2s, kLa_co2;

    double pHLim_hSRB, n_hSRB, l_pH_hSRB, l_h2s_hSRB;
    double pHLim_aSRB, n_aSRB, l_pH_aSRB, l_h2s_aSRB;
    double pHLim_pSRB, n_pSRB, l_pH_pSRB, l_h2s_pSRB;
    double pHLim_c4SRB, n_c4SRB, l_pH_c4SRB, l_h2s_c4SRB;
    double X_hSRB, X_aSRB, X_pSRB, X_c4SRB;
    double l_h2s_h2, l_h2s_ac, l_h2s_pro, l_h2s_c4;

    double COD_Fe3, COD_Fe2, i_Fe3_Fe2, K1_Fe3_Fe2;
    double S_act_Fe3;

    double delta;
    double Sh20;
    int j;

    double TOL = 1e-12;
    double MaxSteps= 1000;

```

```

eps = 0.000001;
Sh20 = x[0]; /* Sh2 */

j = 1;
delta = 1.0;

while ( (delta > TOL || delta < -TOL) && (j <= MaxSteps) ) {

    while ( (delta > TOL || delta < -TOL) && (j <= MaxSteps) ) {

        f_h2_su = mxGetPr(PAR)[18];
        Y_su = mxGetPr(PAR)[27];
        f_h2_aa = mxGetPr(PAR)[28];
        Y_aa = mxGetPr(PAR)[34];
        Y_fa = mxGetPr(PAR)[35];
        Y_c4 = mxGetPr(PAR)[36];
        Y_pro = mxGetPr(PAR)[37];
        K_S_IN = mxGetPr(PAR)[45];
        k_m_su = mxGetPr(PAR)[46];
        K_S_su = mxGetPr(PAR)[47];
        pH_UL_aa = mxGetPr(PAR)[48];
        pH_LL_aa = mxGetPr(PAR)[49];
        k_m_aa = mxGetPr(PAR)[50];
        K_S_aa = mxGetPr(PAR)[51];
        k_m_fa = mxGetPr(PAR)[52];
        K_S_fa = mxGetPr(PAR)[53];
        K_lh2_fa = mxGetPr(PAR)[54];
        k_m_c4 = mxGetPr(PAR)[55];
        K_S_c4 = mxGetPr(PAR)[56];
        K_lh2_c4 = mxGetPr(PAR)[57];
        k_m_pro = mxGetPr(PAR)[58];
        K_S_pro = mxGetPr(PAR)[59];
        K_lh2_pro = mxGetPr(PAR)[60];
        pH_UL_ac = mxGetPr(PAR)[64];
        pH_LL_ac = mxGetPr(PAR)[65];
        k_m_h2 = mxGetPr(PAR)[66];
        K_S_h2 = mxGetPr(PAR)[67];
        pH_UL_h2 = mxGetPr(PAR)[68];
        pH_LL_h2 = mxGetPr(PAR)[69];
        R = mxGetPr(PAR)[77];
        T_base = mxGetPr(PAR)[78];
        T_op = mxGetPr(PAR)[79];
        kLa = mxGetPr(PAR)[94];
        K_H_h2_base = mxGetPr(PAR)[98];
        V_liq = mxGetPr(V)[0];

        // S extension

        Y_c4SRB = 0.0329; // Fedorovich & Kalyuzhnyi (2003)
        Y_pSRB = 0.0329; // Fedorovich & Kalyuzhnyi (2003)
        Y_aSRB = 0.00324; // Fedorovich & Kalyuzhnyi (2003)
        Y_hSRB = 0.0366; // Fedorovich & Kalyuzhnyi (2003)

        k_m_c4SRB = 20.0; // assumed as ADM1 * correction factor to outcompute X_va
        k_m_pSRB = 13.0; // assumed as ADM1 * correction factor to outcompute X_pro
        k_m_aSRB = 8.0; // assumed as ADM1 * correction factor to outcompute X_ac
        k_m_hSRB = 35.0*1.175; // assumed as ADM1 * correction factor to outcompute X_H2

        K_S_c4SRB = 0.20; // assumed as ADM1 * correction factor to outcompute X_va
        K_S_pSRB = 0.11; // assumed as ADM1 * correction factor to outcompute X_pro
        K_S_aSRB = 0.22; // assumed as ADM1 * correction factor to outcompute X_ac
        K_S_hSRB = 7.0e-06/1.175; // assumed as ADM1 * correction factor to outcompute X_H2

        K_S_so4_c4SRB = 0.00021; // Fedorovich & Kalyuzhnyi (2003)

```

```

K_S_so4_pSRB = 0.0002;      // FFedorovich & Kalyuzhnyi (2003)
K_S_so4_aSRB = 0.0001;      // Fedorovich & Kalyuzhnyi (2003)
K_S_so4_hSRB = 0.000104;     // Fedorovich & Kalyuzhnyi (2003)

K_l_h2s_c4 = 7.53e-3*64.0*1.5; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_pro = 7.53e-3*64.0*1.5; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_ac = 7.19e-3*64.0*0.75; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_h2 = 6.25e-3*64.0*0.75; // Fedorovich & Kalyuzhnyi (2003)

K_l_h2s_c4SRB = 8.13e-3*64.0*1.5; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_pSRB = 8.13e-3*64.0*1.5; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_aSRB = 7.80e-3*64.0*0.75; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_hSRB = 7.80e-3*64.0*0.75; // Fedorovich & Kalyuzhnyi (2003)

pH_UL_c4SRB = 7.0;           // Flores-Alsina et al. (2016)
pH_LL_c4SRB = 6.0;           // Flores-Alsina et al. (2016)
pH_UL_pSRB = 7.0;            // Flores-Alsina et al. (2016)
pH_LL_pSRB = 6.0;            // Flores-Alsina et al. (2016)
pH_UL_aSRB = 7.0;            // Flores-Alsina et al. (2016)
pH_LL_aSRB = 6.0;            // Flores-Alsina et al. (2016)
pH_UL_hSRB = 6.0;            // Flores-Alsina et al. (2016)
pH_LL_hSRB = 5.0;            // Flores-Alsina et al. (2016)

k_dec_Xc4SRB = 0.02;         // Batstone et al. (2002)
k_dec_XpSRB = 0.02;          // Batstone et al. (2002)
k_dec_XaSRB = 0.02;          // Batstone et al. (2002)
k_dec_XhSRB = 0.02;          // Batstone et al. (2002)

/// Fe extension

COD_Fe3 = 8.0;
COD_Fe2 = 0.0;
i_Fe3_Fe2 = 8.0-0.0;
K1_Fe3_Fe2 = 1.0e9;

K_H_h2 = K_H_h2_base*exp(-4180.0*(1.0/T_base - 1.0/T_op)/(100.0*R)); /* T adjustment for K_H_h2 */

pH_op = u[27]; /* pH */
S_H_ion = u[28]; /* SH+ */

S_su = u[0];
S_aa = u[1];
S_fa = u[2];
S_va = u[3];
S_bu = u[4];
S_pro = u[5];
S_ac = u[6];
S_h2 = x[0];
S_ch4 = u[8];
S_IC = u[9];
S_IN = u[10];
S_I = u[11];
X_xc = u[12];
X_ch = u[13];
X_pr = u[14];
X_li = u[15];
X_su = u[16];
X_aa = u[17];
X_fa = u[18];
X_c4 = u[19];
X_pro = u[20];
X_ac = u[21];

```

```

X_h2 = u[22];
X_I = u[23];

S_gas_h2 = u[29];
S_gas_ch4 = u[30];
S_gas_co2 = u[31];
S_gas_h2s = u[32];

```

```

S_so4 = u[39];
S_IS = u[40];
X_hSRB = u[41];
X_aSRB = u[42];
X_pSRB = u[43];
X_c4SRB = u[44];

```

```

S_Fe2 = u[46];
S_Fe3 = u[47];

S_h2s = u[105];

```

/ Hill function on SH+ used within BSM2, ADM1 Workshop, Copenhagen 2005. */*

```

pHLim_aa = pow(10,(-(pH_UL_aa + pH_LL_aa)/2.0));
pHLim_h2 = pow(10,(-(pH_UL_h2 + pH_LL_h2)/2.0));
n_aa = 3.0/(pH_UL_aa-pH_LL_aa);
n_h2 = 3.0/(pH_UL_h2-pH_LL_h2);
l_pH_aa = pow(pHLim_aa,n_aa)/(pow(S_H_ion,n_aa)+pow(pHLim_aa ,n_aa));
l_pH_h2 = pow(pHLim_h2,n_h2)/(pow(S_H_ion,n_h2)+pow(pHLim_h2 ,n_h2));

```

```

l_IN_lim = 1.0/(1.0+K_S_IN/(u[10]));
l_h2_fa = 1.0/(1.0+x[0]/K_lh2_fa);
l_h2_c4 = 1.0/(1.0+x[0]/K_lh2_c4);
l_h2_pro = 1.0/(1.0+x[0]/K_lh2_pro);

```

// eliminated l_IP_lim

```

inhib[0] = l_pH_aa*l_IN_lim;
inhib[1] = inhib[0]*l_h2_fa;
inhib[2] = inhib[0]*l_h2_c4;
inhib[3] = inhib[0]*l_h2_pro;
inhib[5] = l_pH_h2*l_IN_lim;

```

```

pHLim_hSRB = pow(10,(-(pH_UL_hSRB + pH_LL_hSRB)/2.0));
pHLim_aSRB = pow(10,(-(pH_UL_aSRB + pH_LL_aSRB)/2.0));
pHLim_pSRB = pow(10,(-(pH_UL_pSRB + pH_LL_pSRB)/2.0));
pHLim_c4SRB = pow(10,(-(pH_UL_c4SRB + pH_LL_c4SRB)/2.0));

```

```

n_hSRB = 3.0/(pH_UL_hSRB-pH_LL_hSRB);
n_aSRB = 3.0/(pH_UL_aSRB-pH_LL_aSRB);
n_pSRB = 3.0/(pH_UL_pSRB-pH_LL_pSRB);
n_c4SRB = 3.0/(pH_UL_c4SRB-pH_LL_c4SRB);

```

```

l_pH_hSRB = pow(pHLim_hSRB,n_hSRB)/(pow(S_H_ion,n_hSRB)+pow(pHLim_hSRB ,n_hSRB));
l_pH_aSRB = pow(pHLim_aSRB,n_aSRB)/(pow(S_H_ion,n_aSRB)+pow(pHLim_aSRB ,n_aSRB));
l_pH_pSRB = pow(pHLim_pSRB,n_pSRB)/(pow(S_H_ion,n_pSRB)+pow(pHLim_pSRB ,n_pSRB));
l_pH_c4SRB = pow(pHLim_c4SRB,n_c4SRB)/(pow(S_H_ion,n_c4SRB)+pow(pHLim_c4SRB ,n_c4SRB));

```

```

l_h2s_hSRB = 1/(1+(S_h2s)/K_l_h2s_hSRB);
l_h2s_aSRB = 1/(1+(S_h2s)/K_l_h2s_aSRB);
l_h2s_pSRB = 1/(1+(S_h2s)/K_l_h2s_pSRB);
l_h2s_c4SRB = 1/(1+(S_h2s)/K_l_h2s_c4SRB);

```

```

l_h2s_h2 = 1/(1+(S_h2s)/K_l_h2s_h2);
l_h2s_c4 = 1/(1+(S_h2s)/K_l_h2s_c4);
l_h2s_pro = 1/(1+(S_h2s)/K_l_h2s_pro);
l_h2s_ac = 1/(1+(S_h2s)/K_l_h2s_ac);

proc5 = k_m_su*S_su/(K_S_su+S_su)*X_su*inhib[0];
proc6 = k_m_aa*S_aa/(K_S_aa+S_aa)*X_aa*inhib[0];
proc7 = k_m_fa*S_fa/(K_S_fa+S_fa)*X_fa*inhib[1];
proc8 = k_m_c4*S_va/(K_S_c4+S_va)*X_c4*S_va/(S_va+S_bu+eps)*inhib[2]*l_h2s_c4;
proc9 = k_m_c4*S_bu/(K_S_c4+S_bu)*X_c4*S_bu/(S_va+S_bu+eps)*inhib[2]*l_h2s_c4;
proc10 = k_m_pro*S_pro/(K_S_pro+S_pro)*X_pro*inhib[3]*l_h2s_pro;
proc12 = k_m_h2*(S_h2)/(K_S_h2+(S_h2))*X_h2*inhib[5]*l_h2s_h2;

proc20 =
1.0*k_m_hSRB*S_h2/(K_S_hSRB+S_h2)*(S_so4/(K_S_so4_hSRB+S_so4))*X_hSRB*L_pH_hSRB*l_h2s_hSRB*l_IN_lim
;

proc30 = 1.0*K1_Fe3_Fe2*S_Fe3*S_h2;

p_gas_h2 = S_gas_h2*R*T_op/16.0;
procT8 = kLa*(S_h2-16.0*K_H_h2*p_gas_h2);

reac8 = (1.0-Y_su)*f_h2_su*proc5+(1.0-Y_aa)*f_h2_aa*proc6+(1.0-Y_fa)*0.3*proc7+(1.0-
Y_c4)*0.15*proc8+(1.0-Y_c4)*0.2*proc9+(1.0-Y_pro)*0.43*proc10 -proc12 - proc20 - proc30 -procT8;

/* Sh2 equation */

Equ = reac8;

/* Gradient of Sh2 equation */

gradEqu = -kLa-K1_Fe3_Fe2*S_Fe3-
(l_h2s_h2*X_h2*inhib[5]*k_m_h2)/(K_S_h2+S_h2)+l_h2s_h2*S_h2*X_h2*inhib[5]*k_m_h2*1.0/pow(K_S_h2+S_h2,2.0)-
(l_h2s_hSRB*L_pH_hSRB*S_so4*X_hSRB*k_m_hSRB)/((K_S_hSRB+S_h2)*(K_S_so4_hSRB+S_so4))+l_h2s_hSRB*L_
pH_hSRB*S_h2*S_so4*X_hSRB*k_m_hSRB*1.0/pow(K_S_hSRB+S_h2,2.0))/(K_S_so4_hSRB+S_so4)+(S_fa*X_fa*inhi
b[0]*k_m_fa*(Y_fa*(3.0/1.0E1)-
3.0/1.0E1)*1.0/pow(S_h2/K_lh2_fa+1.0,2.0))/(K_lh2_fa*(K_S_fa+S_fa))+l_h2s_pro*S_pro*X_pro*inhib[0]*k_m_pro*(Y_p
ro*(4.3E1/1.0E2)-
4.3E1/1.0E2)*1.0/pow(S_h2/K_lh2_pro+1.0,2.0))/(K_lh2_pro*(K_S_pro+S_pro))+l_h2s_c4*(S_bu*S_bu)*X_c4*inhib[0]*k
_m_c4*(Y_c4*(1.0/5.0)-
1.0/5.0)*1.0/pow(S_h2/K_lh2_c4+1.0,2.0))/(K_lh2_c4*(K_S_c4+S_bu)*(S_bu+S_va+eps))+l_h2s_c4*(S_va*S_va)*X_c4
*inhib[0]*k_m_c4*(Y_c4*(3.0/2.0E1)-
3.0/2.0E1)*1.0/pow(S_h2/K_lh2_c4+1.0,2.0))/(K_lh2_c4*(K_S_c4+S_va)*(S_bu+S_va+eps));

