Simulation of mine production using MATLAB and AweSim

Samira Kalantari[1] and Hooman Askari-Nasab

1. Introduction

Abstracting a real system into a system which is not real is called simulation; in other words, simulation is the mathematical representation of the interaction of real-world objects. A system can be specified by a set of variables where each variable can describe the state of the system and changes in the system from state to state. A simulation model involves considering the behavior of the model by changing it from state to state. In case of formulating a problem, simulation should be amongst the first approaches considered, and it's a way of developing the level of understanding of the interactions between different parts of the system (Alan et al., 1999).

1.1. Discrete event simulation

Generally to simulate a system, a mathematical model of the system should be created. Changes in the state of the system can happen over continuous time or at discrete points of time, therefore there are three types of simulation models: continuous, discrete-event simulation, and a combination of two (Raczynski, 2009).

In discrete event simulation the models are restricted to discrete-event models. In a discrete-event model there is a finite number of transitions overtime. In a discrete-event simulation, the goal is defining the states of a system and constructing the activities that move from a state to another state. In other words, discrete-event simulation models the system as a set of individual entities which move along the system in discrete time (Tako and Robinson, 2009).

2. Problem definition

2.1. Simulation of mine scheduling

In this project, a discrete-event simulation model will be used to simulate the mine production schedule. The mine schedule is simulated by the order of extraction of blocks in each period of mining. Four disposal locations are possible for the extracted material, waste dumps, crusher, low-grade stockpiles, and high-grade stockpiles. In this model, the trucks will have different traveling speeds, and payloads. The governing relationships between these objects will be simulated. There are some unexpected events which can make a delay in the schedule such as equipment breakdowns and weather conditions that can be included in the model.

[1] MSc candidate in Mining Engineering

We will use AweSim simulation software (Alan et al., 1999) to model the mine production process. AweSim includes the Visual SLAM language, which we will use to create networks, sub networks, discrete event and continuous models of the mine production. Visual SLAM network structures consist of nodes and branches, which are used to build a network that represents the mining process.

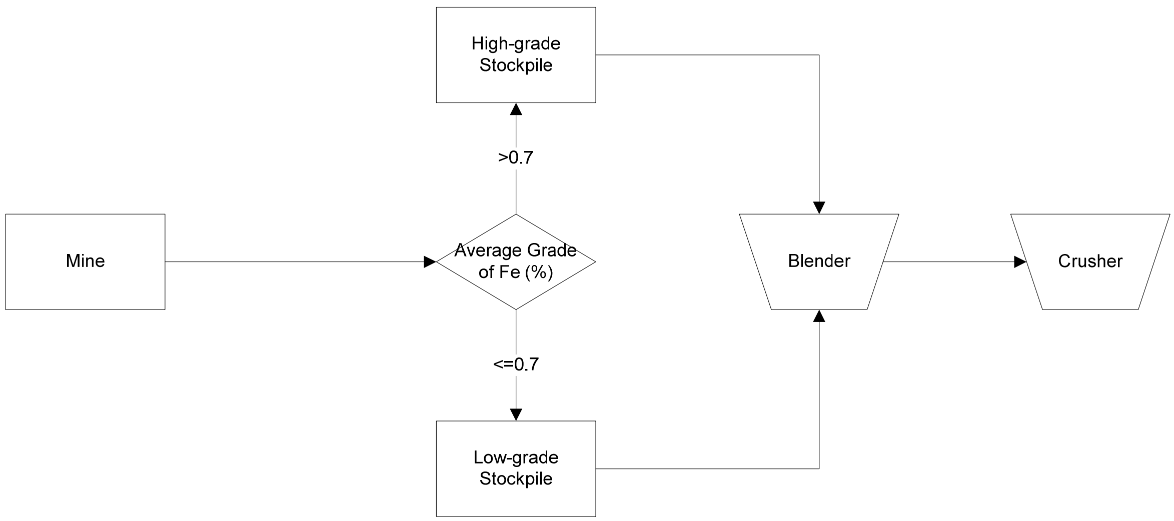


Figure 1: Schematic diagram of the simulation process

Figure 1 shows the schematic diagram of the simulation process, the blocks coming from the mine go to either stockpile 1 ore stockpile 2 according to their average grade of Fe. Different portions of material will go to the blender according to the desired cut-off grade of Fe. Afterwards, the blocks will go to the crusher. We want to simulate this process, and see how the breakdown of blender and crusher will affect the whole system.

3. Methodology

3.1. Building the network

In this study we have used an iron ore block model. Every block has its own properties such as: average grade of ore, ore tonnage, waste tonnage, economic block value, etc. The block model contains 8456 blocks. Three types of blocks are defined within the block model, ore, waste, and air blocks. The following attributes are modeled for each block: coordinates, ore tones, block tonnage, grade of Fe, grade of P, grade of S, etc.

The waste blocks are those that economic block values are negative. The economic block value for the air blocks is equal to zero.

First, we imported the block model data into MATLAB. The imported data is saved within a MATLAB *struct* data structure. The struct consists of one row and 8456 (number of blocks) columns. For statistical analysis we needed to separate different block types. We developed a code in MATLAB code to separate different block type's data. The code for removing waste and air blocks is represented in appendix A.

