Application of Multiagent System and Tabu Search for Truck Dispatching in Open-pit Mines

Gabriel Icarte Ahumada[1,2][a] and Otthein Herzog[3,4,5][b]

[1]International Graduate School for Dynamics in Logistics (IGS), University of Bremen, Bremen, Germany

[2]Faculty of Engineering and Architecture, Arturo Prat University - UNAP, Iquique, Chile

[3]University of Bremen, Germany

[4]Jacobs University Bremen, Germany

[5]Tongji University Shanghai, China

Keywords: Truck Dispatching, Open-pit Mine, Multiagent Systems, Tabu Search, Scheduling, Rescheduling.

Abstract: An important and complex process in the mining industry is the material handling process. In this process, trucks must transport materials extracted by shovels to different places at the mine. To enable efficient material handling processes, the decision on the destination of a truck is crucial. Currently, this process is supported by an approach based on centralized systems that apply dispatching criteria. A disadvantage of this approach is not providing a precise dispatching solution because of missing knowledge about potentially changed external conditions and the dependency on a central node. We previously developed a multiagent system (MAS-TD) to solve this problem. In the MAS-TD, intelligent agents that represent real-world equipment interact with each other to generate schedules. In this paper, we evaluate the MAS-TD by comparing it against a Tabu Search procedure. In the evaluation, simulated scenarios based on actual data from a Chilean open-pit mine were used. The results show that both MAS-TD and the Tabu Search procedure are suitable methods to solve the truck dispatching problem in open-pit mines. However, the schedules generated by MAS-TD are more efficient than the schedules generated by the Tabu Search.

1 INTRODUCTION

In open-pit mines, the material extracted by shovels must be transported by trucks to different destinations at the mine. If a shovel is extracting ore, the extracted material must be transported by a truck and unloaded into a crusher or onto a stockpile. If the extracted material is waste, it must be transported to a waste dump. This process is called material handling and it is an important process for these kinds of mines since the costs involved in this process can represent up to 50% of the entire operational cost (Alarie and Gamache, 2002). Figure 1 shows the operations that a truck must perform from a loading point (shovel) to an unloading point (crusher, stockpile, or waste dump). These operations are performed repeatedly by a truck until the shift ends.

Open-pit mines are closed systems where the operations performed are affected by a dynamic environment. For instance, equipment failures, changes in weather conditions or to the state of the routes affect the performance and availability of equipment and generate delays in material handling (Adams and Bansah, 2016). In this context and, with the process's stochasticity, determining a new destination for the

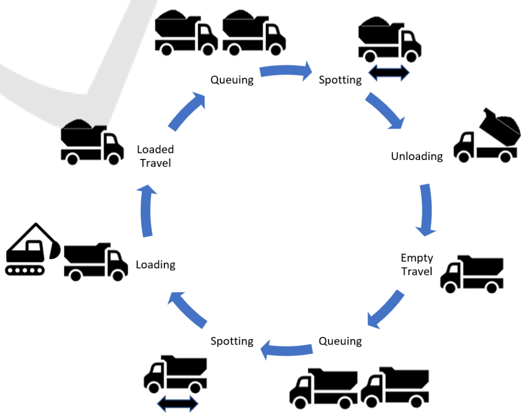

Figure 1: The truck cycle. Adapted from (Icarte et al., 2020).

[a] https://orcid.org/0000-0002-1997-0053
[b] https://orcid.org/0000-0003-4781-2551

Icarte Ahumada, G. and Herzog, O.
Application of Multiagent System and Tabu Search for Truck Dispatching in Open-pit Mines.
DOI: 10.5220/0010391101600170
In *Proceedings of the 13th International Conference on Agents and Artificial Intelligence (ICAART 2021) - Volume 1*, pages 160-170
ISBN: 978-989-758-484-8

truck is not easy.

Currently, the material handling process is supported by centralized systems based on Operation Research methods, heuristic procedures, or simulation modeling (Icarte and Herzog, 2019). Most of these systems use a multistage approach (Alarie and Gamache, 2002), which computes a guideline in the first stage. A later stage uses this guideline and a dispatching criterion (e.g., the current production level of the shovels) to dispatch the trucks in real-time each time that the latter requires.

Despite using these solutions, material handling in open-pit mines is not performed efficiently. For instance, trucks may queue up in front of shovels or crushers while other shovels wait for trucks. This generates inefficiency, high costs, and not achieving production targets. Researchers describe that the current systems don't solve the problem appropriately because they do not provide a precise representation of the activities performed by the equipment (Patterson et al., 2017) and the use of estimated information (Chang et al., 2015; Costa et al., 2005; Krzyzanowska, 2007; Newman et al., 2010).

Icarte et al. (Icarte et al., 2020) developed an alternative solution that allows to organize more efficiently the equipment items operations. The solution is based on a Multiagent system in which intelligent agents represent real-world equipment. These agents interact with each other to generate their schedules.

In this paper, we extend the work presented in (Icarte et al., 2020) by comparing the MAS-TD against a Tabu Search procedure. Many papers have demonstrated the application of Tabu Search for scheduling problems, even for truck dispatching, e.g., (Liao et al., 2014). The comparison was performed by the simulation of scenarios based on actual data from a Chilean open-pit mine.

