



A Multiagent System for Truck Dispatching in Open-pit Mines

Gabriel Icarte^{1,2(✉)}, Paulina Berrios², Raúl Castillo², and Otthein Herzog^{3,4,5}

¹ International Graduate School for Dynamics in Logistics (IGS),
University of Bremen, Bremen, Germany
`gicartea@uni-bremen.de`

² Faculty of Engineering and Architecture, Arturo Prat University, Iquique, Chile
`{gicarte,pberrios,raucasti}@unap.cl`

³ Tongji University, Shanghai, People's Republic of China
⁴ Jacobs University, Bremen, Germany

⁵ Center for Computing and Communication Technologies (TZI),
University of Bremen, Bremen, Germany
`herzog@tzi.de`

Abstract. Material handling is an important logistic process for open-pit mines. In this process, shovels extract materials and load trucks that transport these materials to different destinations. To support this process, different centralized dispatching solutions have been implemented based on mathematical programming, heuristic processes or simulation modelling. Weaknesses in these methods can be observed in addressing the dynamics of a mine and by not providing a precise dispatching solution. In this paper, we present a solution based on Multiagent Systems (MAS) where the equipment items are represented by intelligent agents that interact with each other to meet the production goals at a minimum cost. The results obtained by applying the MAS in a simulated open-pit mine with actual data show more specific solutions than the current centralized solutions in a practical calculation time frame. In addition, the MAS decreases the truck costs on average by 18%.

Keywords: Truck dispatching · Open-pit mine · Multiagent systems · Scheduling

1 Introduction

Material handling is an important logistic process for open-pit mining since it can amount to up to 50% of the operational cost [1]. In this process, trucks and shovels work together to extract and to transport all the material required by the operational plan at minimum cost. In order to reach this objective, dispatching trucks efficiently becomes an important task. However, it is a hard task due to the number of the variables and the dynamics of the environment where the equipment items operate.

The current systems that support the dispatching of trucks follow a centralized approach that commonly uses an allocation model that assigns trucks to trips between loading and unloading points [7]. This solution does not provide a precise operation sequence of the equipment, therefore it cannot secure an efficient use of the equipment. To improve the efficiency of the material handling process, we developed a multiagent system (MAS).

The paper demonstrates the applicability of the MAS. To do this, first, we compare the solutions provided by the MAS against a mathematical model and second, we use actual data from a Chilean open-pit mine to compare the actual transported material and the material that could have been transported following the solution proposed by the MAS.

The remainder of this paper is structured as follows: Sect. 2 presents some background of truck dispatching in open-pit mines. A mixed integer linear programming (MILP) scheduling model is described in Sect. 3. Section 4 presents the distributed approach based on MAS. Section 5 presents the results and discusses the evaluation of the MAS approach in a case study. Finally, conclusions and outlook are presented in Sect. 6.

2 Problem Definition

In the open-pit mine material handling process shovels extract materials and load trucks that transport these materials to different destinations at the mine. If the extracted material is waste, it is transported to a waste dump, and if it is ore, it is transported to a crusher or a stockpile. Figure 1 shows all operations that a truck must perform to transport materials from a loading point to an unloading point. This is called the truck cycle. This cycle is performed and repeated by each truck until the shift ends.

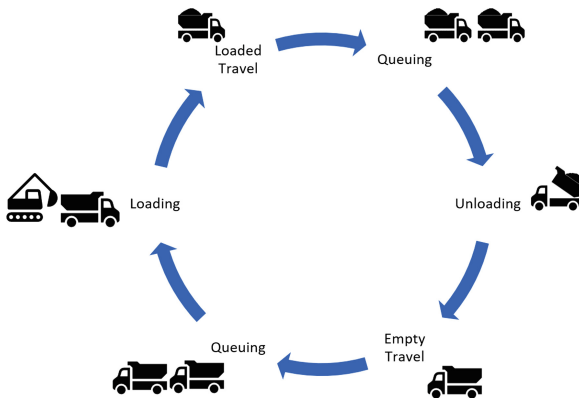


Fig. 1. The truck cycle.

