DESIGN AND DEVELOPMENT OF A DISTRIBUTED EARTHMOVING SIMULATION

Duanshun Li[a], Cristian Petre[b], Cayce Kerr[c], Tim Joseph[d], Simaan AbouRizk[e], and Yasser Mohamed[f]

[a]University of Alberta
[b]University of Alberta
[c]University of Alberta
[d]University of Alberta
[e]University of Alberta
[f]University of Alberta

[a]duanshun@ualberta.ca, [b]petre@ualberta.ca, [c]cayce@ualberta.ca, [d]tim.joseph@ualberta.ca, [e]abourizk@ualberta.ca,
[f]yaly@ualberta.ca

ABSTRACT

Earthmoving operations are critical components of many mining and construction projects. These operations utilize heavy equipment to execute excavation, loading, hauling, dumping, grading, and compacting tasks in a repetitive and almost continuous manner. The efficiency, environmental impact and cost of these operations are affected by many interdependent factors. These include road layout, road material, road maintenance frequency, trucks and excavator characteristics, and weather conditions. This paper describes the design and development of a comprehensive simulation of earthmoving operations. The simulation presented herein is developed with a distributed approach using High Level Architecture (HLA) Standards to enable inclusion of different models and behaviors that affect these operations. The paper presents the components of the simulation and models implemented in each, with more focus on the transportation and hauling component. It also describes the distributed development process and showcases the implementation outcomes and sample use and verification of the simulation.

Keywords: Simulation, High Level Architecture, Earthmoving

1. INTRODUCTION

Despite the considerable complexity and risk involved in construction projects, computer based simulation provides a powerful tool to predict the potential outcome of certain construction plans including time, duration, cash flow, risk, etc. Earthwork operations have been a common application field for simulation since it was introduced to the construction industry in the early 1970s. After the innovation of a graphic based simulation software, CYCLONE (Halpin 1977), application of simulation in construction increased. Earlier research on earthwork simulation mainly focused on estimation (Willenbrock 1972, Clemmens and Willenbrock 1978). The focus then transformed to productivity prediction and fleet optimization (Karshenas 1989, Smith and et al. 1995, Hegazy and Kassab 2003, Marzouk and Moselhi 2003, Marzouk and Moselhi 2004, Moselhi 2009). Hajjar and AbouRizk (1996) proposed a special purpose earthmoving simulation template based on Simphony (AbouRizk and Mohamed 2000). The special purpose template presents the user with graphical, high-level details through object oriented simulation, so the user does not require simulation knowledge, and thus, it enhances the speed of model construction. However, modifications or adjustment to the template could take considerable time for specific projects due to the uniqueness of each construction project.

One drawback shared by all of these models is that most of them are established, at the lowest level, on 4 major components: loading, hauling, dumping and returning, which ignore the existence of detailed models of individual components. However, considerable research has been done on the modeling of those tasks, especially loading and hauling/returning. A number of models (Filla 2005, Coetzee et al. 2007, Nezami et al. 2007, Wang and Yang 2007, Bošnjak et al. 2008, Coetzee and Els 2009, Schmidt et al. 2010) of the loading process have been proposed for excavator simulation based on bucket-soil interaction. While others (Tam et al. 2002, Maciejewski et al. 2003) try to model the productivity of the excavators. The vehicle operation cost (VOC) has been thoroughly investigated. Tan et al. (2012) provide a thorough review of existing VOC models and divide the evaluation of those models into 4 phases: pre-1970s broad level correlation studies, 1970s-1980s regression models, post-1980s mechanistic models and current research on updated vehicle technology and changing vehicle fleet. In this research, 4 major components are identified: fuel consumption, tire use, repair/maintenance and lubricating oil. As the dominating cost, fuel consumption highly depends on the speed and mass of the vehicle, the engine efficiency, the layout of the road, such as gradient and curvature, and the roughness of the road, and the weather conditions, such as air speed. More information on widely used fuel consumption models NIMPAC and HDM-types can be found in Thoresen and Roper (1996). The drawback of those models is that the models neglect the evolution of roughness during the operation of the roads. As stated in Thompson et al. (2003), the roughness is not only related to the wearing course of the road, but also the usage and maintenance frequency. Based on experience in South Africa, Thompson and Visser (2003) also proposed a model aiming to derive the fuel consumption based on rolling resistance estimated along the operation time.