The remainder of this paper is structured as follows: Section 2 presents related work. Section 3 gives an overview of the developed MAS-TD. Then, Section 4 presents the Tabu Search procedure. Section 5 provides the comparison between the MAS-TD and the Tabu Search procedure and a discussion. Finally, conclusions and outlook are presented in Section 6.

2 RELATED WORK

The truck dispatching problem in open-pit mines has been faced in many publications during the last years. These articles show different methods that try to achieve two goals: improve productivity and reduce operating costs (Alarie and Gamache, 2002). For example, Ozdemir and Kumral (Ozdemir and Kumral, 2018) developed an agent-based Petri net simulation model to check whether production targets are feasible and the extent to control the head grade in mineral processing by considering the uncertainties in the mining operation. Xu et al. (Xu et al., 2019) developed an Approximate Dynamic Programming (ADP) algorithm based on Q-Learning. The algorithm implements two models: a static model and a dynamic model. The static model allocates trucks to a loading and an unloading job. The model employs an event-driven method to define the stage when an idling truck appears at any time (maybe more than one truck). Bakhtavar and Mahmoudi (Bakhtavar and Mahmoudi, 2020) developed a two-phase scenario-based robust optimization (SBRO) model by considering the maximization of production, control of ore grade sent to the crusher, minimization of waiting times for trucks and shovels, and trucks with different capacities.

Most of the reports on the truck dispatching problem in open-pit mines follow an allocation model, in which the destination of a truck is determined when it is required. Only a few publications have modeled the problem as a scheduling problem, e.g., Chang et al. (Chang et al., 2015) and Patterson et al. (Patterson et al., 2017) proposed algorithms that generate an initial schedule, which is improved using a metaheuristic method. Their results show that the algorithms generate schedules for different size instances with good results and performance in practical time frames (for the mining industry). Icarte et al. (Icarte et al., 2020) developed a multiagent system with agents representing trucks and shovels. The agents interact with each other to generate schedules. Their results show that the MAS-TD provides schedules in practical time frames and can handle environment dynamics.

3 MULTIAGENT SYSTEM FOR TRUCK DISPATCHING IN OPEN-PIT MINES

In the developed MAS-TD, the agents interact with each other to generate schedules and to maximize the production at minimum cost. Moreover, the agents update the generated schedules when a major unforeseen event occurs at the mine. Table 1 shows the implemented agents, their objectives, and properties.

3.1 Agent Interactions

In order to generate the schedules, the agents negotiate among them by applying the improved Contract Net protocol presented in (Icarte et al., 2020). In brief,

Table 1: Agent description. Adapted from (Icarte et al., 2020).

Agent	Real-world representation	Objective	Properties
truckAgent	Trucks	Create a schedule of the activities of the truck at minimum cost	Capacity, loaded velocity, empty velocity, spotting time and unloading time, layout of the mine
shovelAgent	Shovels, front loaders	Create a schedule of the activities of the equipment that it represents considering its target in the production plan	Capacity, dig velocity, load velocity and the destination of extracted material
unloadingPointAgent	Crusher, stockpiles, waste dumps	Create a schedule of the activities of the equipment that it represents	Number of trucks unloading simultaneously

the *shovelAgents* start negotiation processes sending *call-for-proposal* (CFP) messages to *truckAgents*. In the CFP the point of time is offered when the shovel will be available for loading a truck. *TruckAgents* respond with *proposal* messages pointing out the estimated arrival time at the shovel and the cost of performing all the operations involved. Finally, the *shovelAgent* selects the best proposal. The improved Contract Net protocol enables the agents to manage concurrent negotiations. Figure 2 shows this Contract Net protocol, and Table 2 shows a schedule example for a truck generated by the agents by applying this protocol.

To manage the dynamics of the environment, the agents follow a complete rescheduling strategy. This means that the agents generate new schedules from the point in time where an unforeseen event occur. The agents cancel their assignments, and they interact with each other to generate new schedules.

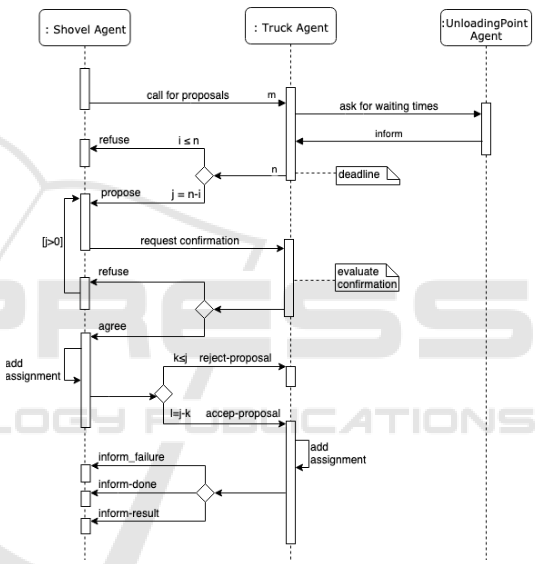

Figure 2: The interaction between the agents using the improved Contract Net protocol with the confirmation stage. Adapted from (Icarte et al., 2020).