At first sight, truck dispatching in open-pit mines seems to be a kind of vehicle routing problem (VRP). However, although there are some similarities,

there also are differences: there is no start and end node (depo), the trucks must go to a pickup node (shovel) and then to a delivery node (crusher or stockpile) and repeat this sequence during the entire shift. This implies that a truck can visit a node more than once. The travel times between nodes are short and the number of nodes is too lower than the number of trucks. This produces waiting time at nodes. These differences, make difficult to apply a pure VRP model.

Different centralized systems have been implemented to support the dispatching of trucks based on mathematical programming, heuristic processes or simulation. The strengths of these methods are their maturity and their well-known implementation. However, the weaknesses can be observed in addressing the dynamics of a mine [2], by not providing a precise solution [7], the use of estimated information [6], and the time needed to calculate a dispatching solution when the model is too complex. A common strategy applied in some centralized systems is based on the multistage approach [1]. This approach uses a guideline that is computed in the upper stage. Then, this guideline is used by the lower stage as a reference to make real-time dispatching decisions.

Despite the use of these systems, the trucks and shovels do not operate efficiently since queues of trucks are built-up in front of shovels and crushers, as well as idle time of shovels. Therefore the problem is how to improve the efficiency in the material handling process.

Alternatively, a more specific solution that would allow the equipment items to operate more efficiently would be to set up schedules for each equipment item with all the operations that it must perform, pointing out the start times, end times, etc.

3 Formalization

To address the problem, a mathematical model is formulated based on the work of Patterson [7], which uses a MILP with the objective of minimizing the energy consumption of the shovels and trucks taking into account the targets of the production plan in an open-pit coal mine. The model uses a sequence of loading ‘slots’ per shovel to organize the operations of trucks and shovels.

In our model, trucks can be assigned to any shovel. The shovels are assigned to one pit and the material extracted by a shovel must be transported to a destination throughout the shift. Shovels can load one truck at once. At a crusher, one truck can unload at once, whereas in a waste dump or a stock pile several trucks can unload simultaneously. The notation of sets, indices, parameters and decision variables used in the model is shown in Table 1.

The objective function (1) is to minimize the cost (in terms of time) that is taken by the shovels and the trucks to perform the operations. Restriction (2) ensures that at most one truck is assigned to each time slot l on each shovel. Restriction (3) ensures that no more than one truck r is loaded on the shovel at once. Restriction (4) ensure that at crushers, no more than one truck can unload at a time. The restriction (5) ensures that an unloading time ($\mu_{s,l}$), starts after the loading starts ($\lambda_{s,l}$) plus the time it takes to perform the loading C_s and

Table 1. Formulation notation.

<i>Set</i>	<i>Index</i>	<i>Description</i>
S	s	Shovels
R	r	Trucks
Ls	l	Slot time of the shovel s
J	j	Destinations
<i>Parameters</i>		
C_s		Loading time of shovel s
C_j		Unloading time at the destination j
C_j^s		Travel time from shovel s to the destination j
C_s^r		Travel time of truck r to shovel s (only at the beginning of the shift)
$C_{s'}^j$		Travel time from the destination j to next shovel s'
A_r		Truck capacity
δ_s		The target of extracted material by shovel s
M		Sufficiently large positive number
<i>Decision variables</i>		
$X_{r,s,l}$		1 if the truck r loads at shovel s in the time slot l, otherwise 0
$\lambda_{s,l}$		Loading start time of shovel s in time slot l
$\mu_{s,l}$		Unloading start time of material extracted by shovel s in time slot l
$\lambda_{r,s,l,s'l'}^{seq}$		1 if truck r was loaded by shovel s in time slot l before being loaded in shovel s' and slot time l'. Otherwise 0

the travel time to the destination C_j^s . The restriction (6) ensures that the next loading of a truck in l' must be after the truck ends the unloading in l .