Despite of the availability of these models, few of them are applied in earthmoving simulation tools. One of the major barriers is the diversity of the models. Users in different regions have different soil or weather conditions; they may also have different data sources, which make inputs quite different. Considering roughness of the road, for example, it can be measured as international roughness index (IRI) or NRM count per kilometer; however, in a large amount of research, it is modeled as rolling resistance. To make full use of these models, a simulation platform that allows users to switch between models and easily make additions or adjustments is necessary. As an international standard for distributed simulation, High Level Architecture (HLA) is an optimal choice. The standard allows the user to switch between different models regardless of the location, programming language or inputs. HLA enables the integration of numerous existing models based on different operation systems or programming languages for loading, hauling and other components, and thus, form a complete comprehensive earthmoving simulation model.

Based on an HLA earthwork simulation platform, this paper describes the implementation of a Mover federate aiming to provide the location and the rolling resistance for each truck during travel between loading and dumping sites to facilitate visualization and fuel consumption, CO_2 emission, and cost estimation done by other federates. The main contribution of this work is that: 1) it derives the location and the rolling resistance according to the input information, such as breakdown state and loading state of the truck provided from other federates based on the model introduced by Thompson and Visser (2003); 2) it accounts for several scenarios of the truck limiting speed strategies and maintenance strategies that affect operation performance. For example, one performance measure that is usually overlooked in earthmoving simulations is the tonnage moved per hour and its effect on tire deterioration or ton kilometers per hour (TKPH). There are limited ratings with regards to this measure that a tire can handle without overheating and causing premature deterioration for each type of tire (Mining 2014). The federate models these effects and allows creation of scenarios where speed is governed by either user input, sampling from fitted distribution or TKPH limits.

2. SIMULATION STRUCTURE

The earthmoving operation is modeled as an HLA federation that has five main federates that interact during runtime to simulate different behaviors. These federates are Controller federate, Loader federate, Mover federate, Weather federate and Breakdown federate. Figure 1 illustrates the federation and the objectives of each federate. The federates are time regulated and the time step used is 1 minute.

Each federate in the federation is implemented by a separate development team/individual. The federation uses an HLA run time infrastructure developed at the University of Alberta (AbouRizk et al. 2009) with API available for .Net languages, Java, and Python. All federates were developed in Python except for the Controller federate, which was developed using C#.

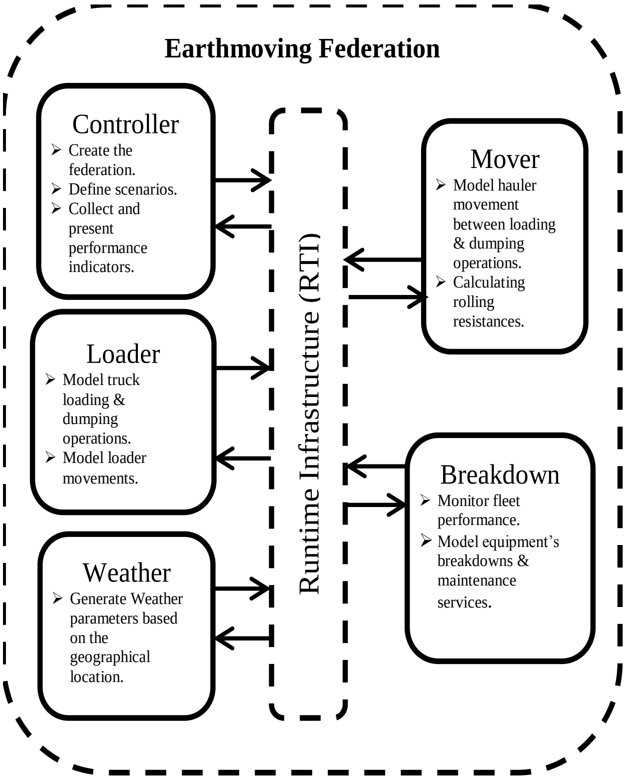

Figure 1: Federation Structure

The Controller federate is responsible for starting the HLA service and creating the earthmoving federation. The federate also controls the starting point of the simulation and initializes some of the essential attributes for the project object classes, such as the fleet size and combination, and the road sections for hauling and returning operations. At the same time, the federate receives results from other federates and presents them in a graphical and statistical fashion.

A Loader federate is established to simulate the excavating operation. The main job of this federate is to model the excavating process, which includes updating the loading status and the gross weight of the vehicle; meanwhile, it is also responsible for the dumping process related to the loading status.

To make the model more complete and realistic, modeling other critical factors, such as breakdown events and the weather condition, is essential. In the federation, the weather condition will be forecasted hourly based on a realistic historical data set by the Weather federate, while the Breakdown federate is responsible for the breakdown events for each piece of equipment.