3.2 Decision Making

ShovelAgents must decide what is the best received proposal. To make this decision, the *shovelAgents* use the utility function proposed in (Icarte et al., 2020), which promotes those proposals that decrease the shovel's waiting time and minimize the cost to perform the truck operations.

TruckAgents make two decisions. The first one is deciding whether or not to send a proposal to a *shovelAgent* that sent a *call-for-proposal* message. To make this decision, the *truckAgent* checks its schedule and determines if there is a free time slot for the offered time in the CFP. If yes, it calculates the total time to perform all the operations and determines if it fits into its schedule. If it does not fit into the schedule, it sends a *reject* message. If it fits into the schedule, it sends a proposal.

The second one is to decide on whether or not to confirm a previously sent proposal. To make this decision, the *truckAgent* considers the shovel idle time (informed by the *requestConfirmation* message sent by the *shovelAgent*) and the negotiations which the *truckAgent* is taking part in. If the shovel idle time is higher or equal than one minute, the *truckAgent* sends an *acceptConfirmation* message. If the shovel idle time is less than one minute, the *truckAgent* checks if it is taking part in another negotiation with more potential benefit for the truck (i.e., with a lower cost to perform the operations). If yes, the *truckAgent* sends a *refuseConfirmation* message. Otherwise, it sends an *acceptConfirmation* message.

Table 2: Example of schedule created for a truck.

Assignment	Destination	Start Time of the Trip	Arrival Time	Start Time of the Spotting	Start Time of the Loading or Unloading	End Time of the Assignment
0	Shovel.01	00:32:11	00:55:29	01:02:43	01:03:56	01:06:21
1	WasteDump.02	01:06:21	01:25:23	01:25:23	01:26:03	01:27:23
2	Shovel.04	02:11:24	02:20:27	02:30:35	02:31:10	02:34:29
3	WasteDump.03	02:34:29	02:56:18	02:56:18	02:58:28	03:00:25
4	Shovel.04	03:27:28	03:31:37	03:38:37	03:39:59	03:43:25
5	WasteDump.03	03:43:25	03:59:28	03:59:28	04:01:09	04:02:01

4 TABU SEARCH FOR TRUCK DISPATCHING IN OPEN-PIT MINES

Metaheuristics, particularly the Tabu Search (Glover, 1989), are commonly employed to overcome the complexity of scheduling formulations, especially for problems dealing with heterogeneous vehicle fleets (Koç et al., 2016). The Tabu Search algorithm explores neighborhoods to find improved solutions starting from an initial solution. It uses knowledge (tabu list) about recently visited solutions. The algorithm rejects neighborhood moves that result in a solution already in the tabu list to avoid cycling around local optima. During the algorithm's main loop, the best neighborhood that is not in the tabu list is selected and added to the tabu list. If it is the best solution found so far, it is recorded as such. The algorithm terminates after reaching a stop condition and returns the best solution found.

Algorithm 1 implements the Tabu Search described above. In lines 1-5, the initial setup of the algorithm is done. In *sBest* the initial solution provided by Disp-ALG (detail in subsection 4.1) is stored, and is set as the current solution in *bestCandidate*. Also, the tabu list is created and stored in *tabuList*. The main loop starts in line 6. This loop searches for a better solution until a time-out is reached. In line 7, the function *getNeighbors* (*bestCandidate*) generates new feasible solutions through small changes in the stored solution in *bestCandidate*. Changes in a solution are explained in subsection 4.2. Lines 8 - 13 look for the best candidate among the neighbors that is not in the tabu list. If this best candidate is better than the best solution found so far, it is set as the best solution (lines 14-16). The best candidate is added to the tabu list (line 17), and if the tabu list is full, the oldest element is removed (line 18-20). Finally, when the *stopTime* is reached, the algorithm returns the best solution found.

Algorithm 1: Tabu Search.

Input: executionTime
Output:
1: sBest ← DISP-ALG
2: bestCandidate ← sBest
3: tabuList ← ∅
4: stopTime ← currentTime+executionTime
5: tabuList.push(sBest)
6: **while** currentTime<stopTime **do**
7:   sNeighborhood ← GETNEIGHBORS(bestCandidate)
8:   bestCandidate ← sNeighborHood.firstElement()
9:   **for** sCandidate ∈ sNeighborHood **do**
10:     **if not** (tabuList.contains(sCandidate)) **and** (fitness(sCandidate) > fitness(bestCandidate)) **then**
11:       bestCandidate ← sCandidate
12:     **end if**
13:   **end for**
14:   **if** fitness(bestCandidate) > fitness(sBest) **then**
15:     sBest ← bestCandidate
16:   **end if**
17:   tabuList.push(bestCandidate)
18:   **if** tabuList.size > maxTabuSize **then**
19:     tabuList.removeFirst()
20:   **end if**
21: **end while**
22: **return** sBest

4.1 Initial Solution

Disp-ALG builds feasible schedules for the equipment involved in the material handling process. These feasible schedules are used as the initial solution in the Tabu Search algorithm presented above.