/*iteration loop*/

delta = Equ;
x[0] = Sh20-delta/gradEqu; /* Calculate the new Sh2 */

if (x[0] <= 0) {
    x[0] = 1e-12; /* to avoid numerical problems */
}

Sh20 = x[0];
++j;
}
}
}
/*
* mdlDerivatives - compute the derivatives
*/

```

```

static void mdlDerivatives(double *dx, double *x, double *u, SimStruct *S, int tid)
{

}

/*
 * mdlTerminate - called when the simulation is terminated.
 */
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE /* Is this file being compiled as a MEX-file? */
#include "simulink.c" /* MEX-file interface mechanism */
#else
#include "cg_sfun.h" /* Code generation registration function */
#endif

/*
 * adm1_ODE_bsm2.c is a C-file S-function for IAWQ AD Model No 1.
 * In addition to the ADM1, temperature dependency and dummy states are added.
 * Some details are adjusted for BSM2 (pH inhibition, gas flow output etc).
 * In this the model derivatives of ion states and hydrogen gas are set to zero.
 * Instead they are calculated using algebraic equations and a pH solver
 * using the Newton-Raphson method and an equivalent H2-solver). This way the main stiffness is removed.
 * In addition to the ADM1, temperature dependency and dummy states are added.
 * Some details are adjusted for BSM2 (pH inhibition, gas flow output etc).
 *
 * C file – S function adm1_DAE2_synAMD
 *
 * Copyright (2006):
 * Dr Christian Rosen, Dr Darko Vrecko and Dr Ulf Jeppsson
 * Dept. Industrial Electrical Engineering and Automation (IEA)
 * Lund University, Sweden
 * http://www.iea.lth.se/
 *
 * Modified by Xavier Flores-Alsina
 * May 2013
 *
 * Modified by Natalie Greenway
 * University of the Witwatersand
 * August 2017
 */

#define S_FUNCTION_NAME adm1_DAE2_synAMD

#include "simstruc.h"
#include <math.h>

#define XINIT      ssGetArg(S,0) /* Initial conditions - adm1int_AMD */
#define PAR        ssGetArg(S,1) /* Parameters - adm1int_AMD */
#define V          ssGetArg(S,2) /* Volume - adm1int_AMD */

```

```

#define KIN_PAR          ssGetArg(S,3)    /* chemical parameters - precipitationint_adm1_AMD */
#define PRECRX_PAR       ssGetArg(S,4)    /* chemical parameters - precipitationint_adm1_AMD */
#define CHEM_PAR         ssGetArg(S,5)    /* chemical parameters - speciationint_adm1_amd*/
#define BIOS             ssGetArg(S,6)    /* INT_BSM2 */
#define ALGEBRAIC        ssGetArg(S,7)    /* INT_BSM2 */

/*
 * mdlInitializeSizes - initialize the sizes array
 */
static void mdlInitializeSizes(SimStruct *S)
{
    ssSetNumContStates( S, 38); /* number of continuous states */
    ssSetNumDiscStates( S, 0);  /* number of discrete states */
    ssSetNumInputs(     S, 110); /* number of inputs */
    ssSetNumOutputs(    S, 49);  /* number of outputs */
    ssSetDirectFeedThrough(S, 1); /* direct feedthrough flag */
    ssSetNumSampleTimes( S, 1);  /* number of sample times */
    ssSetNumSFcnParams(  S, 8);  /* number of input arguments */
    ssSetNumRWork(       S, 0);  /* number of real work vector elements */
    ssSetNumIWork(       S, 0);  /* number of integer work vector elements*/
    ssSetNumPWork(       S, 0);  /* number of pointer work vector elements*/
}

/*
 * mdlInitializeSampleTimes - initialize the sample times array
 */
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, CONTINUOUS_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0.0);
}

/*
 * mdlInitializeConditions - initialize the states
 */
static void mdlInitializeConditions(double *x0, SimStruct *S)
{
    int i;

    for (i = 0; i < 38; i++) {
        x0[i] = mxGetPr(XINIT)[i];
    }
}

/*
 * mdlOutputs - compute the outputs
 */
static void mdlOutputs(double *y, double *x, double *u, SimStruct *S, int tid)
{
    double R, T_base, T_op;
    double P_atm, p_gas_h2o, P_gas, k_P, q_gas;
    double K_H_h2o_base;
    double S_gas_h2, S_gas_ch4, S_gas_co2, S_gas_h2s;
    double p_gas_h2, p_gas_ch4, p_gas_co2, p_gas_h2s;
    double xtemp[38];
    double algebraic;

    int i,j;

    R      = mxGetPr(PAR)[77];
    T_base = mxGetPr(PAR)[78];

```

```

T_op      = mxGetPr(PAR)[79];
P_atm     = mxGetPr(PAR)[93];
K_H_h2o_base = mxGetPr(PAR)[95];
k_P       = mxGetPr(PAR)[99];
algebraic = mxGetPr(ALGEBRAIC)[0];

for (j = 0; j < 38; j++) {
    if (x[j] < 1.0e-8)
        xtemp[j] = 0.0;
    else
        xtemp[j] = x[j];
}

for (i = 0; i < 7; i++) {
    y[i] = xtemp[i];
}

if (algebraic < 0.5)
    y[7] = x[7]; /* Sh2 */
else
    y[7] = u[39]; /* Sh2 */

for (i = 0; i < 16; i++) {
    y[i+8] = xtemp[i+8];
}

y[24] = u[24]; /* flow */

y[25] = T_op - 273.15; /* Temp = 35 degC */

y[26] = u[7]; /* H2 */

y[27] = -log10(u[39]); /* pH */
y[28] = u[39]; /* SH+ */

y[29] = xtemp[24]; /* Sh2 */
y[30] = xtemp[25]; /* Sch4 */
y[31] = xtemp[26]; /* Sco2 */
y[32] = xtemp[27]; /* D2 - H2S */

S_gas_h2 = xtemp[24];
S_gas_ch4 = xtemp[25];
S_gas_co2 = xtemp[26];
S_gas_h2s = xtemp[27];

p_gas_h2 = S_gas_h2*R*T_op/16.0;
p_gas_ch4 = S_gas_ch4*R*T_op/64.0;
p_gas_co2 = S_gas_co2*R*T_op;
p_gas_h2s = S_gas_h2s*R*T_op/64.0;
P_gas = p_gas_h2 + p_gas_ch4 + p_gas_co2 + p_gas_h2s + p_gas_h2o;

y[33] = p_gas_h2;
y[34] = p_gas_ch4;
y[35] = p_gas_co2;
y[36] = p_gas_h2s;

q_gas = k_P*(P_gas - P_atm);
if (q_gas < 0)
    q_gas = 0.0;

y[37] = P_gas; /* total head space pressure from H2, CH4, CO2 and H2O */

```

```

y[38] = q_gas * P_gas/P_atm; /* The output gas flow is recalculated to atmospheric pressure (normalization) */

/* Bio S equations */

for (i = 0; i < 7; i++) {
    y[i+39] = xtemp[28+i];
}

/* Scat+ and San- */

for (i = 0; i < 2; i++) {
    y[i+46] = xtemp[35+i];
}

/* Multiple minerals */

for (i = 0; i < 1; i++) {
    y[i+48] = xtemp[37+i];
}

/* Ends on 48, therefore there are 49 outputs. */

}

/*
 * mdlUpdate - perform action at major integration time step
 */
static void mdlUpdate(double *x, double *u, SimStruct *S, int tid)
{
}

/*
 * mdlDerivatives - compute the derivatives
 */
static void mdlDerivatives(double *dx, double *x, double *u, SimStruct *S, int tid)
{
    double f_sl_xc, f_xl_xc, f_ch_xc, f_pr_xc, f_li_xc, N_xc, N_l, N_aa, C_xc, C_sl, C_ch;
    double C_pr, C_li, C_xl, C_su, C_aa, f_fa_li, C_fa, f_h2_su, f_bu_su, f_pro_su, f_ac_su;
    double N_bac, C_bu, C_pro, C_ac, C_bac, Y_su, f_h2_aa, f_va_aa, f_bu_aa, f_pro_aa, f_ac_aa;
    double C_va, Y_aa, Y_fa, Y_c4, Y_pro, C_ch4, Y_ac, Y_h2;
    double k_dis, k_hyd_ch, k_hyd_pr, k_hyd_li, K_S_IN, k_m_su, K_S_su, pH_UL_aa, pH_LL_aa;
    double k_m_aa, K_S_aa, k_m_fa, K_S_fa, K_lh2_fa, k_m_c4, K_S_c4, K_lh2_c4, k_m_pro, K_S_pro;
    double K_lh2_pro, k_m_ac, K_S_ac, K_lnh3, pH_UL_ac, pH_LL_ac, k_m_h2, K_S_h2, pH_UL_h2, pH_LL_h2;
    double k_dec_Xsu, k_dec_Xaa, k_dec_Xfa, k_dec_Xc4, k_dec_Xpro, k_dec_Xac, k_dec_Xh2;
    double R, T_base, T_op, pK_w_base, pK_a_va_base, pK_a_bu_base, pK_a_pro_base, pK_a_ac_base,
    pK_a_co2_base, pK_a_IN_base;
    double K_w, K_a_va, K_a_bu, K_a_pro, K_a_ac, K_a_co2, K_a_IN, K_H_co2, K_H_ch4, K_H_h2;
    double K_A_Bva, K_A_Bbu, K_A_Bpro, K_A_Bac, K_A_Bco2, K_A_BIN;
    double P_atm, p_gas_h2o, P_gas, k_P, kLa, K_H_h2o_base, K_H_co2_base, K_H_ch4_base, K_H_h2_base, factor;
    double V_liq, V_gas;
    double eps, pH_op, phi, S_H_ion;

    double proc1, proc2, proc3, proc4, proc5, proc6, proc7, proc8, proc9;
    double proc10, proc11, proc12, proc13, proc14, proc15, proc16, proc17, proc18, proc19;
    double proc20, proc21, proc22, proc23, proc24, proc25, proc26, proc27, proc28, proc29;
    double proc30, proc31, proc32;

    double procT8, procT9, procT10, procT11;

```

```

double l_pH_aa, l_pH_ac, l_pH_h2, l_IN_lim, l_h2_fa, l_h2_c4, l_h2_pro, l_nh3;

double reac1, reac2, reac3, reac4, reac5, reac6, reac7, reac8, reac9, reac10;
double reac11, reac12, reac13, reac14, reac15, reac16, reac17, reac18, reac19, reac20;
double reac21, reac22, reac23, reac24, reac25, reac26, reac27, reac28, reac29, reac30;
double reac31, reac32, reac33;

double N_pr;

double stoich1_c, stoich2_c, stoich3_c, stoich4_c, stoich5_c, stoich6_c, stoich7_c, stoich8_c, stoich9_c, stoich10_c,
stoich11_c, stoich12_c, stoich13_c, stoich14_c, stoich15_c, stoich16_c, stoich17_c, stoich18_c, stoich19_c, stoich20_c,
stoich21_c, stoich22_c, stoich23_c;
double stoich1_n, stoich2_n, stoich3_n, stoich4_n, stoich5_n, stoich6_n, stoich7_n, stoich8_n, stoich9_n, stoich10_n,
stoich11_n, stoich12_n, stoich13_n, stoich14_n, stoich15_n, stoich16_n, stoich17_n, stoich18_n, stoich19_n,
stoich20_n, stoich21_n, stoich22_n, stoich23_n;
double stoich1_s, stoich2_s, stoich3_s, stoich4_s, stoich5_s, stoich6_s, stoich7_s, stoich8_s, stoich9_s, stoich10_s,
stoich11_s, stoich12_s, stoich13_s, stoich14_s, stoich15_s, stoich16_s, stoich17_s, stoich18_s, stoich19_s, stoich20_s,
stoich21_s, stoich22_s, stoich23_s;

double xtemp[38], inhib[6]; // need to check inhib[6]

double S_act_H, S_act_NH4, S_act_Ac, S_act_Pr, S_act_CO3, S_act_SO4, S_act_Fe2, S_act_Fe3, S_act_HS,
S_act_Bu, S_act_Va, S_act_NO3, S_act_NO2, gama, mu;
double S_conc_H, S_conc_NH4, S_conc_Ac, S_conc_Pr, S_conc_CO3, S_conc_SO4, S_conc_Fe2, S_conc_Fe3,
S_conc_HS, S_conc_Bu, S_conc_Va, S_conc_NO3, S_conc_NO2;
double S_act_S;
double S_su, S_aa, S_fa, S_va, S_bu, S_pro, S_ac, S_h2, S_ch4, S_IC, S_IN, S_I, X_xc, X_ch, X_pr, X_li, X_su,
X_aa, X_fa, X_c4, X_pro, X_ac, X_h2, X_I, S_co2, S_nh3, S_h2s, S_so4, S_IS, X_IS, S_Fe2, S_Fe3;
double S_gas_h2, S_gas_ch4, S_gas_co2, S_gas_h2s;
double p_gas_h2, p_gas_ch4, p_gas_co2, p_gas_h2s, q_gas;

double pHLim_aa, pHLim_ac, pHLim_h2, a_aa, a_ac, a_h2, n_aa, n_ac, n_h2;

// Bio S module

double Y_c4SRB, Y_pSRB, Y_aSRB, Y_hSRB;
double k_m_c4SRB, k_m_pSRB, k_m_aSRB, k_m_hSRB;
double K_S_c4SRB, K_S_pSRB, K_S_aSRB, K_S_hSRB;
double K_S_so4_c4SRB, K_S_so4_pSRB, K_S_so4_aSRB, K_S_so4_hSRB;
double K_I_h2s_c4, K_I_h2s_pro, K_I_h2s_ac, K_I_h2s_h2;
double K_I_h2s_c4SRB, K_I_h2s_pSRB, K_I_h2s_aSRB, K_I_h2s_hSRB;
double pH_UL_c4SRB, pH_LL_c4SRB, pH_UL_pSRB, pH_LL_pSRB, pH_UL_aSRB, pH_LL_aSRB, pH_UL_hSRB,
pH_LL_hSRB;
double k_dec_Xc4SRB, k_dec_XpSRB, k_dec_XaSRB, k_dec_XhSRB;
double K_H_h2s_base, K_H_h2s;
double pHLim_hSRB, n_hSRB, l_pH_hSRB, l_h2s_hSRB;
double pHLim_aSRB, n_aSRB, l_pH_aSRB, l_h2s_aSRB;
double pHLim_pSRB, n_pSRB, l_pH_pSRB, l_h2s_pSRB;
double pHLim_c4SRB, n_c4SRB, l_pH_c4SRB, l_h2s_c4SRB;
double X_hSRB, X_aSRB, X_pSRB, X_c4SRB;
double l_h2s_h2, l_h2s_ac, l_h2s_pro, l_h2s_c4;
double Bio_S;

// Fe module

double COD_Fe3, COD_Fe2, l_Fe3_Fe2, K1_Fe3_Fe2, K2_Fe3_Fe2;

// Precipitation Module

double k_fes, n_fes;
double KT_H2O, KT_S, KT_FeSs;
double SI_fes;
double P_fes;

```

```

double X_FeS;