Most of the time and cost spent on vehicles is during traveling. As the most critical part of the simulation, the Mover federate changes the moving state of the trucks based on the road layout, loading states and the breakdown states received from other federates. The position and rolling resistance will be derived based on the course wear of the road and the speed sampled from fitted distributions. Meanwhile, the results will be published to other federates for further calculation. Focusing on the comprehensive modeling of the traveling process, the mover federate will be detailed in the following sections.

Each federate uses its own model to update object attributes under its jurisdiction separately, while the communication between federates is established through the shared federation object model (FOM). The FOM describes the types of objects and attributes that are exchanged and the parameters that should be transferred for interactions during the simulation. The major objects the earthmoving operation is dealing with are equipment and the road network; thus, they are recognized as the major part of the object classes in our implementation as illustrated in Table 1. The publish/subscribe (P/S) properties for each attribute is also given in the table, where C, W, M, L and B indicate Controller, Weather, Mover, Loader and Breakdown separately. The abbreviation will be applied hereafter for convenience in this paper. In our current implementation, weather updating is treated as the sole interaction class, and the structure of it is presented in Table 2.

3. MOVER FEDERATE

The mover federate aims to provide the location and the rolling resistance for each truck during travel between loading and dumping sites to facilitate visualization and fuel consumption, CO2 emission, and cost estimation simulated by other federates. In addition, statistics regarding the cycle time for the trucks are collected and published to the Controller federate for visualization and performance analyzing. The federate requires certain inputs from other federates as well. The key inputs include road layout and road wearing course material, truck model, breakdown state and loading state. Generally, the federate execution can be divided into three stages: Initialization, Instance Creation and Updating. The order of initialization and instance creation is given in Figure 2.

Table 1: Object Class Structure Table

Attribute		**Type**	**C**	**W**	**M**	**L**	**B**
Equipment							
Location		Point3D	S		P S	S	S
Model		String	P S		S	S	S
BreakdownState		Enum	S		S	S	P
Capacity		Float	P		S	S	
EquipmentAvailable		Float	S				P
EquipmentUnavailable		Float	S				P
	Truck						
	LocationType	Enum			P	S	
	LoadState	Enum			S	P	S
	MovingState	Enum	S		P	S	
	PayLoad	Float	S		S	P	
	RollingResistance	Float	S		P		
	RoadSegmentID	Integer	S		P		
	WeightEmpty	Float	P		S		S
	CycleTime	Float	S		P		
	HaulTime	Float	S		P		
	ReturnTime	Float	S		P		
	TKPHLimit	Float	P		S		S
	TireTKPHAchieved	Int64	S				P
	Excavator						
	ProductionRate	Float				P	
	UtilityRate	Float	S			P	
RoadSegment							
Id		Integer	P		S		
Node1		Point3D	P		S		
Node2		Point3D	P		S		
Material		String	P		S		
DailyKTonnage		Float	S		P		
MaintenanceDate		Date					
Road							
Sections		IntegerArray	P		S		
Project							
Location		String	P	S			
StartDay		Date	P	S	S		

Table 2: Interaction Class Structure Table

Parameter	**Type**	**C**	**W**	**M**	**L**	**B**
CurrentWeather		S	P	S	S	S
Temperature	Float	S	P	S	S	S
WindSpeed	Float	S	P	S	S	S
Visibility	Float	S	P	S	S	S
SnowFall	Float	S	P	S	S	S
SnowDepth	Float	S	P	S	S	S
Precipitation	Float	S	P	S	S	S

Figure 2: Initialization and Instance Creation

To evaluate the performance of the fleet under different speed control strategies, switches and other parameters are provided to allow the user to simulate different scenarios. The lowest speed and highest speed are provided as the hard limit of the maximum and minimum speed of each moving truck. The maintenance interval of the road, which will influence the rolling resistance and furthermore the cost, is also provided and can be modified using one of the switches. The interval can be specified either as a constant or as a fitted distribution. Considering that exceeding the TKPH limit of the tires may cause them to breakdown prematurely and finally influence the production of the system, two Boolean parameters are specified as switches to control the speed of the trucks. The first switch allows the user to choose between sampling the speed of the trucks from past data or setting it as a constant value derived from the TKPH limit and the gross vehicle weight. The other one is related to the speed sampling scenario and allows the user to specify whether to apply a speed limit derived from the TKPH limit as an additional maximum speed limit for each truck, or not. However, whether the TKPH limit is used to derive the speed of the truck or as an additional limit, the truck will never achieve this value as long as there are stops during any given hour due to loading or dumping. The switches and parameters are organized in an initialization file which is easy to modify, while the configurations will be read from the file and applied to the simulation at the beginning of the federate execution.