The main algorithm (Algorithm 2) consists of 3 steps: the first one looks for a shovel task defined as *(s, j, tll)*. The second step looks for the best truck *t* to perform the shovel task. If the second step is successful, the algorithm proceeds with the third step, which consists of adding the activities and times to the schedules of the truck t and shovel s; otherwise, the shovel task is discarded. These steps are repeated until the last activity of the shovels exceeds the end of the shift or the target in a production plan is achieved. To determine whether these conditions are met, the algorithm invokes the Boolean function *isConditionsMeet*. More details of Disp-ALG can be found in (Icarte et al., 2020).

Algorithm 2: Disp-ALG.

Input: S, T, Plan, H
Output: shovelSchds, truckSchds

```
shovelSchds ← ∅; truckSchds ← ∅;
while not isConditionsMeet do
    s, j, tll ← FindJob(S, shovelSchds, H)
    if s not null then
        T, act, act′ ← FindTruck(T, s, j, tll, truckSchds)
        if t not null then
            Schedule(t,s,j,tll,shovelSchds, truckSchds, act, act')
        end if
    end if
end while
```

4.2 Neighborhoods

The solution space is explored by performing three operators to generate neighborhoods from the initial solution. Figure 3 shows examples of generated possible neighborhoods. The swap operator, shown in Figure 3(a), generates the first one. The swap operator swaps two truck assignments, ignoring unfeasible swaps. A feasible swap is only considered if the swap decreases the cost to perform the assignments. The second neighborhood, shown in Figure 3(b), is generated by the change operator. This operator moves an assignment from one truck to another one only if the second truck can perform the assignment in a shorter time than the first truck. The new assignment operator generates the last neighborhood, shown in Figure 3(c). The new assignment operator adds a new assignment by seeking a free time slot to perform a loading operation in the shovel. Then it looks for a truck to perform the loading operation.

To select the best neighborhood, an evaluation is performed based on the efficiency of the schedules. The ratio *TotalMaterialToBeTransported / TotalCosts* determines the efficiency of schedules. The neighborhood with the highest ratio is selected as the current solution for an iteration in the Tabu Search algorithm.

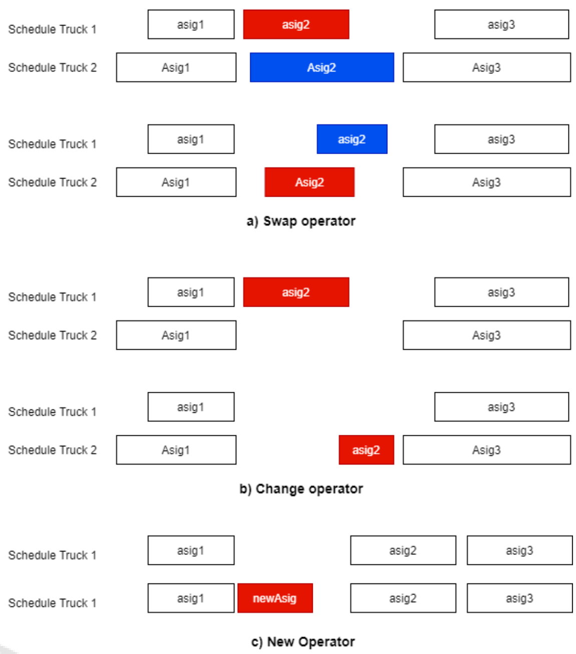

Figure 3: Operators to generate neighborhoods.

5 EVALUATION AND DISCUSSION

The evaluation aims to determine which method is more appropriate to solve the truck dispatching problem in open-pit mines. To achieve this, the evaluation compares the results obtained by applying the MAS-TD and the Tabu Search procedure on simulated scenarios based on actual data.

The evaluation was split into two parts: the first one considers the methods' performance to generate schedules. The second one considers how the methods react to the dynamics of the environment. The following output performance measures are compared in the evaluation:

- Production: It is the total material transported by the trucks during a shift. It is measured in tons.
- Costs: Because the actual data does not provide information on costs such as operator salaries, fuel costs, and maintenance costs, the truck travel times are considered costs. They are measured in hours.
- Required time to generate schedules: It is the computation time that a method requires to generate schedules. It is measured in minutes.
- Hauling cost: The ratio between all the material transported and all the costs to transport these materials. It is measured in tons per hour.

5.1 Experimental Setup

In the evaluation, four simulated scenarios are used. These scenarios are based on actual data from an open-pit copper mine in Chile. The scenarios consider different heterogeneous fleets of trucks and shovels in twelve-hour shifts. Table 3 shows some information about the simulated scenarios.

The actual data, such as velocities and capacities, is used to set the properties of the agents. Table 4 shows the property sets of the agents. The simulations ran in PlaSMA (Warden et al., 2010), which is an agent-based event-driven simulation platform created for the simulation and evaluation of multiagent systems. It focuses on simulating logistics processes and is based on the FIPA-compliant Java Agent Development Framework (JADE) (Bellifemine et al., 2007). All simulations ran on a laptop computer with an Intel Xeon 3 Gigahertz CPU, 32 gigabytes of RAM, and Windows 10.