$$Min \sum_{\forall s,l} (C_s^j + C_s + C_j^s + C_j) \tag{1}$$

$$\sum_{\forall r} X_{r,s,l} \leq 1 \quad \forall l, s \tag{2}$$

$$\lambda_{s,l+1} - \lambda_{s,l} \geq C_s \quad \forall l, s \tag{3}$$

$$\mu_{s,l+1} - \mu_{s,l} \geq C_j \quad \forall s, l \tag{4}$$

$$\mu_{s,l} \geq \lambda_{s,l} + C_s + C_j^s \quad \forall l, s \tag{5}$$

$$\lambda_{s',l'} \geq \mu_{s,l} + C_{s'}^j + C_j - M(2 - \lambda_{r,s,l,s'l'}^{seq} - X_{r,s,l}) \quad \forall r, l, s, s'l' \tag{6}$$

$$X_{r,s,l} = \sum_{s',l'} \lambda_{r,s',l',s,l}^{seq} \quad \forall r, l, s \tag{7}$$

$$X_{r,s,l} = \sum_{s',l'} \lambda_{r,s',l',s,l}^{seq} \quad \forall r, l, s \tag{8}$$

$$\sum_{r,l} X_{r,s,l} A_r \geq \delta_s \quad \forall s \tag{9}$$

$$\lambda_{s,1} \geq C_s^r - M(1 - \lambda_{r,0,0,s,1}^{seq}) \quad \forall r, s \quad (10)$$

$$\sum_{s,l} \lambda_{r,0,0,s,l}^{seq} = 1 \quad \forall r \quad (11)$$

$$\sum_{s,l} \lambda_{r,s,l,0,0}^{seq} = 1 \quad \forall r \quad (12)$$

Restrictions (7) and (8) ensure that the sequence of each truck has one predecessor and a successor. Restriction (9) ensures that the proposed loading targets in the production plan for each shovel are met. Restriction (10) ensures that a truck travel time to its first loading point must be considered at the beginning of the shift. Restriction (11) ensures that all trucks start in a dummy pit. This pit represents the initial place where a truck is at the beginning of the shift. Restriction (12) ensures that all trucks also end the shift in a dummy pit.

4 An Alternative Solution Approach: Multiagent System

A Multiagent System (MAS) is a system collection of agents that are intelligent software programs representing an entity from the real world and/or provide a certain service [4]. The agents act autonomously and make decisions to reach the objectives of their represented entities using their specific data, communication mechanisms and sharing their knowledge. A problem can be divided into smaller problems that the agents can solve optimally due to the smaller complexity of the problem.

4.1 Scheduling MAS Architecture

The objective of the implemented MAS is to accomplish the goals of the production plan at minimal cost. Applying this approach allows us to model truck dispatching in a way that is closer to reality and to avoid the weaknesses of centralized systems. The agents implemented in the MAS include the following ones:

- Truck agent: This agent represents a truck of the real world. Its objective is to create a schedule of the operations of the truck at minimal cost. The main specific data used are capacity, loaded velocity, empty velocity, spotting time and unloading time. In addition, the agent uses the layout of the mine. The agent can play the role of a participant in a negotiation process.
- Shovel agent: This agent represents a shovel of the real world. Its objective is to create a schedule of the operations of the equipment that it represents considering its target in the production plan. The main specific data are capacity, dig velocity, load velocity and the destination of extracted material. The agent can play the role of an initiator in a negotiation process.
- UnloadingPoint agent: This agent represents a crusher, stockpile or waste dump of the real world. Its objective is to create a schedule of the operations of the equipment that it represents. The main specific data is the number of trucks unloading simultaneously.

4.2 Interaction in the Scheduling MAS

In order to create the schedules, the agents must interact with each other using the Contract Net Protocol (CNP) [9] which is a well-known negotiation mechanism for task sharing with near-optimal solutions. In this context, the CNP works as follows: a Shovel agent starts a negotiation process sending a call for proposals (CFP) to the Trucks agents pointing out the time when the shovel is available to load a truck and the idle time from the last loading.