In the Instance Creation stage, the federate listens to the instance creation messages and creates local records of the instances. The road layout is the combination of a group of road segments and one or several lists of ordered road segment IDs called road sections, indicating different hauling roads. For each truck there must be a hauling road assigned to it, which means the truck should not be created before the creation of the road section; the order of initialization and instance creation is given in Figure 2. Apart from the coordinates, the material of the road should also be provided from the Controller. This is needed in order to derive the rolling resistance. To simplify the interface of the Controller, a local record of properties of the wearing course material is kept by the Mover federate.

In the most important stage of the federation execution, the updating stage, the federate will listen to the messages from other federates related to the breakdown state and loading state in order to derive the location and rolling resistance for each truck before sending the updated values to the RTI. Certain statistics, such as the cycle time, hauling trip duration, return trip duration for each truck and daily tonnage hauled over each road segment, are also collected for visualization and evaluation.

The updating procedure is illustrated in Figure 3. To make the illustration clearer, vertical lines and horizontal lines are used to indicate the time line and the order of the procedures separately. Each updating cycle starts from the end of time i which is achieved when all federates ask for time advance request to time i+1. Once it is achieved, the Mover federate starts to receive messages from the RTI. The Mover will calculate the location and rolling resistance of each truck according to that information and the local record and store it in a local updates list. Once the time reaches i+1 and all of the messages have been processed, it will publish all of the updates to other federates and ask for time advance request to time i+2.

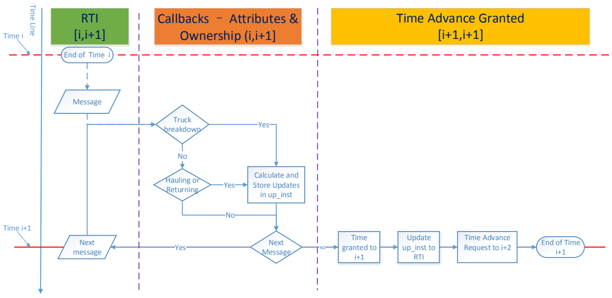

Figure 3: Flow Chart of the Mover Federate Updating Process

3.1. Location Updating

The location of the vehicles is updated every minute and published to other federates. The method used to update the location is quite straight forward. The travelling distance is calculated based on kinematics theory using the derived speed. The speed of the vehicles is controlled by the switches and distributions specified in the initialization file, as stated previously. Combined with specified road layout as constraints, the 3D location can be easily derived.

3.2. Rolling Resistance Updating

Resistance is one of the most significant factors that will influence the productivity of the road network. The total resistance (TR) is the combination of the grade resistance (GR) and the rolling resistance (RR). It is usually specified as effective grade (EG) in the form of percentage of gross vehicle mass (GVM). It can be written as:

$$\begin{aligned} TR &= RR + GR \\ EG(\%) &= RR(\%) + GR(\%) \end{aligned} \tag{1}$$

The GR can be easily derived from the geometry layout of the road segments and will not be the major concern in this paper. On the contrary, as a factor highly related to the roughness of the road, rolling resistance is usually recognized as an indicator of the roughness, similar to the international roughness index (IRI) or NRM counts per kilometer. Increased rolling resistance reduces fleet productivity, and leads to increases in operating costs. It is mainly composed of five major components:

- internal power train friction,
- tire flexing under load,
- tire penetration,
- road deflection, and
- air resistance.

Though a rule-of-thumb formula for RR estimation exists and is widely used in the heavy equipment manufacturing industry, it is supposed to be underestimated on an unpaved surface (Karaftath 1988, Tannant and Regensburg 2001). To address the problem, a number of models based on tire-soil interaction and surface roughness are proposed. Karaftah (1988) argues that the general model underestimates the rolling resistance of non-driven tires and proposes a method to calculate the rolling resistance and the speed attainable during hauling by taking the torque into consideration. However, this model includes too much trivial detail of the tire soil interaction, which makes it too sensitive for simulation. The other disadvantage of this model is that it does not quantify the properties of the wearing course of the road and does not take them into consideration. Another mechanistic-empirical model proposed by Paterson (1987) quantifies the deterioration of the road as roughness which is the combination of structural deformation, surface defects and age and environmental factors. Harvey (2012) revised this formula based on this method which can be illustrated as:

$$R(t) = [R_0 + 134SNCK^{-5}NE(t)]e^{mt} \tag{2}$$

where m is the environmental coefficient, t is the pavement age, SNCK is the modified structural number adjusted for the effect of cracking, and ΔNE is millions of equivalent standard axle (ESA) loads per lane during the period, and *NE(t)* is cumulative ESAs until time ten millions per lane. The problem with this model is that although the roughness is quantified, the transformation from roughness to rolling resistance is still missing. The problem is addressed by Thompson and Visser (2003), in which a formula for rolling resistance estimation, while taking deterioration of the road and the material of the wearing course into consideration, is provided. In this method, the wearing course is quantified by California Bearing Ration (CBR), Grading Coefficient (GC), Shrinkage Product (SP), Plastic Index (PI) and other characteristics. The roughness is derived from the wearing course material, daily tonnage hauled on the road and the number of days since the last road maintenance.