5.2 Schedule Generation

The evaluation of the schedule generation was split in two parts. The first one aimed to generate schedules for maximizing the production. The second one aimed to generate schedules considering a hypothetical production plan. In both evaluations, the Tabu Search procedure ran with different execution times (5, 10, 15, and 20 minutes).

5.2.1 Schedule Generation for Maximizing Production

Regarding computation time to generate the schedules, both methods generated schedules in practical frame times (for the mining industry). Figure 4 shows the computation time of the methods to generate the schedules for each shift. MAS-TD generated the schedules in 20.48 minutes on average. The Tabu Search procedure generated the schedules in 5, 10, 15, and 20 minutes as it was pre-set. Also, Disp-ALG was included in the comparison since it generates the initial schedules used by Tabu Search. Disp-ALG was the fastest method to generate schedules. On average, Disp-ALG required 0,1 minutes to generate the schedules.

Figure 5 shows the quantity of the material transported in the schedules generated by the methods. Regarding production, the schedules generated by Disp-ALG move the lowest quantity of material in each of the four scenarios. After 20 minutes running time, Tabu Search improves the schedules generated by Disp-ALG by 0.76% on average. MAS-TD

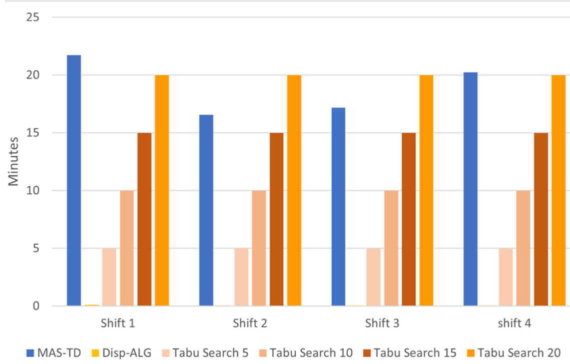

Figure 4: Computation time to generate schedules of MAS-TD, Tabu Search and Disp-ALG.

is the method whose schedules could move the highest amount of material. The activities in its schedules transported 9.93% more material than the activities in the schedules generated by Tabu Search.

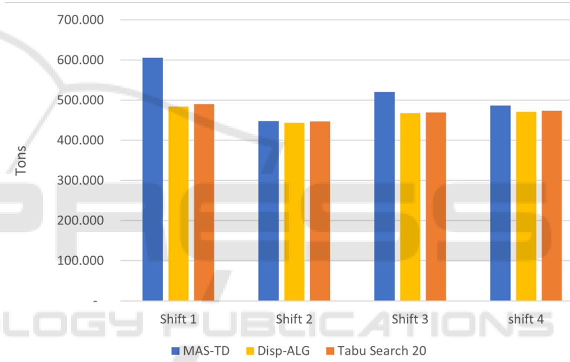

Figure 5: Production of MAS-TD, Tabu Search and Disp-ALG.

Figure 6 shows the costs of the generated schedules by the methods for each shift. Regarding costs, Tabu Search decreases the costs of the schedules generated by Disp-ALG by 0.92% on average. The schedules generated by MAS-TD are more expensive than the activities in the schedules generated by Tabu Search by 4.59% on average. This is because the MAS-TD generated schedules with more production, and therefore more operations to be performed than the schedules obtained by Tabu Search.

Figure 7 shows the efficiency of the generated schedules by the methods for each shift. Regarding the efficiency of the schedules, determined by the ratio of *total Material Transported / Costs*, it is a little bit higher in MAS-TD compared to the Tabu Search and Disp-ALG, except for the shift 1 and 3, in which the difference is higher. The generated schedules by MAS-TD are more efficient than the ones obtained by Tabu Search by 5.26% on average. Tabu search in-

Table 3: Simulated scenarios.

Scenario ID	Number of Trucks	Number of Shovels	Material Transported (tons)	Cost (hours)	Brief Description
1	97	10	455.696,23	780,74	Highest production
2	102	12	378.069,92	772,84	Highest number of trucks
3	95	13	325.899,61	757,62	Highest number of shovels
4	96	12	394.759,09	791,59	Many events (truck and shovel failures)

Table 4: Property values for the simulations.

Equipment	Property	Unit	Min Value	Max Value
Trucks	Velocity loaded	[km/hr]	20	25
	Velocity empty	[km/hr]	40	55
	Capacity	[tons]	230	370
	Spotting time	[sec]	20	80
	Current load	[tons]	0	370
Shovel	Capacity	[tons]	35	80
	Load time	[sec]	8	30
	Dig time	[sec]	8	20
	Destination	Location at mine (crusher, stockpile or waste dump)		
Crusher	Equipment discharging	[number of trucks]	1	1
Stockpile	Equipment discharging	[number of trucks]	1	20
Waste Dumps	Equipment discharging	[number of trucks]	1	20

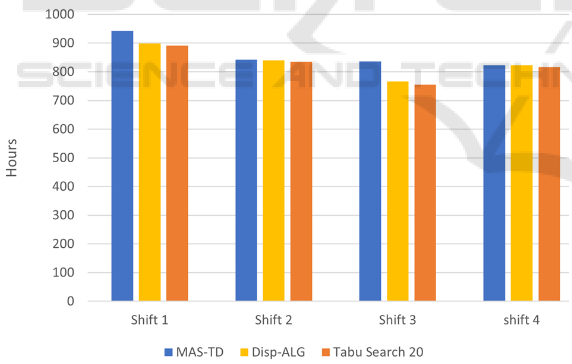

Figure 6: Costs of MAS-TD, Tabu Search and Disp-ALG.