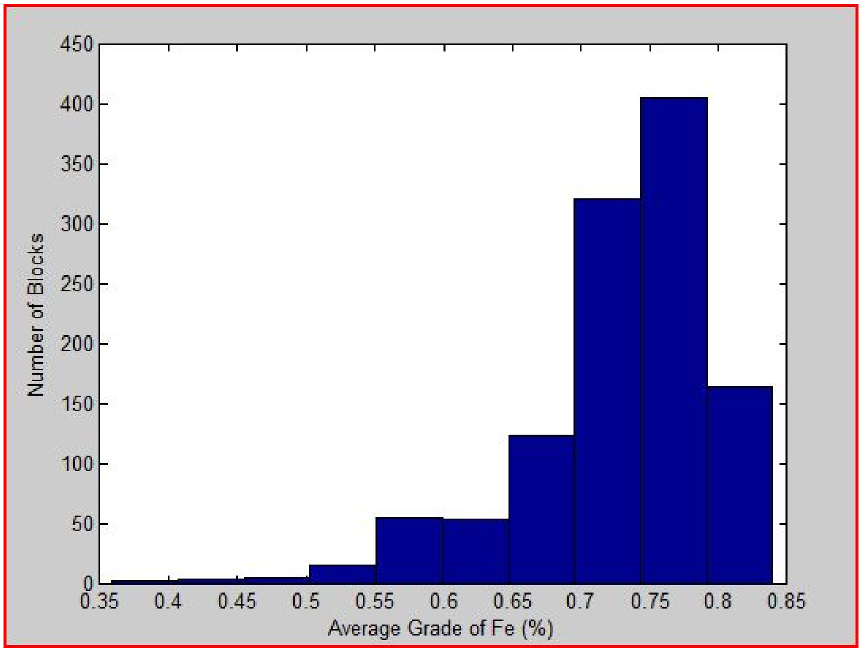


Figure 2: Histogram of the average grade of Fe (%) for blocks

Figure 1 illustrates the histogram of grade distribution of Fe% in the block model. We have used the histogram in Figure 2 to fit a probability density function. In simulation with AweSim, a distribution to represent the data is used where the distribution provides more information than the direct use of data. The question is whether or not the distribution fits the corresponding data. Using the difittool in Matlab, the best distribution on data was found, which is shown in figure which is the Weibull distribution 2.

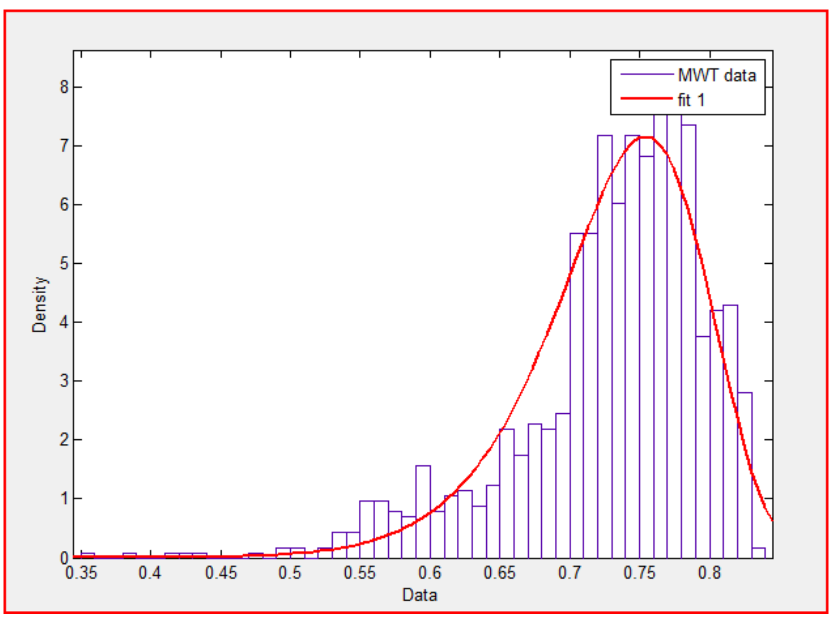


Figure 3: Weibull distrubution

Table 1: Data from the distribution of figure 2.

DISTRIBUTION	MEAN	VARIANCE	STD.	β	α
Weibull	0.7313	0.0037	0.0608	0.7570	14.7234

Similarly, using MATLAB we should find the tonnage of each block. In order to import this data in Awesim, we fit a distribution on the tonnage of the blocks. The best distribution fitting on this data (block tonnage) is the normal distribution, the data for this distribution is provided in table 2.

Table 2: Data for the distribution for block tonnage.

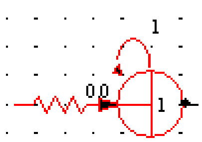
DISTRIBUTION	MEAN	VARIANCE	STD.
Normal	705180	6.53e+10	255572.49

As this project deals with the section moving from the stockpiles to the crusher, we can have two different stockpiles according to the grade of Fe, low-grade stockpile and high-grade stockpile. A condition can be set in order to separate the blocks according to their average grade of Fe.

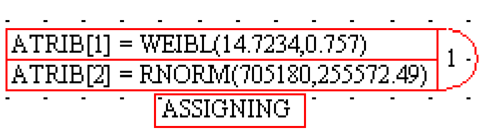
If average grade of Fe in the block >0.7 → goes to the high-grade stockpile

If average grade of Fe in the block <=0.7→ goes to the high-grade stockpile

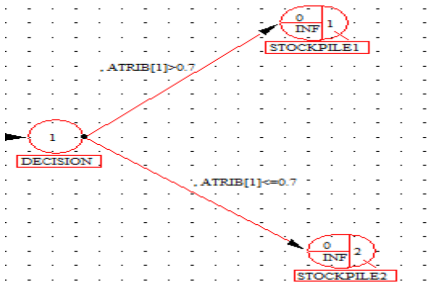
In Awesim, after creating the entities with the create node, the time between entities can be considered, in this project, the entities are assumed to be block models, and the time between created entities can be set to 1 unit of time. Create node generates entities and routes them into the system over activities emanating from this node.