Based on the survey and evaluation of existing models, the model provided in Thompson and Visser (2003) is applied in the Mover federate for rolling resistance estimation. In this model, roughness defect score (RDS) is defined to quantify the effect of the road wearing course material and its interaction with the tire in the context of the traffic speed and total traffic volume on a given haul road.

This model considers the rolling resistance as a function of the vehicle speed and aforementioned RDS, which incorporates the wearing course parameters, as well as the number of days passed since the road was last maintained, together with the total volume of traffic within that period. Assume the daily tonnage hauled on the road segment is KT, the RDS of the specified road segment is provided in Equation 3 given the parameters of the road wearing course. The model requires information with regards to the 100% Mod. California Bearing Ratio (CBR), the Plasticity Index (PI), SP and GC. The last two parameters required can be defined either independently or derived from the passing rate through gradation test (Thompson 2003).

$$RDS = RDSMIN + \left[\frac{RDSMAX - RDSMIN}{1 + e^{RDSI}}\right] \tag{3}$$

where

$$RDSMIN = 31.1919 - 0.05354SP - 0.0152CBR$$
$$RDSMAX = 7.6415 + 0.4214KT + 0.3133GC + 0.4952RDSMIN$$
$$RDSI = 1.768 + 0.001D\,(2.69KT - 72.75PI - 2.59CBR - 9.35GC + 1.67SP$$

With speed V, the rolling resistance of the vehicle on this road segment is given in Equation 4.

$$RR = RRMIN + RDS \cdot e^{RRI} \tag{4}$$

where

$$RRMIN = e^{-1.8166 + 0.0028V}$$
$$RRI = 6.068 - 0.00385RDS + 0.0061V$$

3.3. Tire Capacity

The travel conditions such as speed, payload, road conditions and cycle time influence the performance of a truck and its breakdown likelihood directly (Zhou 2006). One measure of the work a tire has to perform in a hauling operation is the ton kilometer per hour (TKPH). There are limiting ratings with regards to this measure that a tire can handle without overheating and causing premature deterioration for each type of tire (Ming 2014).

Although dynamic conditions govern the load distribution between tires depending also on payload and ground conditions (Joseph 2003), an empirical method of deriving the tire load when exact data is not available is to divide the truck load by the number of tires used. The Mover federate implements this concept and allows its use either as a limiting factor or as a means of deriving the hauling speeds. For either case, although it is recognized that a truck will not spend its entire time travelling, and will be allowed to cool down during loading and dumping, it was decided that for a safe implementation in the case when the TKPH limits are used to derive hauling speeds, these cooling times will be ignored as a means of implementing a factor of safety in the operation. However, as will be described in more detail in Section 4, the user will have the ability to choose if they wish to consider this metric in the simulation or not.

For a truck with 6 tires, the speed limit of the truck will be:

$$V_{max} = \frac{6 \cdot TKPH}{GVW} \tag{5}$$

where GVW is the total mass of both the vehicle and the payload.

4. SUPPORTED SIMULATION SCENARIOS

The federate user can change types and parameters of the distributions used to sample the travel speed from, as well as select speed limits that are to be applied. The tire speed limitations can also be derived from their TKPH limits received from the controller federate; this option can be switched on and off at the user's choice. The default data used for the speed distributions allows for four different types of trucks. In addition, there is another set accepted in case the truck types received from the controller do not coincide with those in the data input file; for the case study this is an average distribution of the available data sets.

Soil parameters can be modified according to the specifications of a particular application. Currently, the federate supports two sets of material types, each with up to three materials. The first type is specified in terms of the sieve percentages, whereas the second accepts readily defined parameters. Once the parameters of these materials are specified, the user can select the material applicable to each road segment from the Controller federate by specifying the material name.