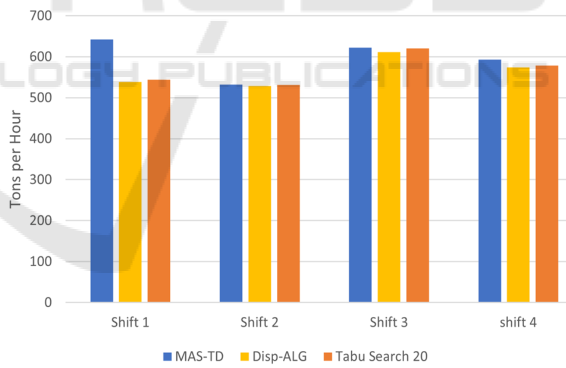

Figure 7: Efficiency of the generated schedules by the methods for each shift.

creases the efficiency of the generated schedules by Disp-ALG through the improvement operations by 0.93% on average.

5.2.2 Schedule Generation with a Production Plan

Differently from the previous evaluation, all methods were set to achieve the targets of a production plan at minimum costs. The production plan is based on actual data. Figure 8 shows the computation time of the methods to generate schedules for each shift.

Regarding computation times, the methods have the same pattern as the previous evaluation (for maximizing the production). It means that Disp-ALG is the fastest method to generate schedules, MAS-TD generates schedules in practical time frames (for the mining industry), and Tabu Search generates the schedules in 5, 10, 15, and 20 minutes as it was pre-set. Compared to the previous evaluation, MAS-TD and Disp-ALG generate the schedules in less time. This is because the methods are set to achieve a lower production level than in the previous evaluation.

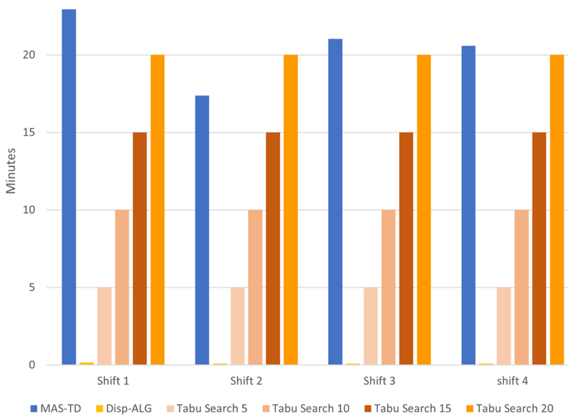

Figure 8: Computation time of the methods to generate schedules (with production plan) for each shift.

Figure 9 shows the targets in the production plan, and the production level reached for the methods for each shift. Regarding production, all the methods generate schedules that reach the production level in the production plan. Due to the fact that generated schedules by Disp-ALG reach the production level pointed out in the production plan, Tabu Search does not improve these schedules (from the perspective of the production level).

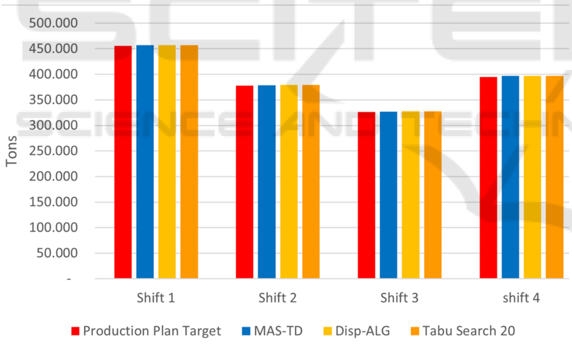

Figure 9: Production level planed and production level reached by the methods for each shift.

Figure 10 shows the costs of the generated schedules by the methods for each shift. Regarding costs, Tabu Search decreases the costs of the schedules generated by Disp-ALG by 1.44% on average. The costs in the schedules generated by MAS-TD are lower than the costs in the obtained schedules by Tabu Search by 1.32% on average.