After entering of the blocks to the system, the grade of Fe and the tonnage of each block should be assigned to it. Using the distributions fitted to the data, the ATRIB[1] is the average grade of Fe of each block, and ATRIB[2] is the tonnage of each block. Assign node is used to assign values to the activities which are passing through this node. Therefore, the assign node should be as follows.



After assigning the average grade and the ore tonnage, the blocks can be going to either the low-grade and high-grade stockpiles according to ATRIB[1]. Queue node: entities wait at this node for service.



The above figure shows two queue nodes which are high-grade and low-grade stockpiles, stockpile 1 contains the blocks with the grade of Fe greater than 0.7, and stockpile 2 contains the blocks with the grade of Fe less than or equal to 0.7.

After sending the entities to two different queue nodes (stockpiles) these entities should go to the blender, to have different portions of each stockpiles in the blender, the duration for sending blocks from these two stockpiles should be different. In other words, in order to achieve the desired grade variation after being processed in the blender, the duration for sending blocks from stockpile 1 and stockpile 2 should be different.

Therefore, the duration for two different paths should vary in such a way that the final grade variation is in the desired range. In order to count the number of blocks going to blender from stockpile 1 and stockpile 2, an assign node was created.

$$XX[2]=XX[2]+1 \quad (1)$$

In equation (1), the initial value for XX[2] is zero, and when an entity departs from stockpile1, XX[2] is increased by 1. Similarly, an assign node should be created to count the number of blocks going from stockpile2 to the blender, which is shown in equation (2).

$$XX[3]=XX[3]+1$$

Where the initial value for XX[3] in equation (2) is zero. In addition, the ore tonnage of each block can be calculated by assigning an attribute which is the product of the average grade of Fe of the block and the block tonnage of equation (3).

$$\text{Ore Tonnage of the block} = \text{average grade of Fe} * \text{block tonnage} \quad (3)$$

Therefore, the value of XX[2] and XX[3] will be dependent on the duration of the activity. To find the average grade of the blocks which are going to the blender from the high-grade stockpile the ore tonnage of the block and the average grade should be found. In order to find the average grade of the blocks after blending, in the assign node these values should be assigned which is showed in equation (4).

$$\sum_{i=1}^{n} X_i * O_i \tag{4}$$

Where:
n, number of blocks coming from stockpile1
X_i, is the average grade of Fe for the i^{th} block
O_i, is the ore tonnage of the i^{th} block.

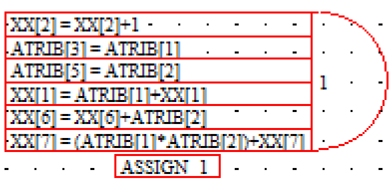
XX[7]=(ATRIB[1]*ATRIB[2])+XX[7] (5)

In equation (5), XX[7] will calculate the desired summation of the blocks coming from stockpile 1. In addition, for finding the average grade of Fe after being processed in the blender, the total tonnage coming from stockpile 1 should be calculated, which are shown in equation (6) and equation (7).

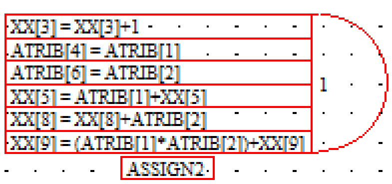
XX[8]=ATRIB[2]+XX[8] (6)

XX[9]=(ATRIB[1]*ATRIB[2])+XX[9] (7)

The assign node for stockpile 1 is as follows:



And similarly, the assign node for stockpile 2 is as follows:



The above expression is assigned with the initial value of XX[7] equal to zero. Therefore,

After the await node for the blender a collect node should be created in order to calculate the average grade of the blender. This average grade depends on the number of the blocks coming from stockpile number one, and also on the number of the blocks coming from stockpile number 2, respectively XX[2] and XX[3].

The average grade of the blender should be calculated is shown in equation (8).

$$g = \frac{\sum_{i=1}^{2} grade * tonnage}{tonnage1 + tonnage2} \tag{8}$$

The nominator of equation (8) is equal to:

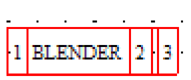
$$\sum_{i=1}^{n} X_i * O_i + \sum_{i=1}^{m} X_i * O_i \tag{9}$$

Where:
n is the number of blocks coming from stockpile1.
And m is the number of blocks coming from stockpile2.
The nominator of equation (8) is equal to:

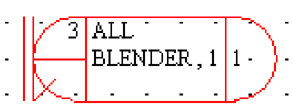
$$\sum_{i=1}^{n} O_i + \sum_{i=1}^{m} O_i \tag{10}$$

3.2. Blender

The blocks will wait in the await node for the blender, the general process of the blender is mixing the blocks to get the desired average grade of the mixture. The blocks which are waiting in the await node for the blender, will be sent to the blender, and the capacity for the blender is 2 which means that the crusher (resource) can process 2 blocks at time. And the processing time for the crusher is 7 unites of time.



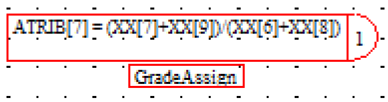
The figure above shoes the resource node which is the blender for this system; resource block identifies the resource name or label and the initial capacity for the resource. The initial capacity of the resource is 10, and the await node which will work with the resource is as follows:

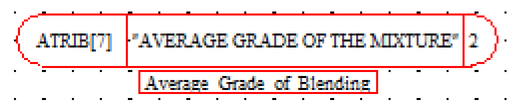


Await node stores entities waiting for units of resources. The entities are blocked in this node, which means that the entities (blocks) should stay in this node until the resource be free.