// solving method

double algebraic;

int i;

eps = 0.000001;

f_sl_xc    = mxGetPr(PAR)[0];
f_xl_xc    = mxGetPr(PAR)[1];
f_ch_xc    = mxGetPr(PAR)[2];
f_pr_xc    = mxGetPr(PAR)[3];
f_li_xc    = mxGetPr(PAR)[4];
N_xc       = mxGetPr(PAR)[5];
N_l        = mxGetPr(PAR)[6];
N_aa       = mxGetPr(PAR)[7];
C_xc       = mxGetPr(PAR)[8];
C_sl       = mxGetPr(PAR)[9];
C_ch       = mxGetPr(PAR)[10];
C_pr       = mxGetPr(PAR)[11];
C_li       = mxGetPr(PAR)[12];
C_xl       = mxGetPr(PAR)[13];
C_su       = mxGetPr(PAR)[14];
C_aa       = mxGetPr(PAR)[15];
f_fa_li    = mxGetPr(PAR)[16];
C_fa       = mxGetPr(PAR)[17];
f_h2_su    = mxGetPr(PAR)[18];
f_bu_su    = mxGetPr(PAR)[19];
f_pro_su   = mxGetPr(PAR)[20];
f_ac_su    = mxGetPr(PAR)[21];
N_bac      = mxGetPr(PAR)[22];
C_bu       = mxGetPr(PAR)[23];
C_pro      = mxGetPr(PAR)[24];
C_ac       = mxGetPr(PAR)[25];
C_bac      = mxGetPr(PAR)[26];
Y_su       = mxGetPr(PAR)[27];
f_h2_aa    = mxGetPr(PAR)[28];
f_va_aa    = mxGetPr(PAR)[29];
f_bu_aa    = mxGetPr(PAR)[30];
f_pro_aa   = mxGetPr(PAR)[31];
f_ac_aa    = mxGetPr(PAR)[32];
C_va       = mxGetPr(PAR)[33];
Y_aa       = mxGetPr(PAR)[34];
Y_fa       = mxGetPr(PAR)[35];
Y_c4       = mxGetPr(PAR)[36];
Y_pro      = mxGetPr(PAR)[37];
C_ch4      = mxGetPr(PAR)[38];
Y_ac       = mxGetPr(PAR)[39];
Y_h2       = mxGetPr(PAR)[40];
k_dis      = mxGetPr(PAR)[41];
k_hyd_ch   = mxGetPr(PAR)[42];
k_hyd_pr   = mxGetPr(PAR)[43];
k_hyd_li   = mxGetPr(PAR)[44];
K_S_IN     = mxGetPr(PAR)[45];
k_m_su     = mxGetPr(PAR)[46];
K_S_su     = mxGetPr(PAR)[47];
pH_UL_aa   = mxGetPr(PAR)[48];
pH_LL_aa   = mxGetPr(PAR)[49];
k_m_aa     = mxGetPr(PAR)[50];
K_S_aa     = mxGetPr(PAR)[51];
k_m_fa     = mxGetPr(PAR)[52];
K_S_fa     = mxGetPr(PAR)[53];

```

```

K_lh2_fa    = mxGetPr(PAR)[54];
k_m_c4      = mxGetPr(PAR)[55];
K_S_c4      = mxGetPr(PAR)[56];
K_lh2_c4    = mxGetPr(PAR)[57];
k_m_pro     = mxGetPr(PAR)[58];
K_S_pro     = mxGetPr(PAR)[59];
K_lh2_pro   = mxGetPr(PAR)[60];
k_m_ac      = mxGetPr(PAR)[61];
K_S_ac      = mxGetPr(PAR)[62];
K_l_nh3     = mxGetPr(PAR)[63];
pH_UL_ac    = mxGetPr(PAR)[64];
pH_LL_ac    = mxGetPr(PAR)[65];
k_m_h2      = mxGetPr(PAR)[66];
K_S_h2      = mxGetPr(PAR)[67];
pH_UL_h2    = mxGetPr(PAR)[68];
pH_LL_h2    = mxGetPr(PAR)[69];
k_dec_Xsu   = mxGetPr(PAR)[70];
k_dec_Xaa   = mxGetPr(PAR)[71];
k_dec_Xfa   = mxGetPr(PAR)[72];
k_dec_Xc4   = mxGetPr(PAR)[73];
k_dec_Xpro  = mxGetPr(PAR)[74];
k_dec_Xac   = mxGetPr(PAR)[75];
k_dec_Xh2   = mxGetPr(PAR)[76];
R           = mxGetPr(PAR)[77];
T_base      = mxGetPr(PAR)[78];
T_op        = mxGetPr(PAR)[79];
pK_w_base   = mxGetPr(PAR)[80];
pK_a_va_base = mxGetPr(PAR)[81];
pK_a_bu_base = mxGetPr(PAR)[82];
pK_a_pro_base = mxGetPr(PAR)[83];
pK_a_ac_base = mxGetPr(PAR)[84];
pK_a_co2_base = mxGetPr(PAR)[85];
pK_a_IN_base = mxGetPr(PAR)[86];
K_A_Bva     = mxGetPr(PAR)[87];
K_A_Bbu     = mxGetPr(PAR)[88];
K_A_Bpro    = mxGetPr(PAR)[89];
K_A_Bac     = mxGetPr(PAR)[90];
K_A_Bco2    = mxGetPr(PAR)[91];
K_A_BIN     = mxGetPr(PAR)[92];
P_atm       = mxGetPr(PAR)[93];
kLa         = mxGetPr(PAR)[94];
K_H_h2o_base = mxGetPr(PAR)[95];
K_H_co2_base = mxGetPr(PAR)[96];
K_H_ch4_base = mxGetPr(PAR)[97];
K_H_h2_base = mxGetPr(PAR)[98];
k_P         = mxGetPr(PAR)[99];

V_liq       = mxGetPr(V)[0];
V_gas       = mxGetPr(V)[1];

// Nitrogen extension

N_pr        = mxGetPr(PAR)[100];
// S extensions //

Bio_S       = mxGetPr(BIOS)[0];

Y_c4SRB     = mxGetPr(PAR)[101];
Y_pSRB      = mxGetPr(PAR)[102];
Y_aSRB      = mxGetPr(PAR)[103];
Y_hSRB      = mxGetPr(PAR)[104];

Y_m_c4SRB   = mxGetPr(PAR)[105];

// Fedorovich & Kalyuzhnyi (2003)
// Fedorovich & Kalyuzhnyi (2003)
// Fedorovich & Kalyuzhnyi (2003)
// Fedorovich & Kalyuzhnyi (2003)

// assumed as ADM1 * correction factor to outcompute X_va

```

```

k_m_pSRB = mxGetPr(PAR)[106]; // assumed as ADM1 * correction factor to outcompute X_pro
k_m_aSRB = mxGetPr(PAR)[107]; // assumed as ADM1 * correction factor to outcompute X_ac
k_m_hSRB = mxGetPr(PAR)[108]; // assumed as ADM1 * correction factor to outcompute X_H2

K_S_c4SRB = mxGetPr(PAR)[109]; // assumed as ADM1 * correction factor to outcompute X_va
K_S_pSRB = mxGetPr(PAR)[110]; // assumed as ADM1 * correction factor to outcompute X_pro
K_S_aSRB = mxGetPr(PAR)[111]; // assumed as ADM1 * correction factor to outcompute X_ac
K_S_hSRB = mxGetPr(PAR)[112]; // assumed as ADM1 * correction factor to outcompute X_H2

K_S_so4_c4SRB = mxGetPr(PAR)[113]; // Fedorovich & Kalyuzhnyi (2003)
K_S_so4_pSRB = mxGetPr(PAR)[114]; // Fedorovich & Kalyuzhnyi (2003)
K_S_so4_aSRB = mxGetPr(PAR)[115]; // Fedorovich & Kalyuzhnyi (2003)
K_S_so4_hSRB = mxGetPr(PAR)[116]; // Fedorovich & Kalyuzhnyi (2003)

K_l_h2s_c4 = mxGetPr(PAR)[117]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_pro = mxGetPr(PAR)[118]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_ac = mxGetPr(PAR)[119]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_h2 = mxGetPr(PAR)[120]; // Fedorovich & Kalyuzhnyi (2003)

K_l_h2s_c4SRB = mxGetPr(PAR)[121]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_pSRB = mxGetPr(PAR)[122]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_aSRB = mxGetPr(PAR)[123]; // Fedorovich & Kalyuzhnyi (2003)
K_l_h2s_hSRB = mxGetPr(PAR)[124]; // Fedorovich & Kalyuzhnyi (2003)

pH_UL_c4SRB = mxGetPr(PAR)[125]; // Flores-Alsina et al. (2016)
pH_LL_c4SRB = mxGetPr(PAR)[126]; // Flores-Alsina et al. (2016)
pH_UL_pSRB = mxGetPr(PAR)[127]; // Flores-Alsina et al. (2016)
pH_LL_pSRB = mxGetPr(PAR)[128]; // Flores-Alsina et al. (2016)
pH_UL_aSRB = mxGetPr(PAR)[129]; // Flores-Alsina et al. (2016)
pH_LL_aSRB = mxGetPr(PAR)[130]; // Flores-Alsina et al. (2016)
pH_UL_hSRB = mxGetPr(PAR)[131]; // Flores-Alsina et al. (2016)
pH_LL_hSRB = mxGetPr(PAR)[132]; // Flores-Alsina et al. (2016)

k_dec_Xc4SRB = mxGetPr(PAR)[133]; // Batstone et al. (2002)
k_dec_XpSRB = mxGetPr(PAR)[134]; // Batstone et al. (2002)
k_dec_XaSRB = mxGetPr(PAR)[135]; // Batstone et al. (2002)
k_dec_XhSRB = mxGetPr(PAR)[136]; // Batstone et al. (2002)

K_H_h2s_base = mxGetPr(PAR)[137]; // Flores-Alsina et al. (2016)

// Fe extension

COD_Fe3 = mxGetPr(PAR)[138]; // Flores-Alsina et al. (2016)
COD_Fe2 = mxGetPr(PAR)[139]; // Flores-Alsina et al. (2016)
i_Fe3_Fe2 = mxGetPr(PAR)[140]; // Flores-Alsina et al. (2016)
K1_Fe3_Fe2 = mxGetPr(PAR)[141]; // Flores-Alsina et al. (2016); only a small part will be converted.
Precipitation will be faster
K2_Fe3_Fe2 = mxGetPr(PAR)[142]; // Flores-Alsina et al. (2016); this will contribute to make conversion
faster

// Multiple precipitation

k_fes = mxGetPr(KIN_PAR)[0];
n_fes = mxGetPr(KIN_PAR)[1];

KT_FeSs = mxGetPr(PRECRX_PAR)[0];

// solving method

algebraic = mxGetPr(ALGEBRAIC)[0];

```

```

for (i = 0; i < 38; i++) {
    if (x[i] < 0)
        xtemp[i] = 0;
    else
        xtemp[i] = x[i];
}

S_su   = xtemp[0];
S_aa   = xtemp[1];
S_fa   = xtemp[2];
S_va   = xtemp[3];
S_bu   = xtemp[4];
S_pro  = xtemp[5];
S_ac   = xtemp[6];

if (algebraic < 0.5)
    S_h2 = xtemp[7]; /* Sh2 */
else
    S_h2 = u[45]; /* Sh2 */

S_ch4   = xtemp[8];
S_IC    = xtemp[9];
S_IN    = xtemp[10];
S_I     = xtemp[11];
X_xc    = xtemp[12];
X_ch    = xtemp[13];
X_pr    = xtemp[14];
X_li    = xtemp[15];
X_su    = xtemp[16];
X_aa    = xtemp[17];
X_fa    = xtemp[18];
X_c4    = xtemp[19];
X_pro   = xtemp[20];
X_ac    = xtemp[21];
X_h2    = xtemp[22];
X_I     = xtemp[23];

S_so4   = xtemp[28];
S_IS    = xtemp[29];
X_hSRB  = xtemp[30];
X_aSRB  = xtemp[31];
X_pSRB  = xtemp[32];
X_c4SRB = xtemp[33];

S_gas_h2 = xtemp[24];
S_gas_ch4 = xtemp[25];
S_gas_co2 = xtemp[26];
S_gas_h2s = xtemp[27];

S_Fe2   = xtemp[35];
S_Fe3   = xtemp[36];

S_co2   = u[104];
S_nh3   = u[100];
S_h2s   = u[93]*64.0*1.0;

S_act_H = u[37];
S_act_NH4 = u[38];
S_act_Ac = u[39];
S_act_Pr = u[40];
S_act_CO3 = u[41];

```

```

S_act_SO4 = u[42];
S_act_Fe2 = u[43];
S_act_Fe3 = u[44];
S_act_HS = u[45];
S_act_Bu = u[46];
S_act_Va = u[47];
S_act_NO3 = u[48];
S_act_NO2 = u[49];
mu      = u[50];
gama    = u[51];

S_conc_H  = u[52];
S_conc_NH4 = u[53];
S_conc_Ac = u[54];
S_conc_Pr = u[55];
S_conc_CO3 = u[56];
S_conc_SO4 = u[57];
S_conc_Fe2 = u[58];
S_conc_Fe3 = u[59];
S_conc_HS = u[60];
S_conc_Bu = u[61];
S_conc_Va = u[62];
S_conc_NO3 = u[63];
S_conc_NO2 = u[64];

S_act_S = (S_act_HS)/(S_act_H);

X_FeS    = xtemp[37];

factor    = (1.0/T_base - 1.0/T_op)/(100.0*R);
K_H_h2    = K_H_h2_base*exp(-4180.0*factor);      /* T adjustment for K_H_h2 */
K_H_ch4    = K_H_ch4_base*exp(-14240.0*factor);   /* T adjustment for K_H_ch4 */
K_H_co2    = K_H_co2_base*exp(-19410.0*factor);   /* T adjustment for K_H_co2 */
K_H_h2s    = K_H_h2s_base*exp(-19180.0*factor);   /* T adjustment for K_H_h2s */
p_gas_h2o  = K_H_h2o_base*exp(5290.0*(1.0/T_base - 1.0/T_op)); /* T adjustment for water vapour saturation
pressure */

// needs to be adjusted

S_H_ion = u[36];
pH_op = -log10(u[36]); /* pH */

p_gas_h2 = S_gas_h2*R*T_op/16.0;
p_gas_ch4 = S_gas_ch4*R*T_op/64.0;
p_gas_co2 = S_gas_co2*R*T_op;
p_gas_h2s = 1.0*S_gas_h2s*R*T_op/64.0;

P_gas = p_gas_h2 + p_gas_ch4 + p_gas_co2 + p_gas_h2o + p_gas_h2s;

/* Hill function on SH+ used within BSM2, ADM1 Workshop, Copenhagen 2005. */

pHLim_aa = pow(10,(-(pH_UL_aa + pH_LL_aa)/2.0));
pHLim_ac = pow(10,(-(pH_UL_ac + pH_LL_ac)/2.0));
pHLim_h2 = pow(10,(-(pH_UL_h2 + pH_LL_h2)/2.0));
n_aa     = 3.0/(pH_UL_aa-pH_LL_aa);
n_ac     = 3.0/(pH_UL_ac-pH_LL_ac);
n_h2     = 3.0/(pH_UL_h2-pH_LL_h2);
l_pH_aa  = pow(pHLim_aa,n_aa)/(pow(S_H_ion,n_aa)+pow(pHLim_aa,n_aa));
l_pH_ac  = pow(pHLim_ac,n_ac)/(pow(S_H_ion,n_ac)+pow(pHLim_ac,n_ac));
l_pH_h2  = pow(pHLim_h2,n_h2)/(pow(S_H_ion,n_h2)+pow(pHLim_h2,n_h2));

l_IN_lim = 1.0/(1.0+K_S_IN/S_IN);

```

```

I_h2_fa = 1.0/(1.0+S_h2/K_Ih2_fa);
I_h2_c4 = 1.0/(1.0+S_h2/K_Ih2_c4);
I_h2_pro = 1.0/(1.0+S_h2/K_Ih2_pro);
I_nh3 = 1.0/(1.0+ S_nh3/K_I_nh3);

inhib[0] = I_pH_aa*I_IN_lim;
inhib[1] = inhib[0]*I_h2_fa;
inhib[2] = inhib[0]*I_h2_c4;
inhib[3] = inhib[0]*I_h2_pro;
inhib[4] = I_pH_ac*I_IN_lim*I_nh3;
inhib[5] = I_pH_h2*I_IN_lim;