When a Truck agent receives the CFP it must evaluate it. The agent checks its schedule and asks the UnloadingPoint agent for information about the prospective waiting time. With this information, the Truck agent calculates the arrival time and the cost to perform all the operations and decides to send a proposal or to refuse the CFP. If it sends a proposal, the Truck agent waits for the answer. The Shovel agent receives the proposals and after receiving all proposals or if the deadline is expired, looks for the best proposal. Then, the Shovel agent sends an acceptance message to the Truck agent that offered the best proposal and sends a rejection message to the other Truck agents. The Truck agent that receives the acceptance of its proposal adds a new assignment to its schedule. The Shovel agent adds it to its schedule. If the Shovel agent does not receive proposals, the negotiation is finished.

As the agents work in parallel, several CNP negotiations are done concurrently. As a consequence, a Truck agent may receive several CFPs. If the Truck agent sends a proposal answering one of this CFPs, it must wait for the answer from the Shovel agent, and therefore, the other received CFPs are refused. This situation can generate that the Truck agent refuses a CFP that is a better option than the CFP answered previously. This problem is also called “the eager bidder problem” [8]. To avoid this problem, a confirmation stage was included in the CNP that works as follows: when the Shovel agent finalizes the evaluation of the proposals, it sends a confirmation message to the Truck agent with the best proposal. The Truck agent that receives the confirmation message, could refuse the confirmation (in the case that it has received a better CFP), otherwise it can accept the confirmation. If the Truck agent refuses the confirmation of the Shovel agent, the Shovel agent sends a confirmation message to the next best proposal received. In this way, the Truck agent could decommit a previous proposal sent. If the Shovel agent receives only rejections from the Truck agents, the negotiation process is ended. Figure 2 depicts the interaction between the agents using the CNP with confirmation stage. Table 2 shows a schedule example for a truck created by the MAS using this protocol.

4.3 Decision Making

The decision making process among agents is one of the most important characteristics of a MAS. A bad design of the decision making could generate bad results or lets the agents take more time for their decisions affecting the performance of the MAS.

Table 2. Example of schedule created for a truck.

Assignment	Destination	Trip start time	Arrival time	Spotting start time	Loading or unloading start time	Assignment end time
0	Shovel.01	00:47:01	01:20:23	01:20:23	01:21:36	01:23:12
1	WasteDump.02	01:23:12	01:32:33	01:32:33	01:32:33	01:33:23
2	Shovel.04	02:10:39	02:18:47	02:18:47	02:20:00	02:21:12
3	WasteDump.03	02:21:12	02:26:38	02:26:38	02:26:38	02:27:28
4	Shovel.04	02:27:28	02:31:37	02:31:37	02:32:50	02:34:02
5	WasteDump.03	02:34:02	02:39:28	02:39:55	02:39:55	02:40:45

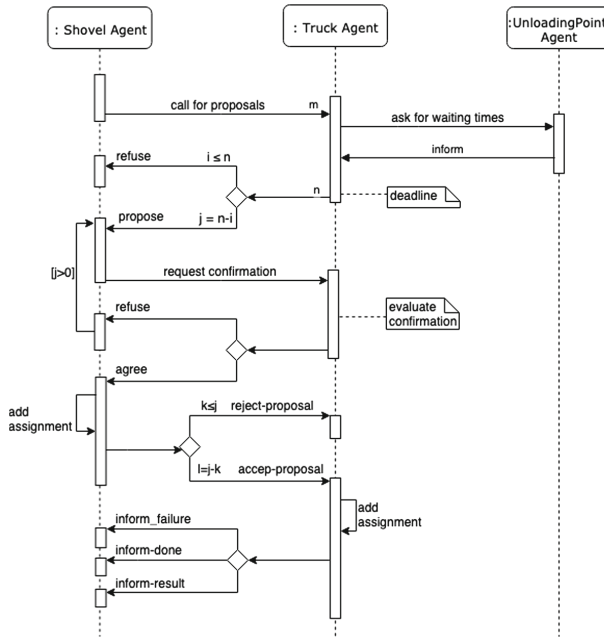


Fig. 2. The interaction between the agents using the CNP with the confirmation stage.