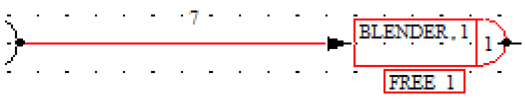
The whole process of the blender is blending the blocks coming from the two stockpiles in such a way that the grade variation of the blocks going to the crusher and afterwards to the mill be in-spec. In order to achieve the desired grade variation the duration of moving blocks from the two stockpiled should vary. The duration difference shows the difference in the number of blocks going from two stockpiles to the blender.

Colct node is a location where statistics can be collected on any expression. The colct node should calculate the average grade of blending, therefore an assign node should be created, and an attribute should be assigned to that value:



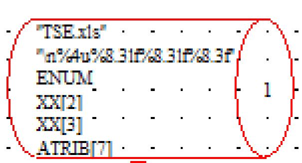


As the processing time for the blender is 7 units of time, this amount of time will take to process each block, therefore the duration for sending the blocks from the recourse to the free node is 7 units of time. Free node releases units of resource type when an entity arrives to the node.



After being processed in the blender, the blocks will be sent to the await node for the crusher. The duration for this process is assumed to be 2.5 units of time which means that 2.5 units of time will take to send the blocks from blender to the crusher.

The blocks will gather up in the await node for the crusher, and in order to send all of them to the crusher the blocking will be done. Blocking will force the blocks to wait in that await node until the crusher be free. The initial capacity for the crusher assumed to be 5, which means it can process 5 blocks at the same time, and the processing time is assumed to be 10 which means that the time for each block to be processed in the crusher is 10 units of time. In order to import the data for average grade of time, number of blocks coming from the stockpile 1, and number of blocks coming from stockpile 2, a write node is defined. Write node writes one or more values to an external file.

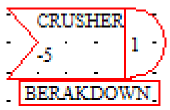


This will import this data in an Excel file.

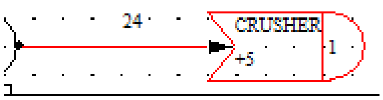
3.3. Breakdown of the crusher

It can be assumed that after 200 units of simulation time, the crusher will break down, therefore the waiting time for the blocks in the crusher await node will increase. The process of fixing and repairing the crusher will take 24 units of time, and after that the crusher will process the waiting blocks:

The process of breaking down and fixing the crusher can be simulated using the Alter node. The Alter node will change the capacity of the resource; therefore in this case, the Alter will change the capacity of the blender to -2 which means that the crusher is out of service.

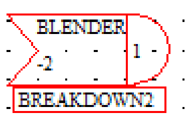


The figure above shows the Alter node which works with the crusher resource, and change the capacity of the crusher by -2, and the initial capacity for the crusher is 2, therefore this Alter node will change the resource capacity to zero which shows that the crusher is out of service. After 24 hours of time, the crusher is fixed; another Alter node should increase the crusher capacity to the initial value:



3.4. Breakdown of the blender

After 200 units of time of the simulation, the blender will be out of service and the fixing process will be 24 units of time.



The above Alter node will decrease the blender capacity to zero, indicating that the blender is out of service.

4. Implementation

4.1. Control statement

In the control statement of this simulation, the limits for the maximum index for attributes and max index for XX (global variable) should be defined. The maximum number of attributes is 7, where the maximum number of XX variable is 9.

INTLC is used to assign initial values to the variables.
INTLC,{{XX[2],0},{XX[3],0},{XX[1],0},{XX[5],0},{XX[7],0},{XX[8],0},{XX[9],0},{ XX[6],0}};

5. Results and discussions

5.1. Sensitivity analysis

5.1.1 Running without breaking down of the blender and crusher

In order to find out the influence of the crusher blender on the system, first the network without blender break down should be simulated. The network for this simulation is provided in appendix B, and the summary report of AweSim is provided in appendix C. The summary report of Awesim is indicated in table 3 and 4.

Table 3: AweSim summary report 1.

Label	Mean Value	Std.	Number of Observations	Minimum value	Maximum Value
Average grade	0.719	0.008	86	0.695	0.767
Ore Tonnage stockpile 1	35429938	20486838	100	807127	70773022
Ore Tonnage Stockpile2	20541517	114145323	89	452553	39271039
No. Blocks from STP1	49.849	29.088	86	1	99
No. Blocks from STP2	32.930	19.769	86	0	68

Table 4: AweSim summary report 2.

File Number	Label or Input Location	Average Wait Time
Queue	High-grade stockpile	86.310
Queue	Low-grade stockpile	0.435
Resource	Blender	70.337
Resource	Crusher	17.00