Figure 11 shows the efficiency of the generated schedules by the methods for each shift. Regarding the schedules' efficiency, it is a little bit higher in MAS-TD compared to the efficiency of Tabu Search and Disp-ALG. On average, the generated schedules by MAS-TD are more efficient that the ones obtained by Tabu Search by 1.39%. Tabu search increases the

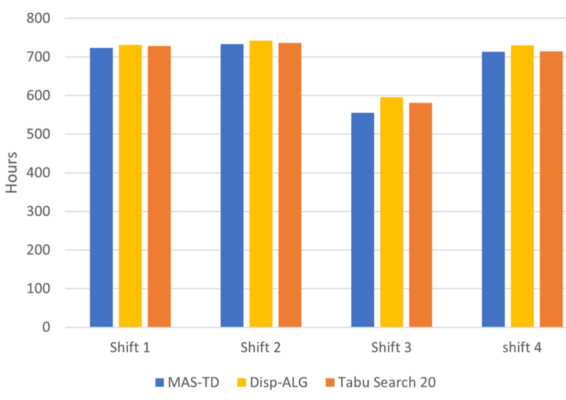

Figure 10: Costs of MAS-TD, Tabu Search and Disp-ALG.

efficiency of the generated schedules by Disp-ALG through the improvement operations by 1.46% on average. This increase in efficiency occurs due to the reduction of the costs calculated by the Tabu Search.

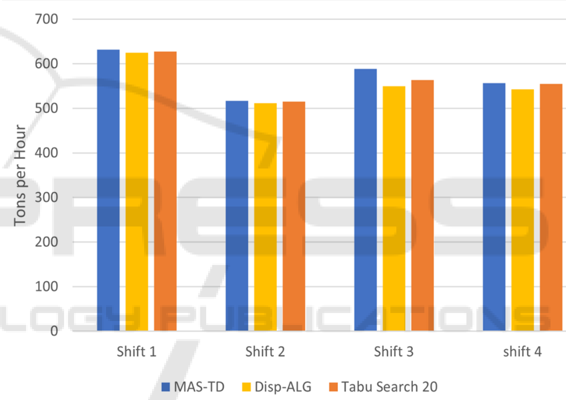

Figure 11: Efficiency of generated schedules by MAS-TD, Tabu Search and Disp-ALG.

5.3 Rescheduling

To evaluate rescheduling, only scenario four was simulated, because in this scenario many equipment items experience failures at different times during the shift. In each event, the methods regenerate the schedules from the moment when the event occurs. The simulations evaluate the capacity of MAS-TD, Tabu Search, and Disp-ALG to react when major events occur at the mine. Tabu search was set to run for five minutes to get a quick solution. In this evaluation, the production reached by the updated schedules and the calculation time to update the schedules were compared.

This evaluation compared two approaches. The first approach is compound only by the MAS-TD, which generates the schedules before the shift starts and generates new schedules each time when an event occurs. The second approach is the compound of Tabu Search and Disp-ALG. In this approach, before the start of the shift, the schedules are generated by Tabu Search. Then, for rescheduling, Disp-ALG is used. Tabu search was not considered for rescheduling because it takes too much time (5 minutes) to react to the mine's new conditions.

Regarding computation time to generate the initial schedules, the first approach (MAS-TD) required around 20 minutes to generate the initial schedules. The second approach (Disp-ALG and Tabu search) was much faster and required a little bit more than five minutes to generate the initial schedules.

Regarding the reaction to the dynamic of the environment, the results demonstrate the same previous pattern: MAS-TD required more time to regenerate all the schedules in comparison to the DISP-ALG. Table 5 shows some computation time to update the schedules at different moment in the shift.

Table 5: Computation time to update the schedules.

Time in the shift	MAS-TD	TS5	Disp-ALG
At the beginning	0:18:15	0:05:00	0:00:05
In the middle	0:10:33	0:05:00	0:00:03
At the end	0:03:46	0:05:00	0:00:02

Regarding production, Figure 12 shows the accumulated production during the shift for each method. The filled lines represent the planned production pointed out in the initially generated schedules. Dotted lines represent the reached production in the updated schedules. The graph shows that the methods can update the schedules when major events occur at the mine. However, the schedules updated by MAS-TD achieve significantly more production (7.19%) than Disp-ALG.

6 DISCUSSION

The results demonstrate that MAS-TD, Tabu Search and Disp-ALG can generate schedules within reasonable time frames (for the mining industry) whether it is for maximizing the production or within the target of a production plan. While Disp-ALG takes only a few second to generate the schedules, MAS-TD required around 20 minutes for the scenarios simulated. The Tabu search computation time depends on the execution time pre-set as parameter. However, it is important to mention that the MAS-TD computation time could be reduced to seconds by employing more

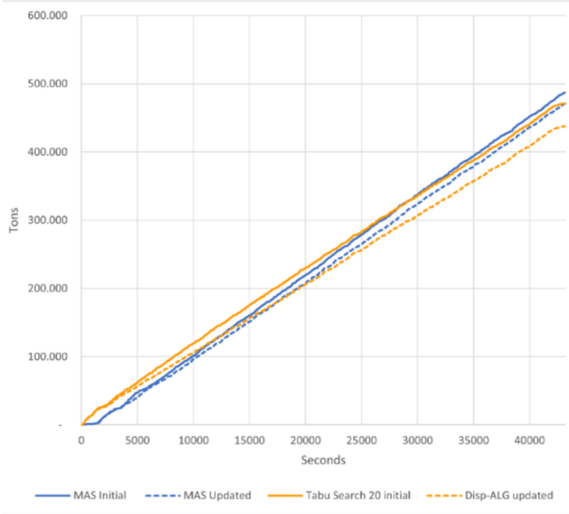

Figure 12: Accumulated production with major events during the shift by original and updated schedules.

cores as the MAS-TD architecture lends itself to concurrent computation tasks.