The Shovel agents receive proposals from the Truck agents. These proposals mention the time that a truck could start the loading at the shovel, and the time that it takes to perform all the operations. After receiving all the proposals (or if the deadline is expired) the Shovel agent evaluates all the proposals using a utility function. This function promotes those proposals that propose to start the loading on time and with the least time to perform all operations. In this way, the Shovel agent selects the proposal that minimizes its idle time and offers the least truck cost.

The Truck agent must decide, after receiving a CFP, if the offer can be performed by the truck. It must determine if the loading time offered by the

shovel fits the schedule of the truck. If there is no time slot, the Truck agent rejects the CFP. If there is a time slot, the Truck agent must calculate the total time that it takes the truck to perform all the operations and it determines if the offer is suitable for the time slot. In this process, the Truck agent applies the Dijkstra algorithm [3] to find the shortest paths (from the last unloading point to the shovel and from the shovel to the destination of the material extracted by that shovel) and calculates the travel times. If the offer suits the time slot, the Truck agent sends a proposal, otherwise rejects the CFP.

Another decision of the Truck agents is on the confirmation stage. After receiving a confirmation message from the Shovel agent, the Truck agent must decide to confirm the message or reject it. If it is taking part in another negotiation process with a potential better award (in this case a negotiation with a lower cost for the truck) the Truck agent will reject the confirmation message of the Shovel agent, otherwise it will accept it.

If the Shovel agent completes a negotiation process without a winner, it starts a new negotiation process, but increases the loading time offered by one minute this time. However, it could happen that the negotiation ends again without a winner, and the Shovel would start another negotiation process adding another minute to the offer. This situation would generate idle time in the schedule of the shovel. To avoid this, the Truck agents consider the shovel idle time from the last loading. If the shovel idle time from the last loading is less than one minute and the Truck agent is taking part in another negotiation with a potential better award, the agent rejects the confirmation message. Nevertheless, if the shovel idle time is higher or equal to one minute, the Truck agent must confirm it. In this way, the Truck agents prefer to achieve the goals of the production plan instead of decreasing their own cost.

5 Results and Discussion

Two experiments were done to validate the approach. The purpose of the first experiment was to compare the time that takes the MAS to generate schedules against the time that takes an exact solver with the implementation of the mathematical model presented in Sect. 3. The purpose of the second experiment was to compare the truck cost obtained by the MAS against actual data.

The experiments use actual data from an open-pit copper mine in Chile. In that mine, the equipment items operate in shifts of 12 hours and the material handling is done with a heterogeneous fleet of trucks and shovels. The specific data of the agents were taken from the actual real-world data. The actual data were generated by DISPATCH (TM), which is a centralized system based on dynamic programming [5]. The implemented MAS was deployed and executed in PlaSMA [10], which is a simulation platform for MAS. The implementation of the mathematical model was done in CPLEX.

In the first experiment, several simulations run with different parameters. All instances use the same data, i.e., the same mine layout, and trucks and shovels with the same characteristics. The differences are the number of equipment items

and the length of the shift. Table 3 shows different instances (H: length of the time horizon of the shift; the number of shovels; the number of trucks) and the performance results of the MAS and CPLEX in term of calculation time for the schedules.

Table 3. Times needed for the MAS and CPLEX to generate the schedules.

Instance	H	Shovels	Trucks	MAS time (min)	CPLEX time (min)
1	1	1	10	0,04	0,06
2	3	3	25	0,43	5,06
3	6	5	40	2,45	>180
4	9	7	60	4,52	–
5	12	9	85	16,74	–

In the case of the bigger instances, the MAS provides the schedules in a practical time frame of about 16 min on a standard PC. This is due to the characteristics of the MAS technology, that lends itself to a mainly distributed organization, parallel processing, and a lower computational power requirement. The results obtained from the mathematical model implemented in CPLEX show that is not possible to get solutions for the bigger instances in practical time. This is because of the increase of variables generates a large number of combinations that the solver must evaluate.

In the second experiment, 5 real-world shifts were simulated. The MAS generated schedules for the shovels to extract the same amount of material extracted in the actual data. Table 4 shows a comparison between the actual transported material and the material that could have been transported following the schedules proposed by the MAS.

Table 4. Comparison of the production target cost of MAS schedule vs actual data.