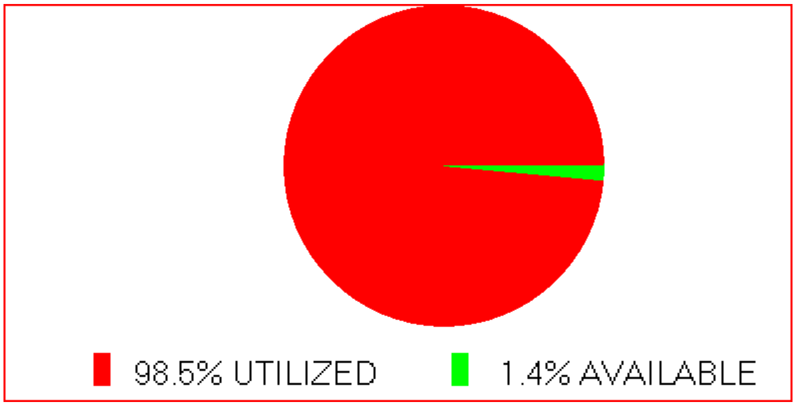


Figure 5: Utilization of the blender

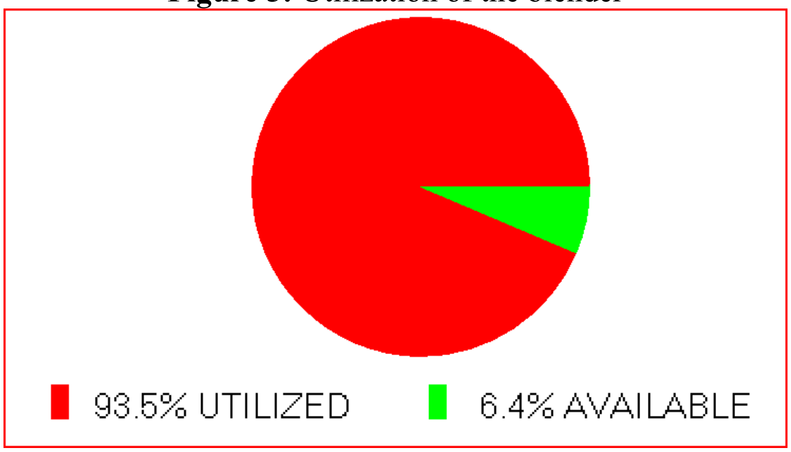


Figure 6: Utilization of the crusher

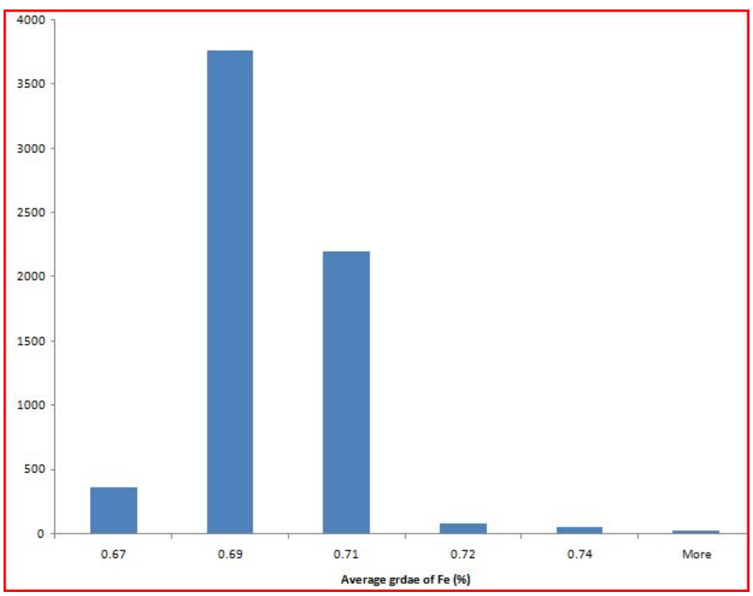


Figure 7: Histogram for average grade of blending after 100 runs

5.1.2 The impact of breakdown of the crusher on the system

a) Average waiting time for the crusher:

To consider the impact of the breakdown of the crusher on the system, the Alter node is added to the network. The network is in appendix D and the summary report of Awesim is in appendix E.

Table 5: AweSim report (crusher breakdown has been simulated).

File Number	Label or Input Location	Average Wait Time
Queue	High-grade stockpile	86.310
Queue	Low-grade stockpile	0.435
Resource	Blender	70.337
Resource	Crusher	20.584

Table 5 shows that when the crusher breakdown is simulated in the system, the average waiting time for the crusher at the await node for this resource will increase, which makes sense since the breaking down of the crusher and the time required for fixing it causes the pile up at the await node. According to table 4, the average waiting time for the crusher without simulating the breakdown is 17.00 units, while when the crusher breakdown occurs, the average waiting time for this resource increases to 20.584 units.

b) Utilization of the crusher

From the figure 7 and comparing to figure 5 it can be seen that when the crusher breakdown is simulated in the system, the utilization is increased to 97.1%.

5.2. The impact of breakdown of the blender on the system

a) Average waiting time for the blender

Table 6: AweSim summary report (breakdown of the blender).

FILE NUMBER	LABEL OR INPUT LOCATION	AVERAGE WAIT TIME
Queue	High-grade stockpile	86.310
Queue	Low-grade stockpile	0.435
Resource	Blender	71.036
Resource	Crusher	16.247

Table 6 and comparing it to table 4 shows that by simulating the blender breakdown in the system, the average waiting time for the blender resource will increase to 71.036, while the waiting time for the crusher decreases. The network for this simulation is in appendix F, and the AweSim summary report is in appendix G. The decrease in the average waiting time at the crusher await node is due to the increase in the blender await node, in other words, as the process of blending is before the crusher process, when the waiting time for the first one increases, the waiting time for the other resource will decrease.

b) Utilization of the blender

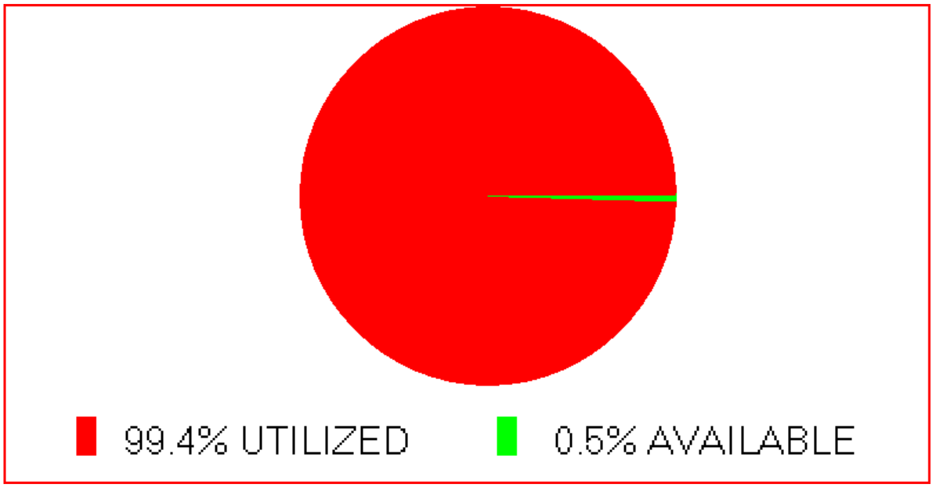


Figure 8: Utilization of the blender when blender breakdown occurs in the system

Figure 7 shows that when the breakdown of the blender occurs in the system, the utilization percent of the blender is increased to 99.4% which means that the idle time for the server is decreased.