// S related inhibition functions

pHLim_hSRB = pow(10,(-(pH_UL_hSRB + pH_LL_hSRB)/2.0));
pHLim_aSRB = pow(10,(-(pH_UL_aSRB + pH_LL_aSRB)/2.0));
pHLim_pSRB = pow(10,(-(pH_UL_pSRB + pH_LL_pSRB)/2.0));
pHLim_c4SRB = pow(10,(-(pH_UL_c4SRB + pH_LL_c4SRB)/2.0));

n_hSRB = 3.0/(pH_UL_hSRB-pH_LL_hSRB);
n_aSRB = 3.0/(pH_UL_aSRB-pH_LL_aSRB);
n_pSRB = 3.0/(pH_UL_pSRB-pH_LL_pSRB);
n_c4SRB = 3.0/(pH_UL_c4SRB-pH_LL_c4SRB);

I_pH_hSRB = pow(pHLim_hSRB,n_hSRB)/(pow(S_H_ion,n_hSRB)+pow(pHLim_hSRB ,n_hSRB));
I_pH_aSRB = pow(pHLim_aSRB,n_aSRB)/(pow(S_H_ion,n_aSRB)+pow(pHLim_aSRB ,n_aSRB));
I_pH_pSRB = pow(pHLim_pSRB,n_pSRB)/(pow(S_H_ion,n_pSRB)+pow(pHLim_pSRB ,n_pSRB));
I_pH_c4SRB = pow(pHLim_c4SRB,n_c4SRB)/(pow(S_H_ion,n_c4SRB)+pow(pHLim_c4SRB ,n_c4SRB));

I_h2s_hSRB = 1/(1+(S_h2s)/K_I_h2s_hSRB);
I_h2s_aSRB = 1/(1+(S_h2s)/K_I_h2s_aSRB);
I_h2s_pSRB = 1/(1+(S_h2s)/K_I_h2s_pSRB);
I_h2s_c4SRB = 1/(1+(S_h2s)/K_I_h2s_c4SRB);

I_h2s_h2 = 1/(1+(S_h2s)/K_I_h2s_h2);
I_h2s_c4 = 1/(1+(S_h2s)/K_I_h2s_c4);
I_h2s_pro = 1/(1+(S_h2s)/K_I_h2s_pro);
I_h2s_ac = 1/(1+(S_h2s)/K_I_h2s_ac);

// traditional ADM1 kinetic processes

proc1 = 0.0;
proc2 = k_hyd_ch*X_ch;
proc3 = k_hyd_pr*X_pr;
proc4 = k_hyd_li*X_li;
proc5 = k_m_su*S_su/(K_S_su+S_su)*X_su*inhib[0];
proc6 = k_m_aa*S_aa/(K_S_aa+S_aa)*X_aa*inhib[0];
proc7 = k_m_fa*S_fa/(K_S_fa+S_fa)*X_fa*inhib[1];
proc8 = k_m_c4*S_va/(K_S_c4+S_va)*X_c4*S_va/(S_va+S_bu+eps)*inhib[2]*I_h2s_c4;
proc9 = k_m_c4*S_bu/(K_S_c4+S_bu)*X_c4*S_bu/(S_va+S_bu+eps)*inhib[2]*I_h2s_c4;
proc10 = k_m_pro*S_pro/(K_S_pro+S_pro)*X_pro*inhib[3]*I_h2s_pro;
proc11 = k_m_ac*S_ac/(K_S_ac+S_ac)*X_ac*inhib[4]*I_h2s_ac;
proc12 = k_m_h2*(S_h2)/(K_S_h2+(S_h2))*X_h2*inhib[5]*I_h2s_h2;
proc13 = k_dec_Xsu*X_su;
proc14 = k_dec_Xaa*X_aa;
proc15 = k_dec_Xfa*X_fa;
proc16 = k_dec_Xc4*X_c4;
proc17 = k_dec_Xpro*X_pro;
proc18 = k_dec_Xac*X_ac;
proc19 = k_dec_Xh2*X_h2;

// Bio S processes
proc20 =
k_m_hSRB*S_h2/(K_S_hSRB+S_h2)*(S_so4/(K_S_so4_hSRB+S_so4))*X_hSRB*I_pH_hSRB*I_h2s_hSRB*I_IN_lim;

```

```

proc21 = k_dec_XhSRB*X_hSRB;
proc22 =
Bio_S*k_m_aSRB*S_ac/(K_S_aSRB+S_ac)*(S_so4/(K_S_so4_aSRB+S_so4))*X_aSRB*I_pH_aSRB*I_h2s_aSRB*I_IN_
lim;
proc23 = Bio_S*k_dec_XaSRB*X_aSRB;
proc24 =
Bio_S*k_m_pSRB*S_pro/(K_S_pSRB+S_pro)*(S_so4/(K_S_so4_pSRB+S_so4))*X_pSRB*I_pH_pSRB*I_h2s_pSRB*I_I
N_lim;
proc25 = Bio_S*k_dec_XpSRB*X_pSRB;
proc26 =
Bio_S*k_m_c4SRB*S_bu/(K_S_c4SRB+S_bu)*(S_so4/(K_S_so4_c4SRB+S_so4))*X_c4SRB*I_pH_c4SRB*S_bu/(S_va+
S_bu+eps)*I_IN_lim*I_h2s_c4SRB;
proc27 =
Bio_S*k_m_c4SRB*S_va/(K_S_c4SRB+S_va)*(S_so4/(K_S_so4_c4SRB+S_so4))*X_c4SRB*I_pH_c4SRB*S_va/(S_bu+
S_va+eps)*I_IN_lim*I_h2s_c4SRB;
proc28 = Bio_S*k_dec_Xc4SRB*X_c4SRB;
proc29 = 0.0;
// Oxidation Reduction processes - note process number has been modifies

proc30 = K1_Fe3_Fe2*S_Fe3*S_h2/16;
proc31 = K2_Fe3_Fe2*S_Fe3*S_IS/64; // for future use (Fe+3 reduction by oxidizing HS)

// Multiple mineral precipitation

S_act_S = (S_act_HS)/(S_act_H);

SI_fes          = log10(S_act_Fe2*S_act_S/KT_FeSs);

if (SI_fes < 0.0) {P_fes = 0.0;} // changed from 0.0
else { P_fes = 1.0;}

//////////

/* FeS */
proc32 = 1.0*k_fes* X_FeS* pow(pow((S_act_Fe2*S_act_S)/KT_FeSs, 1.0/2.0)-
1.0,n_fes);

// Stripping processes

procT8 = kLa*(S_h2- 16.0*K_H_h2*p_gas_h2);
procT9 = kLa*(S_ch4-64.0*K_H_ch4*p_gas_ch4);
procT10 = kLa*(S_co2-K_H_co2*p_gas_co2);
procT11 = kLa*(S_h2s -64.0*K_H_h2s*p_gas_h2s);

/// source sink calculations

stoich1_c = 0.0;

// this stoich has been modified : Xc is omitted

stoich2_c = -C_ch+C_su;
stoich3_c = -C_pr+C_aa;
stoich4_c = -C_li+(1.0-f_fa_li)*C_su+f_fa_li*C_fa;
stoich5_c = -C_su+(1.0-Y_su)*(f_bu_su*C_bu+f_pro_su*C_pro+f_ac_su*C_ac)+Y_su*C_bac;
stoich6_c = -C_aa+(1.0-Y_aa)*(f_va_aa*C_va+f_bu_aa*C_bu+f_pro_aa*C_pro+f_ac_aa*C_ac)+Y_aa*C_bac;
stoich7_c = -C_fa+(1.0-Y_fa)*0.7*C_ac+Y_fa*C_bac;
stoich8_c = -C_va+(1.0-Y_c4)*0.54*C_pro+(1.0-Y_c4)*0.31*C_ac+Y_c4*C_bac;
stoich9_c = -C_bu+(1.0-Y_c4)*0.8*C_ac+Y_c4*C_bac;
stoich10_c = -C_pro+(1.0-Y_pro)*0.57*C_ac+Y_pro*C_bac;
stoich11_c = -C_ac+(1.0-Y_ac)*C_ch4+Y_ac*C_bac;
stoich12_c = (1.0-Y_h2)*C_ch4+Y_h2*C_bac;
stoich13_c = -C_bac+f_sl_xc*C_sl+f_ch_xc*C_ch+f_pr_xc*C_pr+f_li_xc*C_li+f_xl_xc*C_xl; // this stoich has
been modified : Xc is omitted - need to check equation
stoich14_c = Y_hSRB*C_bac;

```

```

    stoich15_c = -C_bac+f_sl_xc*C_sl+f_ch_xc*C_ch+f_pr_xc*C_pr+f_li_xc*C_li+f_xl_xc*C_xl; // this stoich has been
modified : Xc is omitted
    stoich16_c = -C_ac+Y_aSRB*C_bac;
    stoich17_c = -C_bac+f_sl_xc*C_sl+f_ch_xc*C_ch+f_pr_xc*C_pr+f_li_xc*C_li+f_xl_xc*C_xl; // this stoich has been
modified : Xc is omitted
    stoich18_c = -C_pro+(1.0-Y_pSRB)*0.57*C_ac+Y_pSRB*C_bac;
    stoich19_c = -C_bac+f_sl_xc*C_sl+f_ch_xc*C_ch+f_pr_xc*C_pr+f_li_xc*C_li+f_xl_xc*C_xl; // this stoich has been
modified : Xc is omitted
    stoich20_c = -C_bu+(1.0-Y_c4SRB)*0.8*C_ac+Y_c4SRB*C_bac;
    stoich21_c = -C_va+(1.0-Y_c4SRB)*0.54*C_pro+(1.0-Y_c4SRB)*0.31*C_ac+Y_c4SRB*C_bac;
    stoich22_c = -C_bac+f_sl_xc*C_sl+f_ch_xc*C_ch+f_pr_xc*C_pr+f_li_xc*C_li+f_xl_xc*C_xl; // this stoich has been
modified : Xc is omitted
    stoich23_c = 0.0;

    stoich1_n = 0.0;

    stoich2_n = 0.0;
    stoich3_n = -N_pr+N_aa;
    stoich4_n = 0.0;
    stoich5_n = +Y_su*N_bac;
    stoich6_n = -N_aa +Y_aa*N_bac;
    stoich7_n = Y_fa*N_bac;
    stoich8_n = Y_c4*N_bac;
    stoich9_n = Y_c4*N_bac;
    stoich10_n = Y_pro*N_bac;
    stoich11_n = Y_ac*N_bac;
    stoich12_n = Y_h2*N_bac;
    stoich13_n = -N_bac+f_sl_xc*N_l+f_pr_xc*N_pr+f_xl_xc*N_l; // this stoich has been modified : Xc
is omitted
    stoich14_n = Y_hSRB*N_bac;
    stoich15_n = -N_bac+f_sl_xc*N_l+f_pr_xc*N_pr+f_xl_xc*N_l; // this stoich has been modified : Xc
is omitted
    stoich16_n = Y_aSRB*N_bac;
    stoich17_n = -N_bac+f_sl_xc*N_l+f_pr_xc*N_pr+f_xl_xc*N_l; // this stoich has been modified : Xc
is omitted
    stoich18_n = Y_pSRB*N_bac;
    stoich19_n = -N_bac+f_sl_xc*N_l+f_pr_xc*N_pr+f_xl_xc*N_l; // this stoich has been modified : Xc
is omitted
    stoich20_n = Y_c4SRB*N_bac;
    stoich21_n = Y_c4SRB*N_bac;
    stoich22_n = -N_bac+f_sl_xc*N_l+f_pr_xc*N_pr+f_xl_xc*N_l; // this stoich has been modified : Xc
is omitted
    stoich23_n = 0.0;

    stoich1_s = 0.0;
    stoich2_s = 0.0;
    stoich3_s = 0.0;
    stoich4_s = 0.0;
    stoich5_s = 0.0;
    stoich6_s = 0.0;
    stoich7_s = 0.0;
    stoich8_s = 0.0;
    stoich9_s = 0.0;
    stoich10_s = 0.0;
    stoich11_s = 0.0;
    stoich12_s = 0.0;
    stoich13_s = 0.0;
    stoich14_s = -(1.0-Y_hSRB);
    stoich15_s = 0.0;
    stoich16_s = -(1.0-Y_aSRB);
    stoich17_s = 0.0;
    stoich18_s = -(1.0-Y_pSRB)*0.43;
    stoich19_s = 0.0;
    stoich20_s = -(1.0-Y_c4SRB)*0.20;

```

```

stoich21_s = -(1.0-Y_c4SRB)*0.15;
stoich22_s = 0.0;
stoich23_s = 0.0;