Specifying soil parameters and speed distribution parameters can be achieved by changing the values in the data initialization .csv files. Once all the required information is introduced in the speed and soil .csv files, the user can specify the speed limits and choose which scenario or combination to run from the Init.csv initialization file. This file allows a user to specify the minimum and maximum speed limits for the operation, as well as a maintenance interval for the roads. By setting the Maximum Tire parameter to 0, the model will use the distributions to sample a speed for a truck each time travel starts or the segment changes. Setting this parameter to 1 will bypass the distribution sampling and use a model that computes the speed based on the TKPH limit. This scenario is one where the trucks tires shall never fail, since their haul speeds are derived from their TKPH limit (Michelin 2004), and they are also allowed to cool during loading and dumping.

The last scenario option available is the "TKPHASLIMIT", which expects either a 0 or a 1 for the value as a Boolean switch. Setting it to 1 applies a speed limit based on the TKPH speed derivation in the case where the federate samples truck haul speeds from the given distributions. This check is performed in addition to checking the Highest Speed value, and in case the sampled speed is larger than either of them, the smallest value will be used instead.

5. EXPERIMENT AND RESULTS

5.1. Data Resources

The haul road in the experiments is provided as three road segments with a total length of 2.18 km. Based on a realistic data set collected at an Alberta mining operations site, two distributions are derived to fit the hauling trip speed and the returning trip speed respectively for each truck model. According to the result, the best fits of the hauling trip speed are normal distributions and the best fits of returning trip are triangular distributions based on an evaluation which combines the result from criteria provided by @Risk. The hauling speed distribution parameters for each type of trucks are given in Table 3, where "other" will be applied if the model specified by the user cannot be found in the local record. This table also includes information about the TKPH limit for the tires used and the capacity of the trucks.

Table 3: Hauling trip distributions and TKPH tire limits

Truck	Hauling (Normal) (km/hr)		TKPH Limit for Each Tire	Capacity (tons)
	Mean	Standard Deviation		
CAT 785C	20.0146	5.0947	540	140
CAT 793B	20.5110	5.1015	781	237
CAT 793C	20.6625	5.1017	848	240
CAT 793D	21.3953	5.0717	812	240
Other	20.5613	5.1152	-	

Three typical road wearing course materials are used in the experiment: Stone, Slag and Dolerite. The characteristics of the materials are described as the following table.

Table 4: Material characteristics

Material	CBR	LS	PI	P425	P2	P475	P265
Stone	69	0.5	0	27	32	37	60
Slag	140	0.5	0	17	37	61	100
Dolerite	26	3.5	7	34	39	45	70

5.2. Scenarios and Results

Several experiments are constructed based on the initialization files for scenario configuration. Both the lowest and highest speed limits can be specified in the configuration file. Additionally, another speed limit derived from the TKPH limit can be applied through a switch. If the switch is on, the federate will check if sampled speed satisfies all of the limits or not, if not satisfied the THKPH derived speed limit will be adopted.

The federation execution has been run for 10 days in order to produce an initial set of results. In this experiment, four types of trucks listed in Table 3 are used; these are: 785C, 793B, 793C and 793D. The number of the trucks used of each type is: 2, 1, 3 and 3 respectively.

Totally three scenarios are simulated:

A. Sampled speed according to fitted distribution without speed limits derived from TKPH limits;
B. Sampled speed according to fitted distribution with speed limits derived from TKPH limits;
C. Fixed speed derived from TKPH limits.

In Scenario A - Sampled Speed without TKPH Limits, the speed is sampled from the distribution fitted, and the speed limit derived from the TKPH limit will not be applied. The lowest and highest values allowed for speed are 1 and 50 km/hr respectively. Because the speed limit corresponding to TKPH limit is not adopted in this scenario, there will be no truck breakdowns due to exceeding of the TKPH limit.

Due to the TKPH derived speed limit being applied in scenario B, there will be no truck breakdown due to exceeding of the TKPH limits in this scenario. What should be noted is that, in scenario C, the potential of the tire capacity is maximized compared to the other two scenarios, by setting the speed according to this limiting factor.

The statistics of the production and the cycle times for each scenario are presented in Table 5, where KT is the average daily tonnage hauled on the road, a measure of the production rate.

Table 5: General statistics of the scenarios for the cycles without breakdowns

Scenario	**KT**	**Average Cycle Time (min)**				
		785C	**793B**	**793C**	**793D**	**ALL**
A	369	19.5	19.9	20.1	19.8	19.8
B	330	21.4	22.8	22.5	22.5	22.3
C	398	17.4	19.1	18.2	19.1	18.5

In order to compare the percentages of breakdown cycles between the scenarios, the results have been aggregated for all trucks with the breakdown cycles separated, as summarized in Table 6 below.