5.3. The impact of duration on the average grade

In order to meet the desired range for the average grade the duration for sending blocks from low-grade and high-grade stockpiles should be considered. In other words, to satisfy the conditions for the desired average grade, the portion of blocks which are coming from two different stockpiles should be considered. In the simulation network for this project the duration for sending blocks from high-grade stockpile was set to 3 units of time, while the duration for sending blocks to the blender form low-grade stockpile was 2, and according to table 1 the average grade of the blending was 0.719%.

At this stage it can be assumed that the desired range of the grade is 0.6900 ± 0.01, therefore the portion of blocks from stockpile 2 should be more than the portion of blocks from stockpile 1. To see the impact the durations are changed to 8 and 1.5, the result is as fin table 7.

Table 7: AweSim summary report.

Label	Mean Value	Std.	Number of Observations	Minimum value	Maximum Value
Average grade	0.695	0.005	82	0.675	0.723
Ore Tonnage stockpile 1	15497770	8789186	37	681815	28818244
Ore Tonnage Stockpile2	25822516	13651720	69	633054	4903539
No. Blocks from STP1	18.159	10.70	82	0	37
No. Blocks from STP2	32.646	19.430	82	1	68

Table 8: AweSim summary report.

File Number	Label or Input Location	Average Wait Time
Queue	High-grade stockpile	126.716
Queue	Low-grade stockpile	0.174
Resource	Blender	27.953
Resource	Crusher	20.206

As it can be seen form table 7 and table 8 and comparing them to table 4, increasing the duration of sending blocks from high-grade stockpile, and decreasing the other duration decreases the average grade of blending and increases the average waiting time for blocks in the high-grade stockpile to 126.76 while the average waiting time for the blocks in the low-grade stockpile to be sent to the blender has been decreased to 0.174.

6. References

[1] Alan, A., Pritsker, B., and O'reilly, J., (1999), "Simulation with visual SLAM and AweSim", © System publishing and John Wiley, New York, Second ed, Pages 828.

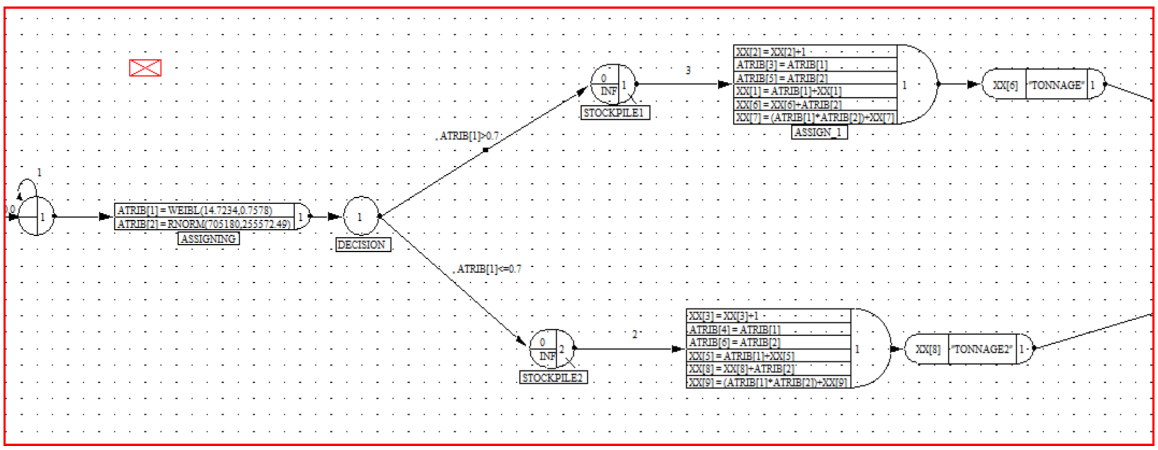
[2] Raczynski, S., (2009), "On validity of discrete event models", © Retrieved april 2009, 2009 from: http://www.raczynski.com/art/validity.doc