// Traditional stoichiometry ADM1

reac1 = proc2+(1.0-f_fa_li)*proc4-proc5; //su
reac2 = proc3-proc6; //aa
reac3 = f_fa_li*proc4-proc7; //fa
reac4 = (1.0-Y_aa)*f_va_aa*proc6-proc8-proc27; //va
reac5 = (1.0-Y_su)*f_bu_su*proc5+(1.0-Y_aa)*f_bu_aa*proc6 - proc9 -proc26; //bu
reac6 = (1.0-Y_su)*f_pro_su*proc5+(1.0-Y_aa)*f_pro_aa*proc6+(1.0-Y_c4)*0.54*proc8 +(1.0-
Y_c4SRB)*0.54*proc27- proc10 -proc24; //pro
reac7 = (1.0-Y_su)*f_ac_su*proc5+(1.0-Y_aa)*f_ac_aa*proc6+(1.0-Y_fa)*0.7*proc7+(1.0-Y_c4)*0.31*proc8 +(1.0-
Y_c4SRB)*0.31*proc27+(1.0-Y_c4)*0.8*proc9 + (1.0-Y_c4SRB)*0.8*proc26+(1.0-Y_pro)*0.57*proc10-proc11 - proc22 +
(1.0-Y_pSRB)*0.57*proc24; //ac
reac8 = (1.0-Y_su)*f_h2_su*proc5+(1.0-Y_aa)*f_h2_aa*proc6+(1.0-Y_fa)*0.3*proc7+(1.0-Y_c4)*0.15*proc8+(1.0-
Y_c4)*0.2*proc9+(1.0-Y_pro)*0.43*proc10-proc12 - proc20 - proc30 -procT8; //h2
reac9 = (1.0-Y_ac)*proc11+(1.0-Y_h2)*proc12-procT9; //ch4
reac10 = -stoich1_c*proc1-stoich2_c*proc2-stoich3_c*proc3-stoich4_c*proc4-stoich5_c*proc5- stoich6_c*proc6-
stoich7_c*proc7-stoich8_c*proc8-stoich9_c*proc9-stoich10_c*proc10-stoich11_c*proc11-stoich12_c*proc12-
stoich13_c*proc13-stoich13_c*proc14-stoich13_c*proc15-stoich13_c*proc16-stoich13_c*proc17-stoich13_c*proc18-
stoich13_c*proc19 - stoich14_c*proc20 - stoich15_c*proc21 - stoich16_c*proc22 - stoich17_c*proc23 -
stoich18_c*proc24 - stoich19_c*proc25 - stoich20_c*proc26- stoich21_c*proc27 - stoich22_c*proc28-
stoich23_c*proc29- procT10; //IC
reac11 = -stoich1_n*proc1-stoich2_n*proc2-stoich3_n*proc3-stoich4_n*proc4-stoich5_n*proc5- stoich6_n*proc6-
stoich7_n*proc7-stoich8_n*proc8-stoich9_n*proc9-stoich10_n*proc10-stoich11_n*proc11-stoich12_n*proc12-
stoich13_n*proc13-stoich13_n*proc14-stoich13_n*proc15-stoich13_n*proc16-stoich13_n*proc17-stoich13_n*proc18-
stoich13_n*proc19 - stoich14_n*proc20 - stoich15_n*proc21 - stoich16_n*proc22 - stoich17_n*proc23 -
stoich18_n*proc24 - stoich19_n*proc25 - stoich20_n*proc26- stoich21_n*proc27 - stoich22_n*proc28-
stoich23_n*proc29; //IN
reac12 = f_sl_xc *(proc13 + proc14 + proc15 + proc16 + proc17 + proc18 + proc19 + proc21 + proc23 + proc25 +
proc28); //SI // this reac has been modified :direct mapping from biomass decay
reac13 = 0.0;
reac14 = f_ch_xc *(proc13 + proc14 + proc15 + proc16 + proc17 + proc18 + proc19 + proc21 + proc23 + proc25 +
proc28) -proc2; //Ch // this reac has been modified: direct mapping from biomass decay
reac15 = f_pr_xc *(proc13 + proc14 + proc15 + proc16 + proc17 + proc18 + proc19 + proc21 + proc23 + proc25 +
proc28) -proc3; //Pr // this reac has been modified: direct mapping from biomass decay
reac16 = f_li_xc *(proc13 + proc14 + proc15 + proc16 + proc17 + proc18 + proc19 + proc21 + proc23 + proc25 +
proc28) -proc4; //Li // this reac has been modified :direct mapping from biomass decay
reac17 = Y_su*proc5-proc13; //Xsu
reac18 = Y_aa*proc6-proc14; //Xaa
reac19 = Y_fa*proc7-proc15; //Xfa
reac20 = Y_c4*proc8+Y_c4*proc9-proc16; //Xc4
reac21 = Y_pro*proc10-proc17; //Xpro
reac22 = Y_ac*proc11-proc18; //Xac
reac23 = Y_h2*proc12-proc19; //Xh2
reac24 = f_xl_xc *(proc13 + proc14 + proc15 + proc16 + proc17 + proc18 + proc19 + proc21 + proc23 + proc25 +
proc28); //XI // this reac has been modified :direct mapping from biomass decay

// S removal equations

reac25 = -(1.0-Y_hSRB)*proc20/64.0 -(1.0-Y_aSRB)*proc22/64.0 - (1.0-Y_pSRB)*0.43*proc24/64.0 - (1.0-
Y_c4SRB)*0.20*proc26/64 - (1.0-Y_c4SRB)*0.15*proc27/64; //SO4
reac26 = -stoich1_s*proc1 -stoich2_s*proc2 -stoich3_s*proc3 -stoich4_s*proc4 -stoich5_s*proc5- stoich6_s*proc6 -
stoich7_s*proc7 -stoich8_s*proc8 -stoich9_s*proc9 -stoich10_s*proc10 -stoich11_s*proc11 -stoich12_s*proc12 -
stoich13_s*proc13 -stoich13_s*proc14 -stoich13_s*proc15 -stoich13_s*proc16 -stoich13_s*proc17 -stoich13_s*proc18 -
stoich13_s*proc19 - stoich14_s*proc20 - stoich15_s*proc21 - stoich16_s*proc22 - stoich17_s*proc23 -
stoich18_s*proc24 - stoich19_s*proc25 - stoich20_s*proc26 - stoich21_s*proc27- stoich22_s*proc28- stoich23_s*proc29
- procT11; //IS
reac27 = (Y_hSRB)*proc20 - proc21; //XhSRB
reac28 = (Y_aSRB)*proc22 - proc23; //XaSRB
reac29 = (Y_pSRB)*proc24 - proc25; //XpSRB

```

```
reac30 = Y_c4SRB*proc26+Y_c4SRB*proc27-proc28;
```

```
// Ions / Cations
```

```
reac31 = 0.0; //Fe2
```

```
reac32 = 0.0; //Fe3
```

```
// Oxidation_Reduction
```

```
reac33 = 0.0; // Iron Sulfide
```

```
// Differential equations
```

```
q_gas = k_P*(P_gas-P_atm);
```

```
if (q_gas < 0)
```

```
    q_gas = 0.0;
```

```
dx[0] = reac1;
```

```
dx[1] = reac2;
```

```
dx[2] = reac3;
```

```
dx[3] = reac4;
```

```
dx[4] = reac5;
```

```
dx[5] = reac6;
```

```
dx[6] = reac7;
```

```
/* calculated in Sh2solv.c */
```

```
dx[7] = reac8;
```

```
dx[8] = reac9;
```

```
dx[9] = reac10; /* SIC */
```

```
dx[10] = reac11; /* SIN */
```

```
dx[11] = reac12;
```

```
dx[12] = reac13;
```

```
dx[13] = reac14;
```

```
dx[14] = reac15;
```

```
dx[15] = reac16;
```

```
dx[16] = reac17;
```

```
dx[17] = reac18;
```

```
dx[18] = reac19;
```

```
dx[19] = reac20;
```

```
dx[20] = reac21;
```

```
dx[21] = reac22;
```

```
dx[22] = reac23;
```

```
dx[23] = reac24;
```

```
/* MASS TRANSFER EQUATIONS */
```

```
dx[24] = procT8*V_liq/V_gas; /* Sh2 */
```

```
dx[25] = procT9*V_liq/V_gas; /* Sch4 */
```

```
dx[26] = procT10*V_liq/V_gas; /* Sco2 */
```

```
dx[27] = procT11*V_liq/V_gas; /* Sh2s */
```

```
/* Bio S reactions (reac35) */
```

```
dx[28] = reac25; /* SO4-2 */
```

```
dx[29] = reac26 - proc31 - 64.0*P_fes*proc32; /* S_IS */
```

```
dx[30] = reac27; /* X_hSRB */
```

```
dx[31] = reac28; /* X_aSRB */
```

```
dx[32] = reac29; /* X_pSRB */
```

```
dx[33] = reac30; /* X_c4SRB */
```

```
dx[34] = (48.0/64.0)*proc31; /* S0 elemental S */
```

```

/* Scations(+) and Sanions(-) */

dx[35] = reac31 + proc30 + (16.0/64.0)*proc31- 8.0*P_fes*proc32; /* Fe+2 in COD units */
dx[36] = reac32 - (1.0/l_Fe3_Fe2)*proc30 - (1.0/l_Fe3_Fe2)*(16.0/64.0)*proc31; /* Fe+3 */

/* Multiple minerals */

dx[37] = P_fes*proc32;          /* Iron Sulfide */

}

/*
 * mdlTerminate - called when the simulation is terminated.
 */
static void mdlTerminate(SimStruct *S)
{
}

#ifdef MATLAB_MEX_FILE /* Is this file being compiled as a MEX-file? */
#include "simulink.c" /* MEX-file interface mechanism */
#else
#include "cg_sfun.h" /* Code generation registration function */
#endif

% Initializing all files in Matlab before running Simulink model

clear
clc

admlint_grass_synAMD;

speciationint_adml_synAMD;
precipitationint_adml_synAMD;

load ADM1_influent_synAMD

% S_Fe_calculation

Bio_S = 1;          % when it is 0, SRB use H2 as electron source
                    % when it is 1, SRB use H2 + organic acids as electron
                    % source