Table 6: Cycle Time Summary for all Scenarios

Cycle Time Summary			
Scenario	***A***	***B***	***C***
	All Cycles		
Count	4635	4047	5059
Average (min)	27.9	32.0	25.6
Minimum (min)	13.0	16.0	16.0
	Breakdown Cycles		
Count	353	359	343
Average (min)	125.9	131.4	124.1
Percentage	7.62%	8.87%	6.78%

5.2.1. Scenarios Interpretation

From these results, we can conclude that the maximum production rate is achieved while the potential of the tire is fully utilized (Scenario C). This can be said by looking at the daily average tonnage hauled on the road with all the trucks (KT values in Table 5) or at the total number of cycles achieved in 10 days of simulated operations for each scenario (Table 6).

Little variation exists between types of trucks and the cycle time when distribution sampling alone (Scenario A) is employed - this is because the input distributions for the different types of trucks are very similar (Table 3). However, one should also consider that these trucks have different capacities and therefore their loading and dumping time would vary in a fashion not analyzed within this study. When considering capacity, which is very similar between trucks 793 A, B and C, we can observe that the average cycle time (presented in Table 5) varies in a consistent fashion to the mean hauling speed (presented in Table 3). This means, for example, that the truck with the highest mean speed (CAT 793D) is also the one experiencing the lowest average cycle times in the case of Scenario A, although by a slim margin.

It is expected that the trucks would break down more frequently when the TKPH-derived speed limit is not enforced (Scenario A) due to the addition of the tire cool down periods enforced by the Breakdown Federate. However, the results do not show a decrease in the percentage of breakdown cycles between Scenario A and B. This can most likely be attributed to the random nature of the mechanical breakdowns simulated outside the scope this federate, which lead to the inability to analyze solely the TKPH breakdowns.

By analyzing the cycle times between the two cases where speed was sampled (A and B), it can be observed that applying a speed limit (in scenario B) has a negative impact on the average cycle time; this is valid whether all cycles are included (top part of Table 6) or only the ones without breakdowns (Table 5). When the speeds are derived from the TKPH limits (Scenario C), the average cycle times will be reduced back to similar values of Scenario A since the simulation will not be using lower speeds than those derived. Moreover, the return times decrease significantly due to the high allowable speed limit for the empty trucks; this behavior is induced by the reduced value for the denominator in Equation 4. This leads to the most efficient operation, where the productivity (in terms of either KT or cycle times) is maximized, while experiencing the minimum percentage of breakdowns.

6. CONCLUSION

A distributed simulation platform based on HLA standard is established to model earthmoving operation. The platform enables the possibility of utilization of existing comprehensive models of the loading and hauling process. As an example of one comprehensive model, the Mover federate is able to successfully update the location of trucks during the execution of an HLA compliant distributed simulation of an earth moving operation, while responding and reacting to updates from the Loader and Breakdown federates. The primary experimentation scenarios supported are related to the way the speed of the trucks is derived, which can be either from past data or can be derived in order to comply with the TKPH limit of the tires. The rolling resistance calculations are performed using a realistic and comprehensive model that accounts for the road wearing course deterioration. The experiment shows that the federate is able to reflect the influence of the fleet composition, the truck type, TKPH limit and the speed limits. Thus, the federate can be used to evaluate the effectiveness of the tire, the speed limit settings and fleet composition while planning the project.

REFERENCES

AbouRizk, S., and Mohamed, Y., 2000. Simphony: an integrated environment for construction simulation. *In Proceedings of the 32nd conference on Winter simulation*, pp.907-1914.

AbouRizk, S. M., Hague, S., & Moghani, E., 2009. Developing a bidding game using high level architecture. *In Proc., 2009 ASCE Int. Workshop on Computing in Civil Engineeringpp*, pp. 513-522). Reston, VA: ASCE.

Bošnjak, S., Zrnić, N., Petković, Z., 2006. Bucket wheel excavators and trenchers - Computer added calculation of loads caused by resistance to excavation. u: Kuzmanović S. (ur.) *Machine Design*, University of Novi Sad, pp.121-128

Clemmens, J. P., and Willenbrock, J. H., 1978. The SCRAPESIM computer simulation. *Journal of the Construction Division*, 104(4), 419-435.

Coetzee, C. J., Basson, A. H., and Vermeer, P. A., 2007. Discrete and continuum modelling of excavator bucket filling. *Journal of terramechanics*, 44(2), 177-186.

Coetzee, C. J., and Els, D. N. J., 2009. The numerical modelling of excavator bucket filling using DEM. *Journal of Terramechanics*, 46(5), 217-227.