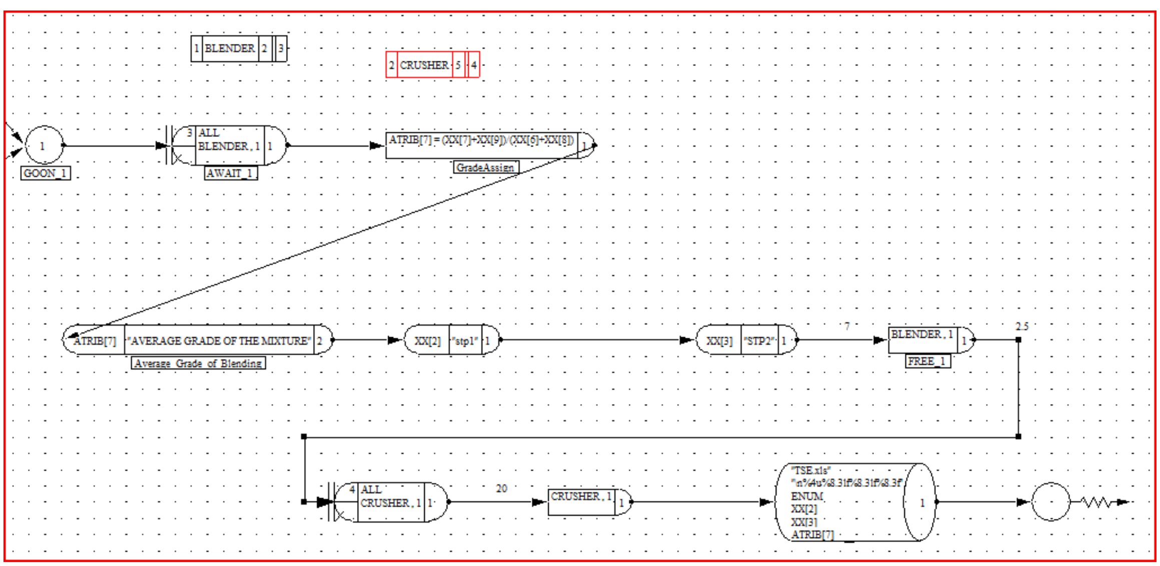
[3] Tako, A. and Robinson, S., (2009), "Comparing discrete event simulation and system dynamics", *Journal of the operational research society*, Vol. 60, pp. 296-312.

Appendix A

```
f=1;
for i=1:8456
    m(i,1)=Blocks(1,i).gradeMWT;
    if m(i,1)>0
        MWT(f,1)=m(i,1);
        f=f+1;
    end
end
hist(MWT)
grid on
```

Appendix B





Appendix C

```
** AweSim SUMMARY REPORT **
Thu Apr 09 11:45:01 2009

Simulation Project : modeling
Modeler : samira
Date :
Scenario : BASECASE

Run number 100 of 100
Current simulation time    : 300.000000
Statistics cleared at time : 0.000000
```

**** OBSERVED STATISTICS REPORT for scenario BASECASE ****

Label	Mean Value	Standard Deviation	Number of Observations	Minimum Value	Maximum Value
TONNAGE	38027424.546	20988046.747	100	681815.021	72801400.864
AVERAGE GRADE OF	0.719	0.008	86	0.709	0.767
stp1	49.849	29.088	86	1.000	99.000
STP2	32.930	19.769	86	0.000	68.000
TONNAGE2	25822516.855	13651720.785	69	633054.661	49035395.754

**** FILE STATISTICS REPORT for scenario BASECASE ****

File Number	Label or Input Location	Average Length	Standard Deviation	Maximum Length	Current Length	Average Wait Time
1	QUEUE STOCKPI	66.747	38.116	131	131	86.310
2	QUEUE STOCKPI	0.100	0.300	1	0	0.435
3	RES. BLENDER	39.623	23.723	83	83	70.337
4	RES. CRUSHER	4.703	3.095	10	10	17.000
0	Event Calendar	9.800	1.555	11	10	3.680

**** SERVICE ACTIVITY STATISTICS REPORT for scenario BASECASE ****

Activity Number	Label or Input Location	Server Capacity	Entity Count	Average Utilization	Standard Deviation
0	Line 11	1	0	1.000	0.000
0	Line 38	1	0	0.460	0.498

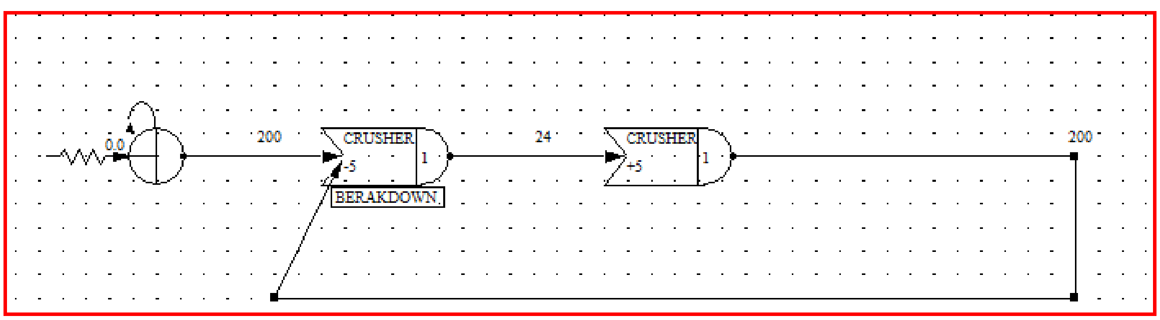
Activity Number	Current Utilization	Average Blockage	Maximum Idle Time or Servers	Maximum Busy Time or Servers
0	1	0.000	0.000	300.000
0	0	0.000	16.000	8.000

**** RESOURCE STATISTICS REPORT for scenario BASECASE ****

Resource Number	Resource Label	Average Util.	Standard Deviation	Current Util.	Maximum Util.
1	BLENDER	1.970	0.222	2	2
2	CRUSHER	4.678	1.128	5	5

Resource Number	Current Capacity	Average Available	Current Available	Minimum Available	Maximum Available
1	2	0.030	0	0	2
2	5	0.322	0	0	5

Appendix D



Appendix E

**** AweSim SUMMARY REPORT ****
Thu Apr 09 11:52:56 2009

```
Simulation Project : modeling
Modeler : samira
Date :
Scenario : BASECASE

Run number 100 of 100
Current simulation time    : 300.000000
Statistics cleared at time : 0.000000
```