ALGEBRAIC = 0;      % when it is 0, H is solved as ODE
                    % when it is 1, H is solved as DAE

```

APPENDIX D RAW EXPERIMENTAL DATA

Table 6.10 pH (ratio 1.5)

Day	pH	
	Sample 1	Sample 2
0	3.19	2.83
2	3.34	3.28
3	3.27	3.29
7	2.91	2.92
8	2.98	2.93
11	2.91	2.91
15	3.34	3.32
16	3.08	3.06
18	3.37	3.37
21	3.4	3.42
23	3.42	3.42
25	3.35	3.4
28	3.35	3.39
30	3.35	3.31

Table 6.11 pH (ratio 6)

Day	pH	
	Sample 1	Sample 2
0	2.77	2.77
4	6.47	6.3
6	6.55	7.15
9	6.82	6.84
12	6.69	6.72
15	6.7	6.83
18	6.78	6.94
21	6.83	7.01
24	6.87	6.91
27	6.83	7.03
30	6.93	7.05

Table 6.12 pH (ratio 30)

Day	pH	
	Sample 1	Sample 2
0	3.23	3.23
2	6.28	6.27
6	6.6	6.61
9	6.75	6.8
14	6.41	6.4
17	6.82	6.87
21	6.53	6.47
24	6.74	6.84
27	6.7	6.61
30	7.69	7.72

Table 6.13 Sulfate Concentration (ratio 1.5)

Day	Sulfate Concentration (mg/L)	
	Sample 1	Sample 2
0	104.69	104.69
7	79.63	52.12
8	75.24	46.05
11	84.93	51.35
15	78.99	54.45
18	83.12	87.25
21	85.58	52.64
23	91.90	55.35
25	79.76	53.93
28	86.61	55.87
30	76.66	81.83

Table 6.14 Sulfate Concentration (ratio 6)

Day	Sulfate Concentration (mg/L)	
	Sample 1	Sample 2
0	783.01	783.01
4	284.74	387.04
6	230.75	215.25
9	253.74	198.98
12	301.01	221.84
15	248.06	194.59
18	204.92	242.38
21	229.33	134.01
24	268.72	257.75
27	251.03	259.94

30	259.17	165.27
----	--------	--------

Table 6.15 Sulfate Concentration (ratio 30)

Day	Sulfate Concentration (mg/L)	
	Sample 1	Sample 2
0	3276.85	3276.85
2	1126.25	830.36
6	977.64	903.72
9	1086.12	1001.57
14	1246.69	1347.03
17	1065.30	981.19
21	1288.07	1410.38
24	485.33	539.84
27	1306.91	1444.92
30	1327.62	1511.05

Table 6.16 COD Concentration (ratio1.5)

Day	COD Concentration (mg/L)	
	Sample 1	Sample 2
0	0.00	0.00
2	73.40	90.85
3	62.42	58.11
7	64.58	39.81
8	53.38	59.62
11	51.01	37.01
15	69.10	24.95
18	127.24	47.78
21	109.37	65.22
23	58.33	71.25
25	90.63	63.07
28	102.48	73.19
30	68.45	58.33

Table 6.17 COD Concentration (ratio 6)

Day	COD Concentration (mg/L)	
	Sample 1	Sample 2
0	0.00	0.00
4	521.59	322.73
6	380.80	370.24
9	407.20	505.76
12	553.27	477.60
15	496.96	322.73
18	391.36	326.25
21	354.40	382.56
24	350.88	440.64
27	368.48	403.68
30	521.59	643.03

Table 6.18 COD Concentration (ratio 30)

Day	COD Concentration (mg/L)	
	Sample 1	Sample 2
0	89.67	89.67
2	1445.50	1133.00
6	700.50	704.67
9	698.83	981.33
14	1095.50	991.33
17	953.83	943.00
21	1645.50	1584.67
24	1257.17	1326.33
27	1705.50	1593.00
30	1792.17	1708.00

Table 6.19 pH AMD and syn AMD

Day	pH (AMD)		pH (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	2.07	2.07	3.23	3.23
2	3.19	3.09	6.28	6.27
6	3.27	3.33	6.6	6.61
9	3.41	3.37	6.75	6.8
14	3.7	3.67	6.41	6.4
17	3.78	3.74	6.82	6.87
21	3.83	3.79	6.53	6.47
24	3.69	3.68	6.74	6.84
27	3.86	3.85	6.7	6.61
30	3.98	3.99	7.69	7.72

Table 6.20 COD Concentration AMD and syn AMD

Day	COD Concentration (mg/L) (AMD)		COD Concentration (mg/L) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	248.83	248.83	89.67	89.67
2	923.83	937.17	1445.50	1133.00
6	690.50	832.17	700.50	704.67
9	967.17	1010.50	698.83	981.33
14	1085.50	1180.50	1095.50	991.33
17	1225.50	1312.17	953.83	943.00
21	1162.17	1308.83	1645.50	1584.67
24	1420.50	1572.17	1257.17	1326.33
27	1208.83	1343.83	1705.50	1593.00
30	1222.17	1428.83	1792.17	1708.00

Table 6.21 Sulfate Concentration AMD and syn AMD

Day	Sulfate Concentration (mg/L) (AMD)		Sulfate Concentration (mg/L) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	1403.58	1403.58	3276.85	3276.85
2	572.80	563.18	1126.25	830.36
6	1338.91	1202.26	977.64	903.72
9	796.90	1155.22	1086.12	1001.57
14	900.45	1216.89	1246.69	1347.03
17	860.00	1152.43	1065.30	981.19
21	775.04	835.39	1288.07	1410.38
24	850.69	1450.17	485.33	539.84
27	1184.94	1338.91	1306.91	1444.92
30	741.12	775.98	1327.62	1511.05

Table 6.22 Sulfide Concentration AMD and syn AMD

Day	Sulfide Concentration (ppm) (AMD)		Sulfide Concentration (ppm) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	0.067	0.067	0.051	0.051
2	0.083	0.082	0.251	0.306
6	0.100	0.094	0.208	0.286
9	0.085	0.122	0.183	0.190
14	0.046	0.055	0.254	0.304
17	0.076	0.552	0.196	0.165
21	0.101	0.166	0.789	0.735
24	0.064	0.229	0.332	0.466
27	0.051	0.245	0.448	0.769
30	0.043	0.172	0.604	0.681

Table 6.23 Dissolved Iron Concentration AMD and syn AMD

Day	Dissolved Iron Concentration (ppm) (AMD)		Dissolved Iron Concentration (ppm) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	185.09	185.09	233.28	233.28
2	64.26	62.25	151.25	149.90
6	60.81	31.25	60.38	25.42
9	108.01	100.48	31.97	19.98
14	125.53	135.36	140.34	128.56
17	135.73	141.32	20.44	22.25
21	140.01	147.31	110.13	108.13
24	168.50	168.48	20.54	12.62
27	168.39	167.96	31.94	40.51
30	160.53	159.97	14.80	2.46

Table 6.24 Total Iron Concentration AMD and syn AMD

Day	Total Iron Concentration (ppm) (AMD)		Total Iron Concentration (ppm) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	368.94	368.94	493.11	493.11
2	75.75	256.38	178.31	302.24
6	153.76	38.52	86.67	184.92
9	246.52	194.81	38.02	27.71
14	386.14	242.83	296.76	279.45
17	329.27	277.23	171.46	161.06
21	248.61	304.42	242.58	294.70
24	410.60	293.70	132.11	48.46
27	321.54	383.93	129.12	92.46
30	502.42	563.21	21.65	101.96

Table 6.25 Sugar Concentration AMD and syn AMD

Day	Sugars Concentration (mg/L) (AMD)		Sugars Concentration (mg/L) (syn AMD)	
	Sample 1	Sample 2	Sample 1	Sample 2
0	2.01	2.01	2.01	2.01
2	4.72	17.30	2.91	2.90
6	17.91	18.94	2.91	2.33
9	9.14	10.40	2.94	2.43
14	3.22	16.03	3.40	2.50
17	22.41	10.80	2.59	16.12
21	13.86	13.78	2.13	2.58
24	22.94	17.82	2.33	3.43
27	11.97	13.73	2.82	2.59
30	25.82	11.63	7.25	4.13

APPENDIX E PUBLICATIONS

Process design for the treatment of acid mine drainage utilising lignocellulosic material as an organic carbon source for the process of dissimilatory sulfate reduction

Natalie Greenway^{1,2,3}, Lizelle van Dyk^{*1,2,3}, Kevin G. Harding^{1,2,3}, Ferdi Postma^{1,2,3,4}, Karl Rumbold^{1,2,5}, Geoffrey S. Simate^{1,2,3}, Craig Sheridan^{1,2,3}

¹ Industrial and Mining Water Research Unit

² Centre in Water Research and Development

³ School of Chemical and Metallurgical Engineering

⁴ Global Change Institute

⁵ School of Microbiology

All of the University of the Witwatersrand, Johannesburg, Private Bag 3, Wits, 2050, South Africa

* Corresponding Author. E-mail: <mailto:Lizelle.VanDyk@wits.ac.za>, Tel: +27117177552

Abstract

Acid mine drainage (AMD) is an acidic wastewater containing high concentrations of heavy metals and sulfates. Its formation is accelerated due to mining activities and it poses environmental and health risks that necessitate the treatment thereof. Whilst a wide range of technologies are available for treating AMD, bioremediation involving anaerobic dissimilatory sulfate reduction (DSR) has proven to be a cost-effective, long-term solution of passive treatment. Furthermore, the utilisation of lignocellulosic material as the organic carbon source for the DSR process increases the potential of the environmental sustainability and economical benefits of the proposed process. This paper adopted the anaerobic digestion model No.1 (ADM1), which was developed by the International Water Association (IWA) in order to establish a generalised model to predict anaerobic digestion (AD), and upgraded it to include sulfate reduction. The ADM1 evaluates biochemical processes as well as ion association/dissociation and gas-liquid transfer during AD. The proposed model implements a two-phase digestion system. The first phase includes the disintegration of the composite material and hydrolysis in sequentially fed leach bed reactors (SLBR). The second phase utilises an upflow anaerobic sludge bed (UASB) for acidogenesis, acetogenesis, methanogenesis and sulfate reduction. The combination of the SLBR-UASB has not previously been developed to full scale and thus this paper also investigates the potential of this development. Therefore, this paper provides a feasibility study of the proposed process evaluating grass bales, maize silage and green weed silage as organic carbon sources. The ramifications of the organic deficient synthetic ratios ($\text{SO}_4^{2-}:\text{COD}$) on methane concentrations were also determined and the study also determined the potential of producing enough methane such that the process is self-sufficient in meeting its energy demands. The model predicted free and total aqueous sulfides, sulfates, butyrate, propionate and acetate, gas flow and the concentrations of methane in the gas phase. The sulfate reduction efficiencies of grass bales, maize silage and green weed silage were found to be 94.1%, 97.8% and 93.7% respectively.

Keywords

Acid mine drainage (AMD)
Dissimilatory sulfate reduction (DSR)
Anaerobic digestion (AD)
Anaerobic digestion model No.1 (ADM1)

Nomenclature

X_i	Particulate component i (kg COD m^{-3})	q	Flow rate ($\text{m}^3 \text{d}^{-1}$)
S_i	Soluble component i (kg COD m^{-3})	V	Volume (m^3)
ρ_j	Kinetic rate of process j (kg COD $\text{m}^{-3} \text{d}^{-1}$)	$p_{\text{gas},i}$	Pressure of gas component i (bar)
$u_{i,j}$	Rate coefficient for component i on process j	P_{gas}	Total pressure (bar)
$\rho_{T,i}$	Mass transfer rate of gas i (kg COD $\text{m}^{-3} \text{d}^{-1}$)	SRT	Solid retention time (d^{-1})
I_{sulfide}	Sulfide inhibition function	ER_i	Efficiency of the retention on bacterial group i
I_{pH}	pH inhibition function	U_{up}	Upward liquid velocity (m s^{-1})
$k_{\text{max},j}$	Monod maximum specific uptake rate (kg COD $\text{S}_i \text{d kg COD}_X^{-1} \text{d}^{-1}$)	Subscripts	
$K_{s,j}$	Half Saturation value (kg COD $\text{S}_i \text{m}^{-3}$)	i	Component i
$K_{I,i}$	50% Inhibitory function (kmol m^{-3})	j	Process j
$K_{A/B,i}$	Acid-base kinetic parameter ($\text{m}^3 \text{kmol}^{-1} \text{d}^{-1}$)	n	Number of leach beds
$K_{a,i}$	Acid-base equilibrium coefficient (kmol m^{-3})	free	Undissociated species
$k_{L,a}$	Gas-liquid transfer coefficient (d^{-1})	gas	Gas phase
$K_{H,i}$	Henry's law coefficient ($\text{kmol m}^{-3} \text{d}^{-1}$)	liq	Liquid phase
$\rho_{\text{substrate}}$	Density of substrate (kg m^{-3})	UASB	Upflow anaerobic sludge bed
$S_{\text{gas},i}$	Gas phase concentration for component i (kg COD m^{-3})	LT	Leachate tank

1.Introduction

Acid mine drainage (AMD) is formed through the biological oxidation of pyrite, resulting in wastewater with low pH (2 to 3) and high concentrations of sulfates (3 000 to 30 000 mg/l), iron (10 to 6 700 mg/l) and other heavy metals (Christensen et al., 1996, Wichern et al., 2009). Mining activities have accelerated the formation of AMD and this acidic liquid drainage pollutes surface and groundwater in countries which mining plays a vital role to the economy. Active treatment is commonly employed to remediate AMD, however, this form of treatment can be costly and requires constant monitoring. Currently, research is being conducted on passive AMD bioremediation technologies. Bioremediation of AMD has the potential to be a cost-effective and viable long-term treatment solution. During passive treatment the biological process, anaerobic dissimilatory sulfate reduction (DSR), is used to neutralize the pH, reduce sulfate and lower heavy metal concentrations in AMD. Bioremediation of AMD utilises sulfate reducing bacteria (SRB) which catalyses DSR under anaerobic conditions (Lindsay et al., 2011). SRB coupled with an organic carbon source, which serves as an electron donor, causes the reduction of sulfate to sulfide (Ramla and Sheridan, 2015). Anaerobic digestion (AD) consists of various intricate biochemical reactions that are difficult to predict. The anaerobic digestion model No.1 (ADM1) was developed by the International Water

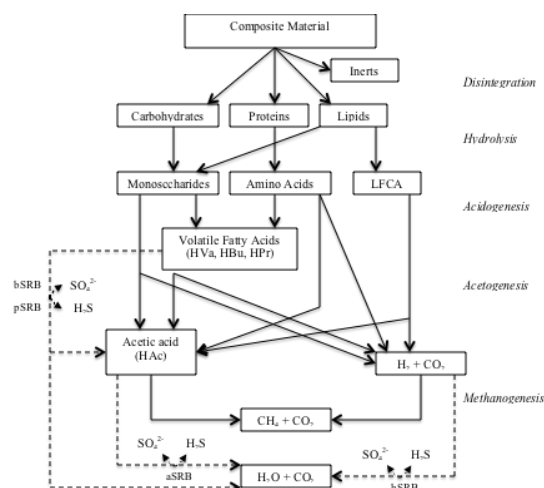


Fig.1 The anaerobic model of the extended ADM1 to include sulfate reduction adapted from Batstone et al. (2002) and Barrera et al. (2013)

Association (IWA) in order to establish a generalised model to predict anaerobic digestion (AD) processes (Batstone et al., 2002). The sophistication and complexity of the ADM1 model involves biochemical and physicochemical processes (Barrera et al., 2015) in order to determine the production of biogas during AD. The biochemical processes include the disintegration of the composite material, hydrolysis of carbohydrates, proteins and lipids, acidogenesis, acetogenesis and methanogenesis as shown in Fig.1 (Batstone et al., 2002). The physicochemical steps evaluate ion association/dissociation and gas-liquid transfer. The ADM1 model aims to develop a common basis for the AD process in order to assist in further research and modeling, increase the application of AD to industry and in full scale design, and aid in operation and optimization (Barrera et al., 2013). The ADM1 developed by Batstone et al. (2002) was not extended to include sulfate reduction due to the complexity of microbial competition, inhibition of H_2S and the composition of H_2S in the gas phase of the AD system. Fedorovich et al. (2003) extended the ADM1 model to include sulfate reduction and is regarded as the most suitable model to predict sulfate removal efficiencies for systems containing SO_4^{2-} :COD ratios above 0.1 gS:gCOD. The model was calibrated against experimental data taken from literature (Omil et al., 1996, Omil et al., 1997) and is able to accurately predict sulfate removal, concentrations of volatile fatty acids (VFAs) in the effluent and the production of biogas. This research aimed to develop a model to predict the behaviour of treating AMD through AD by implementing the extended ADM1 to include sulfate reduction utilising lignocellulosic material as the organic carbon source. A comparison of three different lignocellulosic materials was made: grass bales (comprising of indigenous South African grass), maize silage and green weed silage. Volatile