Filla, R., 2005. An event-driven operator model for dynamic simulation of construction machinery. arXiv preprint cs/0506033.

Hajjar, D., AbouRizk, S., 1996. Building a Special Purpose Simulation Tool for Earth Moving Operations. *Proceedings of the 1996 Winter Simulation Conference*, pp.1313-1320.

Halpin, D. W., 1977. CYCLONE: Method for Modeling of Job Site Processes, *Journal of the Construction Division*, ASCE, 103(3), 489-499.

Harvey M.O., 2012. Optimising road maintenance. *International transport forum discussion papers*. OECD publishing.

Hegazy, T. and Kassab, M., 2003. Resource Optimization Using Combined Simulation and Genetic Algorithms. *Journal of construction engineering and management*, 129(6), 698–705.

Joseph, T. G., 2003. Large mobile mining equipment operating on soft ground. *In Proceedings of the 18th International Mining Congress and Exhibition of Turkey–IMCET*, pp. 10-13.

Karaftath, L. L., 1988. Rolling resistance of off-road vehicles. *Journal of construction engineering and management*, 114(3), 458-471.

Karshenas, S. (1989). Truck Capacity Selection for Earthmoving. *Journal of construction engineering and management*, 115(2), 212–227.

Maciejewski, J., Jarzębowski, A., & Trampczyński, W., 2003. Study on the efficiency of the digging process using the model of excavator bucket. *Journal of terramechanics*, 40(4), 221-233.

Marzouk, M. and Moselhi, O., 2003. Object-oriented Simulation Model for Earthmoving Operations. *Journal of construction engineering and management*, 129(2), 173–181.

Marzouk, M. and Moselhi, O., 2004. Multiobjective Optimization of Earthmoving Operations. *Journal of construction engineering and management*, 130(1), 105–113.

Michelin., 2004. *EM & Industrial Data Book.* Michelin.

Mining, P., 2014. *TKPH*, Precision Mining Software. Available from: http://support.precisionmining.com/spry-help/building-a-spry-model/scenarios/trucks/tkph/ [accessed 14 Apr 2014]

Moselhi, O. and Alshibani, A., 2009. Optimization of Earthmoving Operations in Heavy Civil Engineering Projects. *Journal of construction engineering and management*, 135(10), 948–954.

Nezami, E. G., Hashash, Y., Zhao, D., and Ghaboussi, J., 2007. Simulation of front end loader bucket–soil interaction using discrete element method. *International journal for numerical and analytical methods in geomechanics*, 31(9), 1147-1162.

Paterson WD., 1987. The highway design and maintenance standard series – road deterioration and maintenance effects. *Baltimore*, MD: The Johns Hopkins University Press.

Schmidt, D., Proetzsch, M., and Berns, K., 2010. Simulation and control of an autonomous bucket excavator for landscaping tasks. *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5108-5113.

Smith, S., Osborne, J., and Forde, M., 1995. Analysis of Earth-Moving Systems Using Discrete-Event Simulation. *Journal of construction engineering and management*, 121(4), 388–396.

Tam, C. M., Tong, T. K., and Tse, S. L., 2002. Artificial neural networks model for predicting excavator productivity. *Engineering Construction and Architectural Management*, 9(5‐6), 446-452.

Tan, F., Thoresen, T., & Evans, C. (2012, September). Review of Vehicle Operating Costs and Road Roughness: Past, Current and Future. *In 25th ARRB Conference*, pp.1-24.

Tannant, D. D., and Regensburg, B., 2001. Guidelines for mine haul road design. School of Mining and

Petroleum Engineering, *Department of Civil and Environmental Engineering*, University of Alberta.

Thompson, R., & Visser, A. (2003). Mine haul road maintenance management systems. *Journal of the South African Institute of Mining and Metallurgy*, 103(5), 303-312.

Thoresen, T. and Roper, R., 1996, Review and enhancement of vehicle operating cost models: assessment of non-urban evaluation models, *ARR 279*, ARRB Transport Research Ltd, Vermont South, Vic.

Wang, J., and Yang, H., 2007. Model tests and numerical simulation on bearing capacity of bucket foundations in soft clay under vertical static and cyclic loads. *International Offshore and Polar Engineering Conference*, pp.1547-1552.

Willenbrock, J. H., 1972. Estimating costs of earthwork via simulation. *Journal of the Construction Division*, 98(1), 49-60.

Zhou J., Hall R.A., and Fowler G., and Huntingford K., 2006. Applications of Engineering Analysis to Improve Tire Management. *Procedings of CIM*.