**** OBSERVED STATISTICS REPORT for scenario BASECASE ****

Label	Mean Value	Standard Deviation	Number of Observations	Minimum Value	Maximum Value
TONNAGE	38027424.546	20988046.747	100	681815.021	72801400.864
AVERAGE GRADE OF	0.719	0.008	83	0.709	0.767
stp1	48.542	28.745	83	1.000	98.000
STP2	32.036	19.516	83	0.000	67.000
TONNAGE2	25822516.855	13651720.785	69	633054.661	49035395.754

**** FILE STATISTICS REPORT for scenario BASECASE ****

File Number	Label or Input Location	Average Length	Standard Deviation	Maximum Length	Current Length	Average Wait Time
1	QUEUE STOCKPI	66.747	38.116	131	131	86.310
2	QUEUE STOCKPI	0.100	0.300	1	0	0.435
3	RES. BLENDER	40.017	24.307	86	86	7.337
4	RES. CRUSHER	5.378	3.992	14	13	20.584
0	Event Calendar	11.512	1.742	14	11	4.355

**** SERVICE ACTIVITY STATISTICS REPORT for scenario BASECASE ****

Activity Number	Label or Input Location	Server Capacity	Entity Count	Average Utilization	Standard Deviation
0	Line 11	1	0	1.000	0.000
0	Line 38	1	0	0.460	0.498

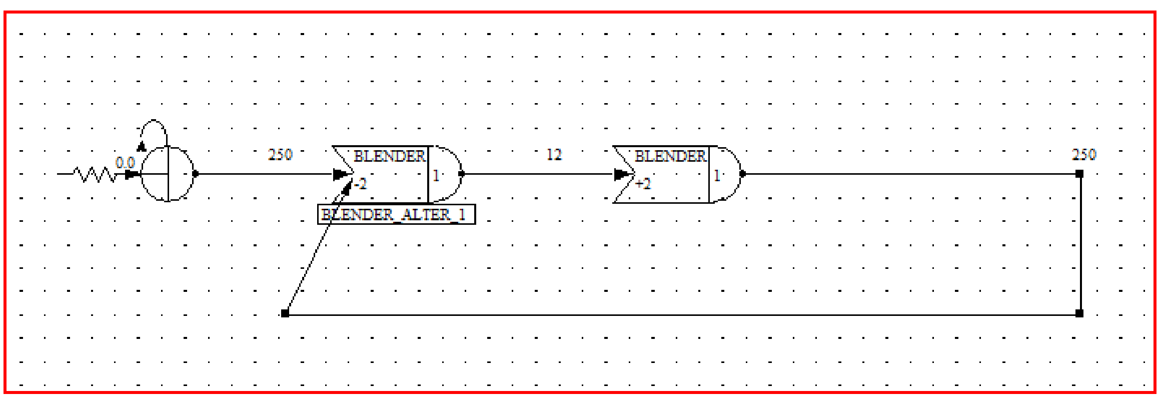
Activity Number	Current Utilization	Average Blockage	Maximum Idle Time or Servers	Maximum Busy Time or Servers
0	1	0.000	0.000	300.000
0	0	0.000	16.000	8.000

**** RESOURCE STATISTICS REPORT for scenario BASECASE ****

Resource Number	Resource Label	Average Util.	Standard Deviation	Current Util.	Maximum Util.
1	BLENDER	1.910	0.385	2	2
2	CRUSHER	4.467	1.365	5	5

Resource Number	Current Capacity	Average Available	Current Available	Minimum Available	Maximum Available
1	2	0.010	0	-2	2
2	5	0.133	0	-5	5

Appendix F



Appendix G

**** AweSim SUMMARY REPORT ****
Thu Apr 09 11:58:24 2009

Simulation Project : modeling
Modeler : samira
Date :
Scenario : BASECASE

Run number 1 of 100
Current simulation time : 300.000000
Statistics cleared at time : 0.000000

**** OBSERVED STATISTICS REPORT for scenario BASECASE ****

Label	Mean Value	Standard Deviation	Number of Observations	Minimum Value	Maximum Value
TONNAGE	38027424.546	20988046.747	100	681815.021	72801400.864
AVERAGE GRADE OF	0.719	0.008	83	0.709	0.767
stp1	48.542	28.745	83	1.000	98.000
STP2	32.036	19.516	83	0.000	67.000
TONNAGE2	25822516.855	13651720.785	69	633054.661	49035395.754

**** FILE STATISTICS REPORT for scenario BASECASE ****

File Number	Label or Input Location	Average Length	Standard Deviation	Maximum Length	Current Length	Average Wait Time
1	QUEUE STOCKPI	66.747	38.116	131	131	86.310
2	QUEUE STOCKPI	0.100	0.300	1	0	0.435
3	RES. BLENDER	40.017	24.307	86	86	71.036
4	RES. CRUSHER	4.387	2.705	9	8	16.247
0	Event Calendar	10.723	1.617	13	10	4.052

**** SERVICE ACTIVITY STATISTICS REPORT for scenario BASECASE ****

Activity Number	Label or Input Location	Server Capacity	Entity Count	Average Utilization	Standard Deviation
0	Line 11	1	0	1.000	0.000
0	Line 38	1	0	0.460	0.498

Activity Number	Current Utilization	Average Blockage	Maximum Idle Time or Servers	Maximum Busy Time or Servers
0	1	0.000	0.000	300.000
0	0	0.000	16.000	8.000

**** RESOURCE STATISTICS REPORT for scenario BASECASE ****

Resource Number	Resource Label	Average Util.	Standard Deviation	Current Util.	Maximum Util.
1	BLENDER	1.910	0.385	2	2
2	CRUSHER	4.678	1.128	5	5

Resource Number	Current Capacity	Average Available	Current Available	Minimum Available	Maximum Available
1	2	0.010	0	-2	2
2	5	0.322	0	0	5