fatty acids (butyrate, propionate and acetate) produced during the acidogenesis and acetogenesis processes are utilized by the SRB as the electron donor during DSR (Fedorovich et al., 2003).

2. METHODS

2.1 Influent characterization

The ADM1 is implemented on the basis of the chemical oxygen demand (COD) and in order for calibration, the fractioning of the total input COD must be determined (Koch et al., 2010, Wichern et al., 2009). The gas calibration is dependent on the portion by which the influent COD (of the composite material) fractionates into carbohydrates, lipids, proteins and inerts. The fraction of non-biodegradable COD denotes the portion of inerts in the system. The fodder analysis was applied to calculate the fractioning of the total COD by determining the total solids (TS), volatile solids (VS) and nutrition content of the three selected lignocellulosic materials (Koch et al., 2010). The Weender analysis further fractionates the grass to determine raw fiber (RF), raw lipids (RW), raw proteins (RP) and N-free extract (NfE), from which the carbohydrate content was calculated from the sum of raw fiber (RF) and N-free extracts (NfE) as shown in Fig.2. The extension of the Van Soest method describes the continued fractioning of grass into neutral detergent fiber (NDF), acid detergent fiber (ADF) and acid detergent lignin (ADL) (Van Soest and Wine, 1967, Koch et al., 2010, Biernacki et al., 2013). The ash and water content of the grass does not contain COD and is therefore not considered in the ADM1 model. The compositions (wt.%) of the grass was determined using the van Soest (Van Soest et al., 1991) method by NviroTek Laboratories, South Africa. The composition of maize silage and green weed silage was taken from literature (Biernacki et al., 2013). The influent characteristics are given in Table 1.

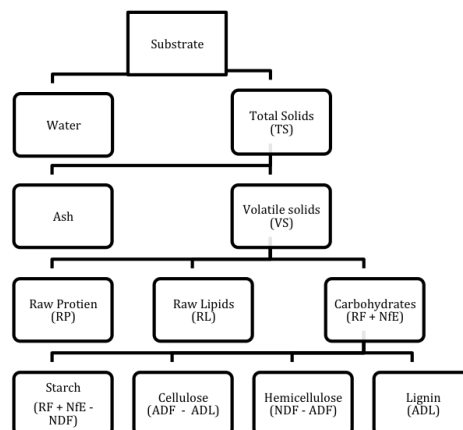


Fig.2 Determination of substrate composition by extended Weender analysis (Koch et al., 2010)

Table 1 Main components of grass bales, maize silage and green weed silage

Parameter	Unit	Grass Bale	Maize Silage	Green weed silage
Volatile solids VS	%TS	88.13	93.2	88.9
Raw protein RP	%TS	5.32	10.3	12.0
Raw fiber RF	%TS	35.17	15.5	30.8
Raw lipid RL	%TS	1.60	5.1	4.5
NfE*	%TS	46.04	62.4	41.6
NDF	%TS	60.65	71.0	65.0
ADF	%TS	38.80	33.4	47.2
ADL	%TS	15.76	11.6	3.2
Reference			Biernacki et al. (2013)	Biernacki et al. (2013)

Gerike (1984) described the oxidation of organic compounds as shown in Eq. (1) and Eq. (2). It can be used to calculate the theoretical oxygen demand (ThOD) of the represented organic compounds.



$$ThCOD = \frac{16(2a + 0.5(b - 3d) - c)}{12a + b + 16c + 14d} \left[\frac{g_{O_2}}{g_{C_2H_4O_2N_d}} \right] \quad (2)$$

Koch et al. (2010) recommended that the composite material (X_c) could be determined by the sum of the fraction and the ThOD of each component (carbohydrates, proteins, lipids and inerts) of the grass as given by Eq. (3).

$$X_c = r_{substrate} \times \left((RF + NfE - ADL) \times ThOD_{ch} + (RP \times ThOD_{pr}) + (RL \times ThOD_{li}) + (ADL \times ThOD_i) \right) \left[\frac{kgCOD}{m^3} \right] \quad (3)$$

The stoichiometric f-factors of the particulate products of the grass is determined by dividing the relative components of the grass by volatile solids (VS) to ensure the sum of all factors is equal to 1. Koch et al. (2010) defined the carbohydrate fraction is the fraction of starch and readily degradable cellulose and hemicellulose ((NDF-ADL)·d). Where d denotes the percentage of degradable cellulose and hemicellulose. The fraction of inerts is the lignin content and the non-degradable part of cellulose and hemicellulose ((NDF-ADL)·(1-d)) (Koch et al., 2010).

$$f_{pr_xc} = \frac{RP}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad (4)$$

$$f_{li_xc} = \frac{RL}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad (5)$$

$$f_{ch_xc} = \frac{(RF + NfE - NDF) + (NDF - ADL) \times d}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad (6)$$

$$f_{i_xc} = \frac{ADL + (NDF - ADL) \times (1 - d)}{VS} \left[\frac{kgCOD}{kgCOD} \right] \quad (7)$$

Where d is estimated to be 0.35. Methods for determining d are described by Koch et al. (2009). The stoichiometric factors for the particulate products through the disintegration of the three lignocellulosic material based on Eqs. (4)-(7) are shown in Table 2. All nomenclature was adapted from Batstone et al. (2002).

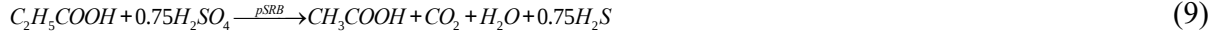
Table 2 Stoichiometric f-factor of disintegration of the composite material (X_c)

Parameter	Description	Grass Bale	Maize Silage	Green weed silage	Unit
f_{ch_xc}	Carbohydrates from X_c	0.412	0.695	0.604	-
f_{pr_xc}	Protein from X_c	0.060	0.110	0.135	-
f_{li_xc}	Lipids from X_c	0.018	0.055	0.051	-
f_{i_xc}	Inerts from X_c	0.510	0.140	0.210	-
Reference			Biernacki et al. (2013)	Biernacki et al. (2013)	

2.2 Adapting the anaerobic digestion model No.1 (ADM1) to include sulfate reduction

Organic deficient wastewaters with SO_4^{2-}/COD ratios above 0.1 use VFAs as an electron donor (Batstone, 2006). The extension of the ADM1 model developed by Fedorovich et al. (2003) included dynamic states for sulfates (S_{SO4}), total aqueous (S_{H2S}), free sulfides ($S_{H2S,free}$), sulfides in the gas phase ($S_{gas,h2s}$) and sulfide reducers (X_{ISRB}). Fedorovich et al. (2003) suggested that fast-growing acidogenic bacteria outcompetes SRB for sugars and amino acids. Therefore ADM1 was modified to incorporate the competition for VFAs by acetogenic bacteria and SRB as well as the competition for acetate and H_2 by methanogenic archaea and SRB. The four groups of SRB included into the extension of the ADM1 include butyrate-degrading (X_{BSRB}), propionate-degrading (X_{PSRB}),

acetate-degrading (X_{aSRB}) and hydrogenotrophic-degrading sulfur-reducing bacteria (X_{hSRB}) as shown in Eqs. (8) to (11) respectively.



The kinetic rate of the DSR process takes into account the concentrations of the electron acceptor (SO_4^{2-}) and the electron donor (VFAs and H_2) as shown in Eq. (12) (Fedorovich et al., 2003).

$$r_j = \frac{k_{max} S_i}{K_{s_i} + S_i} \cdot \frac{S_{SO_4}}{K_{SO_4} + S_{SO_4}} \cdot X \cdot I_{pH} \cdot I_{sulfide} \quad (12)$$

Values for kinetic rates are given in Fedorovich et al. (2003)). Kalyuzhnyi and Fedorovich (1998) recommended that the inhibition function ($I_{sulfide}$) for first order inhibition kinetics should be determined by Eq. (13). This function however becomes unstable at high concentrations of undissociated H_2S . A more stable function, Eq. (14), allows for the function to gradually approach zero (Poinapen and Ekama, 2010). The inhibition function of undissociated H_2S should be considered on all processes of the extended ADM1 model excluding disintegration and hydrolysis.

$$I_{sulfide} = 1 - \frac{S_I}{K_I} \quad (\text{if } S_I > K_I, I_{sulfide} = 0) \quad (13)$$

$$I_{sulfide} = \exp \left[- \left(\frac{S_I}{0.60056 K_I} \right)^2 \right] \quad (14)$$

The pH inhibition function takes into account the upper and lower pH limits on each microbial group and is modeled according to the Hill function as proposed by Rosén and Jeppsson (2005). Sulfuric acid (H_2SO_4) completely dissociates as it is a strong acid ($pK_a < -2$) and sulfide ions (S^{2-}) is considered to be present in small quantities in the liquid phase ($pK_a = 14$), therefore it was assumed to be negligible (Barrera et al., 2015) and were excluded from the model. The acid-base reactions Eq. (15) and Eq. (16) were included in determining the overall charge balance.



The kinetic-rates for the acid-base reactions are described by Eqs. (17) to (18) where the acid-base $K_{A/B}$ was set to $10^{10} \text{ m}^3 \text{ kmol}^{-1} \text{ d}^{-1}$ as suggested by Rosén and Jeppsson (2005). The rate of gas-liquid transfer of H_2S is described by Eq. (19).

$$r_j = K_{A/B, H_2S} \left(S_{HS} \left(S_{H^+} + K_{a, H_2S} \right) - K_{a, H_2S} \times S_{H_2S, total} \right) \quad (17)$$

$$r_j = K_{A/B, SO_4^{2-}} \left(S_{SO_4^{2-}, total} \times S_{H^+} - \left(K_{a, SO_4^{2-}} + S_{H^+} \right) \right) \quad (18)$$

$$r_j = k_L a \left(S_{H_2S, free} - K_{H, H_2S} \times P_{gas, H_2S} \right) \quad (19)$$

2.3 Description of model

The proposed model describes a two-phase digestion system that was simulated in Matlab/Simulink R2016a. The first phase includes the disintegration of the composite material and hydrolysis in sequentially fed leach bed reactors (SLBR). The second phase utilises an upflow anaerobic sludge bed (UASB) for acidogenesis, acetogenesis, methanogenesis and sulfate reduction demonstrated in Fig. 3. The combination of the SLBR-UASB has not previously been developed to full scale and thus this research also investigates the potential of utilising this technology to treat AMD at a full-scale operation.

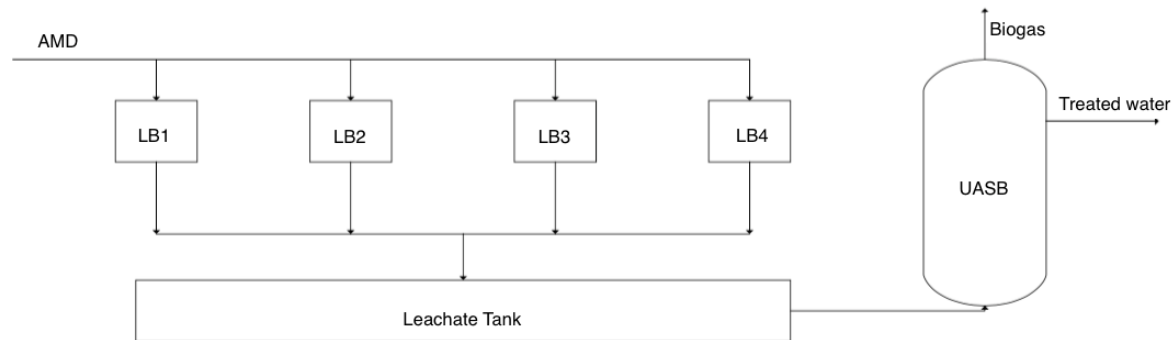


Fig.3 Process flow diagram of sequentially fed leach bed reactors coupled with an upflow anaerobic sludge bed

2.4 Mathematical models

2.4.1 Sequentially fed leach bed reactor

Extracellular biological and non-biological processes drive the breakdown, disintegration and hydrolysis of the composite material (X_c) into its respective particulates (X_i) and soluble (S_i) components, which is assumed to be a first order kinetic reaction. The disintegration rate coefficient (k_{dis}) and hydrolysis rate coefficients (k_{hyd}) are given in Table 3.

Table 3 Kinetic constants sourced from literature for the breakdown of grass under mesophilic conditions

Description		Grass Bales	Maize Silage	Green weed silage	Unit	Reference
Disintegration of grass	k_{dis}	1.7433	0.7705	0.8168	[d ⁻¹]	Biernacki et al. (2013)
Hydrolysis of carbohydrates	k_{hyd_ch}	0.7366	0.6865	0.6659	[d ⁻¹]	Biernacki et al. (2013)
Hydrolysis of proteins	k_{hyd_pr}	0.0104	0.2446	0.0014	[d ⁻¹]	Biernacki et al. (2013)
Hydrolysis of lipids	k_{hyd_li}	0.0149	0.1216	0.513	[d ⁻¹]	Biernacki et al. (2013)

The leach bed reactor was modeled as a batch process for the particulate components as a sieve retains all particulate components in the reactor.

$$\frac{dX_{C,n}}{dt} = -k_{dis}X_{C,n} \quad (20)$$

$$\frac{dX_{i,n}}{dt} = k_{dis}X_{C,n}f_{i,xc} - k_{hyd,i}X_{i,n} \quad (21)$$

Eqs. (20) and (21) describe the break down of the composite material and the break down of the carbohydrates, proteins and lipids respectively. The fermentation monomers (monosaccharaides,

amino acids and long fatty chain acids (LFCA)) produced during hydrolysis are soluble particulates and were assumed in the model to pass into the leachate tank through the sieve with unreacted sulfates.

2.4.2 Leachate Tank

Without the presence of an inoculum and due to the fact that only soluble components from the leach bed is allowed to enter the leachate tank (LT), the LT was modeled as a CSTR (Thamsiroj et al., 2012). As AD does not take place in the LT the continuity equation allowed for the inflow and outflow terms but disregarded the process rate term (Thamsiroj et al., 2012).

$$\frac{dS_{i,LT}}{dt} = -\frac{q_{UASB,in}S_{i,LT}}{V_{LT}} + \frac{q_{LB,n}}{V_{LT}} \sum_{n=1-2} \dot{a} \frac{dS_{i,n}}{dt} \quad (22)$$

2.4.3 Upflow Anaerobic Sludge Bed

This research aimed to analyse if a full-scale operational scheme of a SLBR-UASB would be effective in treating AMD. High-rate UASBs are associated with lower costs, land usages and environmental impacts compared to other existing technologies (Batstone, 2006). The hydraulics of a full-scale UASB was shown to behave as a CSTR (Batstone et al., 2005). Therefore UASB was modeled according to the CSTR dynamic rate equation. The material balance took into account ideal mixing for soluble and gas components in the UASB (Fedorovich et al., 2003) are expressed by Eqs. (22) and (23) respectively. The soluble and particulate components i according to the nomenclature of Batstone et al. (2002) are given in Table 4.

$$\frac{dS_{liq,i,n}}{dt} = \frac{q_{LB,n}}{V_{liq,n}} (S_{in,i,n} - S_{liq,i,n}) + \dot{a} r_{j,i,j} \quad (22)$$

$$\frac{dS_{gas,i}}{dt} = -\frac{S_{gas,i}q_{gas}}{V_{gas}} + r_{T,i} \frac{V_{liq}}{V_{gas}} \quad (23)$$

The term V_{liq}/V_{gas} takes into account the liquid volume dependency of the gas transfer kinetic rates. The gas flow rate of the UASB is determined by:

$$q_{gas} = \frac{RT}{P_{gas} - P_{gas,H_2O}} \cdot V_{liq} \left(\frac{r_{T,H_2}}{16} + \frac{r_{T,CH_4}}{64} + r_{T,CO_2} + \frac{r_{T,H_2S}}{64} \right) \quad (24)$$

The solids retention time (SRT) in the UASB is greater than the hydraulic retention time (HRT) and the needs to be accounted for in the system. Kalyuzhnyi and Fedorovich (1998) expressed the SRT as a function of both the HRT and the upward liquid velocity (U_{up}). Where ER_i is the retention efficiency of the corresponding bacterial group. Fedorovich et al. (2003) determined k_v to be 0.026 and the values for K_{ER}^i for each bacterial group.

$$\frac{1}{SRT} = \frac{q_{liq} \cdot [1 - ER_i(U_{up})]}{V_{liq}} \quad (25)$$

$$ER_i = K_{ER}^i \left[1 - k_v \log_{10} \left(\frac{U_{up}}{2} \right) \right] \quad (26)$$

The material balance for the particulate components in the UASB is described by Eq. (27) where the process rate term includes the decay of each microbial group.

$$\frac{dX_{liq,i}}{dt} = -\frac{q_{liq}X_i}{V_{liq}} [1 - ER_i(U_{up})] + \sum r_{j,i,j} \quad (27)$$

Table 4 State Variables

Soluble Substrates			Particulate Substrates		
i		Unit	i		Unit
1	Monosaccharaides (S_{su})	kg COD m^{-3}	14	Composites (X_c)	kg COD m^{-3}
2	Amino acids (S_{aa})	kg COD m^{-3}	15	Carbohydrates (X_{ch})	kg COD m^{-3}
3	LFCA (S_{fa})	kg COD m^{-3}	16	Proteins (X_{pr})	kg COD m^{-3}
4	Total valerate (S_{va})	kg COD m^{-3}	17	Lipids (X_{li})	kg COD m^{-3}
5	Total butyrate (S_{bu})	kg COD m^{-3}	18	Sugar degraders (X_{su})	kg COD m^{-3}
6	Total propionate (S_{pro})	kg COD m^{-3}	19	Amino acid degraders (X_{aa})	kg COD m^{-3}
7	Total acetate (S_{ac})	kg COD m^{-3}	20	LFCA degraders (X_{fa})	kg COD m^{-3}
8	Hydrogen (S_{h2})	kg COD m^{-3}	21	Valerate & butyrate degraders (X_{c4})	kg COD m^{-3}
9	Sulfate (S_{so4})	kmole m^{-3}	22	Butyrate SRB (X_{bSRB})	kg COD m^{-3}
10	Methane (S_{ch4})	kg COD m^{-3}	23	Propionate degraders (X_{pro})	kg COD m^{-3}
11	Total hydrogen sulfide (S_{h2s})	kmole m^{-3}	24	Propionate SRB (X_{pSRB})	kg COD m^{-3}
12	Inorganic carbon (S_{ic})	kmole C m^{-3}	25	Acetate degraders (X_{ac})	kg COD m^{-3}
13	Inorganic nitrogen (S_{in})	kmole N m^{-3}	26	Acetate SRB (X_{aSRB})	kg COD m^{-3}
Ion State Variable			27	Hydrogen degraders (X_{h2})	kg COD m^{-3}
i		Unit	28	Hydrogenotrophic SRB (X_{hSRB})	kg COD m^{-3}
30	Cations (S_{cat})	kmole m^{-3}	29	Particulate inerts (X_i)	kg COD m^{-3}
32	Anions (S_{an})	kmole m^{-3}	Gas State Variable		
33	Hydrogen ion (S_{h+})	Kmole $H^+ m^{-3}$		Hydrogen ($S_{gas,h2}$)	kg COD m^{-3}
				Methane ($S_{gas,ch4}$)	kg COD m^{-3}
				Carbon dioxide ($S_{gas,co2}$)	kmole C m^{-3}
				Hydrogen Sulfide ($S_{gas,h2s}$)	kmole m^{-3}

3. Results and Discussion

The dynamic model simulated in Matlab/Simulink R2016a analysed the process over a 50-day period. The organic loading rate was kept constant at 10 kg COD $m^{-3} d^{-1}$ for the duration of the process as suggested by Lettinga and Pol (1991). The UASB was modeled to have an upward velocity of 1.5 $m s^{-1}$ and a HRT of 3.3 days. The initial state values were taken from steady state operating values and the characteristics of the lignocellulosic material given in Table 2. The proposed system was modeled to operate at 35°C and all kinetic constants were obtained from Batstone et al. (2002) and Fedorovich et al. (2003). The simulation of the SLBR-UASB configuration initially kept the sulfate concentration of the influent to a maximum of 30 000 mg/L in order to evaluate the efficiency of sulfate reduction for all three types of lignocellulosic materials, with maize silage having the highest reduction of up to 97.8%. The reduction of the sulfate concentrations for the organic carbon sources of grass bales and green weed silage were 94.1% and 93.7% respectively shown in Fig. 4. The conversion of sulfates to sulfides was directly related to the available amount of produced VFA's in the system shown in Fig. 4. Even though Maize silage had the lowest disintegration rate (k_{dis}), which is considered the rate limiting step of the ADM1 process (Batstone et al., 2002), the characteristics of higher protein and lipid content, greater hydrolysis rates and the highest composition of readily degradable carbohydrates led to greater conversion of the composite material to fermentation monomers. The steep incline of VFA's produced correlated to the amount of soluble H_2S in the system. The acidogenesis and acetogenesis steps are inhibited by the increasing concentration of hydrogen sulfide in the system.

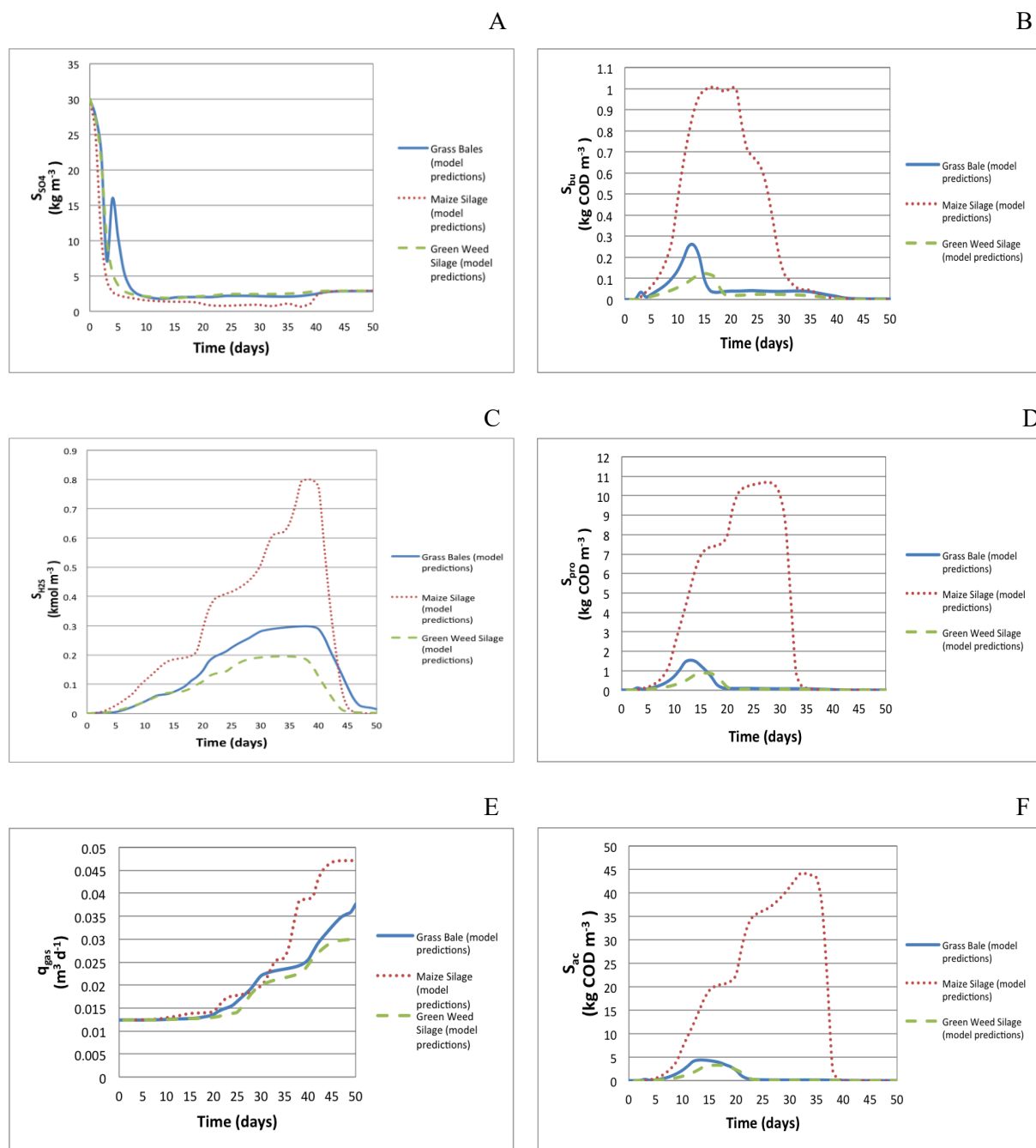


Fig.4 Model predictions: (A) S_{SO4} , (B) S_{bu} , (C) S_{H2S} , (D) S_{pro} , (E) q_{gas} , (F) S_{ac}

The influent sulfate concentration was varied to establish the effect of the SO_4^{2-} :COD ratio could have on methane production. The sulfate concentration in AMD varies from site to site and plays a significant role in methane production. Three influent SO_4^{2-} concentrations were tested i.e. 30 mg/L, 3 000 mg/L and 30 000 mg/L in order to analyse the production of methane at low concentrations of sulfate and at the lower and upper limit of sulfate concentrations of AMD as described by Christensen et al. (1996). There is a direct correlation between the inhibition of free hydrogen sulfide on methanogenesis and the SO_4^{2-} /COD ratio as shown in Fig.5.

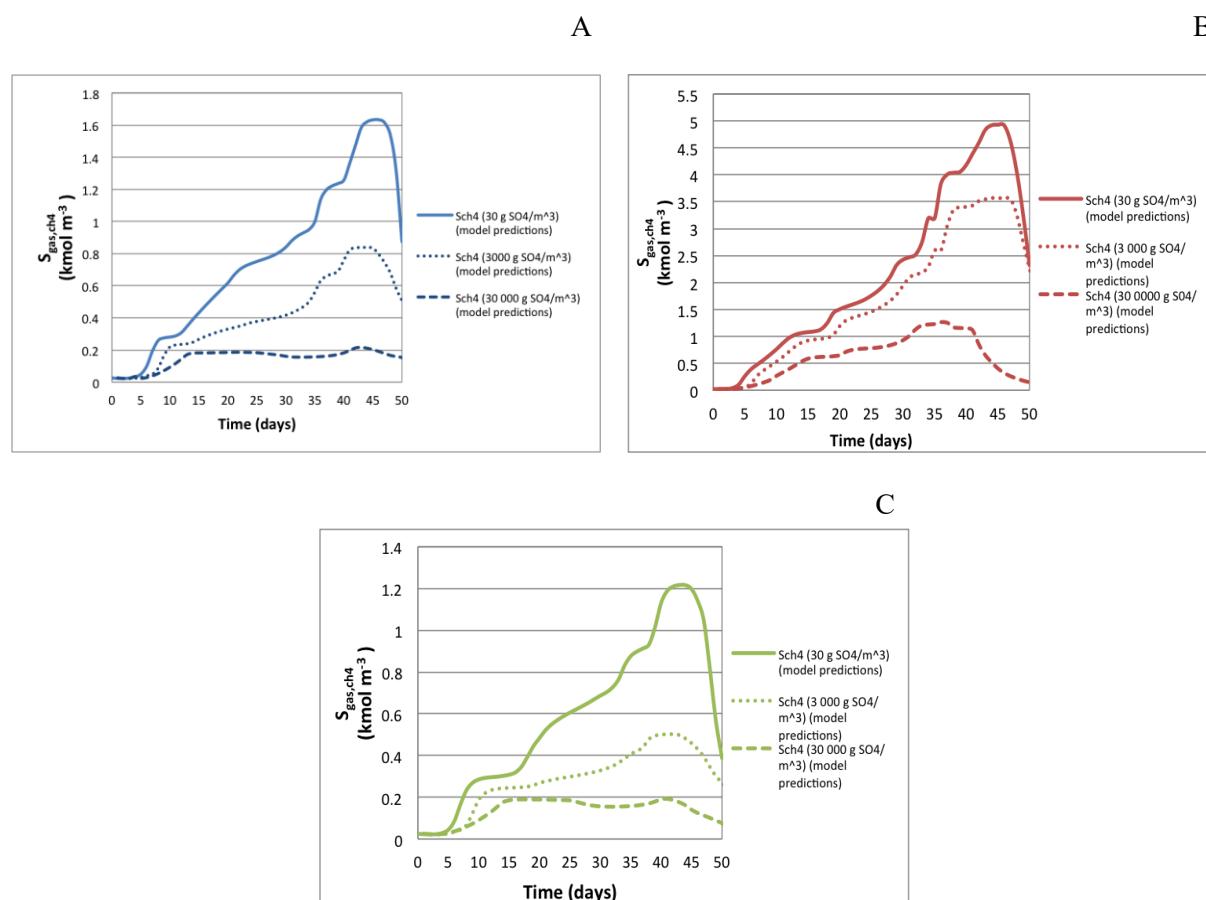


Fig.5 Model predictions of S_{ch4} with an influent sulfate concentration of 30 mg/L, 3 000 mg/L and 30 000 mg/L: (A) grass Bales, (B) maize silage, (C) green weed silage

From Fig.5 it is evident that maize silage yields higher concentrations of methane ($S_{\text{gas,CH}_4}$) than grass silage and green weed silage. When the influent sulfate concentration was set to be a minimum (30mg/L) the maximum possible energy rate produced by maize silage was $1\,104\text{ kWh d}^{-1}$ while grass silage and green weed silage yielded an energy production rate of 365.8 kWh d^{-1} and 272.3 kWh d^{-1} respectively. This is due to a larger availability of the organic carbon source (VFAs) and H_2 , produced by the breakdown of the maize silage, utilised during methanogenesis. Higher concentrations of sulfates coming into the system generates higher free form hydrogen sulfide concentration which leads to a lower production of methane gas as shown in Fig.5. The steep increasing gradient of methane at low concentrations of sulfates (30 mg/L) relates to the stepwise increase of methanogens in the bioreactor (Omil et al., 1996, Omil et al., 1997, Fedorovich et al., 2003). At higher influent concentrations of SO_4^{2-} (3 000mg/L and 30 000 mg/L) the incline of this



gradient significantly reduces as the SRB out compete methanogens for VFA's and H_2 leading to a slower and reduced growth rate of methanogenic archaea. At sites containing higher influent concentrations of sulfates (SO_4^{2-}) it is no plausible for the proposed process to be self-sufficient in meeting energy demands as methane production is inhibited by the greater generation of sulfides in the system.

4. Conclusion

This research aimed to conduct a feasibility study of the SLBR-UASB technology for the full-scale operation of the bioremediation of AMD. The ADM1 was adapted to include sulfate reduction and analysed the potential of three different lignocellulosic materials (grass bales, maize silage and green weed silage) as the organic carbon source for the DSR process. Processes fed with maize silage were shown to have the highest sulfate reduction efficiency (97.8%) and yielded greater methane production. This was due to the higher concentrations of VFAs released by maize silage during acidogenesis and acetogenesis that could be used by methanogens and SRB. Increasing the SO_4^{2-} :COD ratio significantly impacts on methane production as the increased concentration of free hydrogen sulfides inhibits the process of methanogenesis and SRB out-compete methanogens for VFAs and H_2 . The model was simulated in Matlab/Simulink R2016a and is currently undergoing validation.

5. Acknowledgements

The authors would like to acknowledge the Global Change and Sustainability Research Institute (GCSRI), the National Research Fund (NRF) and the Industrial and Mining Water Research Unit (IMWaRU) for financially supporting this research.

7. References

- BARRERA, E. L., SPANJERS, H., DEWULF, J., ROMERO, O. & ROSA, E. 2013. The sulfur chain in biogas production from sulfate-rich liquid substrates: a review on dynamic modeling with vinasse as model substrate. *Journal of Chemical Technology & Biotechnology*, 88, 1405-1420.
- BARRERA, E. L., SPANJERS, H., SOLON, K., AMERLINCK, Y., NOPENS, I. & DEWULF, J. 2015. Modeling the anaerobic digestion of cane-molasses vinasse: Extension of the Anaerobic Digestion Model No. 1 (ADM1) with sulfate reduction for a very high strength and sulfate rich wastewater. *Water Research*, 71, 42-54.
- BATSTONE, D. J. 2006. Mathematical Modelling of Anaerobic Reactors Treating Domestic Wastewater: Rational Criteria for Model Use. *Reviews in Environmental Science and Bio/Technology*, 5, 57-71.
- BATSTONE, D. J., HERNANDEZ, J. & SCHMIDT, J. E. 2005. Hydraulics of laboratory and full - scale upflow anaerobic sludge blanket (UASB) reactors. *Biotechnology and bioengineering*, 91, 387-391.
- BATSTONE, D. J., KELLER, J., ANGELIDAKI, I., KALYUZHNYI, S., PAVLOSTATHIS, S., ROZZI, A., SANDERS, W., SIEGRIST, H. & VAVILIN, V. 2002. The IWA Anaerobic Digestion Model No 1(ADM 1). *Water Science & Technology*, 45, 65-73.
- BIERNACKI, P., STEINIGEWEG, S., BORCHERT, A. & UHLENHUT, F. 2013. Application of Anaerobic Digestion Model No. 1 for describing anaerobic digestion of grass, maize, green weed silage, and industrial glycerine. *Bioresource Technology*, 127, 188-194.
- CHRISTENSEN, B., LAAKE, M. & LIEN, T. 1996. Treatment of acid mine water by sulfate-reducing bacteria; results from a bench scale experiment. *Water Research*, 30, 1617-1624.
- FEDOROVICH, V., LENS, P. & KALYUZHNYI, S. 2003. Extension of Enaerobic Digestion Model No. 1 with processes of sulfate reduction. *Applied Biochemistry and Biotechnology*, 109, 33-45.
- GERIKE, P. 1984. The biodegradability testing of poorly water soluble compounds. *Chemosphere*, 13, 169-190.



International Conference on Energy, Environment and Climate - ICEEECC 2017

- KALYUZHNYI, S. V. & FEDOROVICH, V. V. 1998. Mathematical modelling of competition between sulphate reduction and methanogenesis in anaerobic reactors. *Bioresource Technology*, 65, 227-242.
- KOCH, K., LÜBKEN, M., GEHRING, T., WICHERN, M. & HORN, H. 2010. Biogas from grass silage – Measurements and modeling with ADM1. *Bioresource Technology*, 101, 8158-8165.
- KOCH, K., WICHERN, M., LÜBKEN, M. & HORN, H. 2009. Mono fermentation of grass silage by means of loop reactors. *Bioresource Technology*, 100, 5934-5940.
- LETTINGA, G. & POL, L. H. 1991. UASB-process design for various types of wastewaters. *Water science and technology*, 24, 87-107.
- LINDSAY, M. B. J., BLOWES, D. W., CONDON, P. D. & PTACEK, C. J. 2011. Organic carbon amendments for passive in situ treatment of mine drainage: Field experiments. *Applied Geochemistry*, 26, 1169-1183.
- OMIL, F., LENS, P., HULSHOFF POL, L. & LETTINGA, G. 1996. Effect of upward velocity and sulphide concentration on volatile fatty acid degradation in a sulphidogenic granular sludge reactor. *Process Biochemistry*, 31, 699-710.
- OMIL, F., LENS, P., POL, L. W. H. & LETTINGA, G. 1997. Characterization of biomass from a sulfidogenic, volatile fatty acid-degrading granular sludge reactor. *Enzyme and Microbial Technology*, 20, 229-236.
- POINAPEN, J. & EKAMA, G. 2010. Biological sulphate reduction with primary sewage sludge in an upflow anaerobic sludge bed reactor-Part 6: Development of a kinetic model for BSR. *Water SA*, 36, 203-214.
- RAMLA, B. & SHERIDAN, C. 2015. The potential utilisation of indigenous South African grasses for acid mine drainage remediation. *Water SA*, 41, 247-252.
- ROSÉN, C. & JEPPSSON, U. 2005. Aspects on ADM1 Implementation within the BSM2 Framework. *TEIE*, 7224, Department of Industrial Electrical Engineering and Automation, Lund Institute of Technology.
- THAMSIRIROJ, T., NIZAMI, A. S. & MURPHY, J. D. 2012. Use of modeling to aid design of a two-phase grass digestion system. *Bioresource Technology*, 110, 379-389.
- VAN SOEST, P. J., ROBERTSON, J. B. & LEWIS, B. A. 1991. Methods for Dietary Fiber, Neutral Detergent Fiber, and Nonstarch Polysaccharides in Relation to Animal Nutrition. *Journal of Dairy Science*, 74, 3583-3597.
- VAN SOEST, P. U. & WINE, R. 1967. Use of detergents in the analysis of fibrous feeds. IV. Determination of plant cell-wall constituents. *J. Assoc. Off. Anal. Chem*, 50, 50-55.
- WICHERN, M., GEHRING, T., FISCHER, K., ANDRADE, D., LÜBKEN, M., KOCH, K., GRONAUER, A. & HORN, H. 2009. Monofermentation of grass silage under mesophilic conditions: Measurements and mathematical modeling with ADM 1. *Bioresource Technology*, 100, 1675-1681.