Id	Shift (hours)	Shovels	Trucks	Actual material transported (tons)	Actual travel time (hours)	Simulated material transported (tons)	Simulated travel time (hours)	Delta material transported	Delta travel time
1	12	11	99	350.117	821,9	351.659	597,44	+0,44%	–27,31%
2	12	12	96	350.005	796,36	351.895	668,37	+0,54%	–16,07%
3	12	11	98	404.921	849,97	405.903	713,33	+0,24%	–16,08%
4	12	12	94	409.555	813,01	411.345	676,99	+0,44%	–16,79%
5	12	12	93	386.973	783,49	389.404	656,58	+0,63%	–16,2%

The solution provided by the MAS achieves the targets of the production plan at, on average, decreased cost by 18% even with marginally bigger production goals. One of the reasons for these savings is that the MAS travel times of a truck are smaller than the travel times in the actual data since the agents in the

MAS use the shortest path for their travels, whereas the truck operators in the real world decide by themselves which path to follow. Another reason is the use of specific data to allow for more adapted calculations of the operation times of every equipment, and in this way, the agents can create more appropriate and efficient schedules.

6 Conclusions

A Multiagent System for truck dispatching in open-pit mines has been presented. Experimental results show that the MAS provides more precise solutions than a centralized system within a practical computation time frame. In addition, the generated schedules by the MAS are more efficient since they decrease the truck cost on average by 18% meeting even marginally bigger production goals.

Future investigations will address two aspects: on the one hand, the dynamic of the material handling process, e.g., dealing with a major change in the mine such as equipment failures or changes in the mine layout. In this case, the affected agents will have to react appropriately, interacting with each other to update their schedules. On the other hand, the MAS will be compared against other methods that provide solutions in practical frame time such as metaheuristics algorithms.

References

1. Alarie, S., Gamache, M.: Overview of solution strategies used in truck dispatching systems for open pit mines. *Int. J. Min. Reclam. Environ.* **16**(1), 59–76 (2002). <https://doi.org/10.1076/ijsm.16.1.59.3408>. <http://www.tandfonline.com/doi/abs/10.1076/ijsm.16.1.59.3408>
2. Bastos, G.S., Souza, L.E., Ramos, F.T., Ribeiro, C.H.C.: A single-dependent agent approach for stochastic time-dependent truck dispatching in open-pit mining. In: 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 1057–1062 (2011). <https://doi.org/10.1109/ITSC.2011.6082902>
3. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numer. Math.* **1**(1), 269–271 (1959)
4. Gath, M.: Optimizing transport logistics processes with multiagent-based planning and control. Ph.D. thesis (2015)
5. Munirathinam, M., Yingling, J.C.: A review of computer-based truck dispatching strategies for surface mining operations. *Int. J. Surf. Min. Reclam. Environ.* **8**(1), 1–15 (1994). <https://doi.org/10.1080/09208119408964750>
6. Newman, A.M., Rubio, E., Caro, R., Weintraub, A., Eurek, K.: A review of operations research in mine planning. *Interfaces* **40**(3), 222–245 (2010). <https://doi.org/10.1287/inte.1090.0492>
7. Patterson, S.R., Kozan, E., Hyland, P.: Energy efficient scheduling of open-pit coal mine trucks. *Eur. J. Oper. Res.* **262**(2), 759–770 (2017). <https://doi.org/10.1016/j.ejor.2017.03.081>
8. Schillo, M., Kray, C., Fischer, K.: The eager bidder problem: a fundamental problem of DAI and selected solutions. In: Proceedings of the 1st international Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2002), January, pp. 599–606 (2002). <https://doi.org/10.1145/544862.544886>

9. Smith, R.G.: The contract net protocol: high-level communication and control in a distributed problem solver. *IEEE Trans. Comput.* **C-29**(12), 1104–1113 (1980). <https://doi.org/10.1109/TC.1980.1675516>
10. Warden, T., Porzel, R., Gehrke, J.D., Herzog, O., Langer, H., Malaka, R.: Towards ontology-based multiagent simulations: plasma approach (2007)