From the perspective of the production tasks, the schedules generated by the MAS-TD achieve a higher production rate than the generated schedules by Tabu Search and Disp-ALG. This is also the case for the efficiency of the schedules. The better results in the scheduling solution provided by the MAS-TD are due to the fact that the matching between a shovel task and a truck is more precise: The Tabu Search starts the improvement of the schedules generated by Disp-ALG, and Disp-ALG follows a sequential creation of the schedule: it looks for the best truck for a shovel task and then continues with the following shovel task. However, this selected truck may be more appropriate for another shovel task. This may cause an inefficient use of the trucks leading to a lower quality of the schedules. The MAS-TD avoids this situation because of the concurrent negotiation mechanism. The mechanism allows for a *shovelAgent* to decide on the most appropriate truck's proposal. Also, a *truckAgent* can confirm, or choose to not confirm a previously sent proposal to a *shovelAgent* with high idle time. This allows for a more precise match between shovels and trucks.

Considering the results of cost when the methods generated schedules with a production plan, the costs of the schedules generated by MAS-TD were a little bit lower (1.32%) than the costs generated by Tabu Search. This happens due to the fact that *truckAgents* have the chance to select a shovel with a lower cost. Disp-ALG also looks for a truck to perform the operations for a shovel in the shortest time. However, when it finds it, there is no chance of changing it in the future, i.e., if there is another shovel task with less time, it is not possible to change it. Tabu search might find this improvement and make the change, but it might take time to find it.

Regarding rescheduling, MAS-TD, Tabu Search, and Disp-ALG can update schedules when major events occur at the mine. However, the production results and the needed computation time that the methods take to generate the schedules differs among them. Disp-ALG is the fastest method to update the schedules; however, its schedules are the worst in terms of production rates compared to the generated schedules by MAS-TD and Tabu Search.

In the simulation, MAS-TD applied a rescheduling strategy for the complete schedule because it aims at the maximum production level. This strategy takes much more time than the time required by Disp-ALG (as is shown in Table 5), because it generates all the schedules from scratch. However, some assignments can be made within a short time frame, due to two reasons: first, the algorithms that generate the schedules in MAS-TD are anytime algorithms, i.e., they can be stopped at any time, and still will generate a feasible schedule solution. Second, the negotiation process is fast and many assignments can be computed within a few seconds. This allows the fleet to continue working based on a preliminary generated assignment, while MAS-TD continues to generate the schedules. Figure 13 shows that most of the successful negotiations, i.e., assignments computed, are done in less than one second in all scenarios.

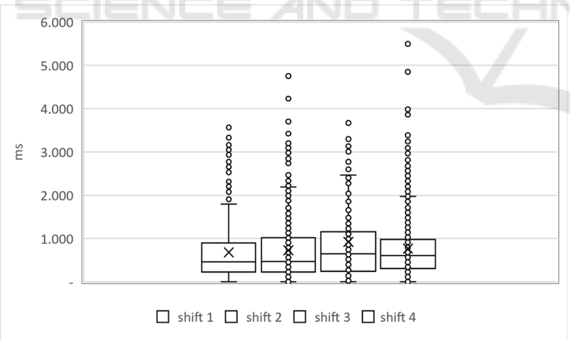

Figure 13: Box and whiskers diagram shows that the mean of the duration of the successful negotiations is around 0.5 seconds.

7 CONCLUSIONS

A major process in open-pit mines is material handling. In this process, dispatching a truck becomes a complex decision because of the stochasticity of the process and the dynamics of the environment. In general, a centralized system supports this process. As an alternative to currently used systems, we present a multiagent system (MAS-TD) with agents that represent real world equipment items. In addition, the agents can react to the dynamics of the mine environment and generate new schedules when equipment items encounter failures.

To evaluate the proposed MAS, we compare it against a Tabu Search procedure. Our results demonstrate that the MAS-TD is a more appropriate method than the Tabu Search procedure to solve the truck dispatching problem in open-pit mines, because the MAS-TD generates more efficient schedules than the Tabu Search procedure. In addition, when critical events occur in the mine environment, the MAS-TD regenerates its schedules and achieves a higher production than the schedules generated by Tabu Search and regenerated by Disp-ALG.

In our further research, we will consider more scenarios that include other events such as shovel and truck delays. To decrease the time needed by MAS-TD for rescheduling, we will evaluate a partial regeneration of the schedules.

REFERENCES

Adams, K. K. and Bansah, K. K. (2016). Review of Operational Delays in Shovel-Truck System of Surface Mining Operations. In *4th UMaT Biennial International Mining and Mineral Conference*, pages 60–65, Tarkwa, Ghana.

Alarie, S. and Gamache, M. (2002). Overview of Solution Strategies Used in Truck Dispatching Systems for Open Pit Mines. *International Journal of Surface Mining, Reclamation and Environment*, 16(1):59–76.

Bakhtavar, E. and Mahmoudi, H. (2020). Development of a scenario-based robust model for the optimal truck-shovel allocation in open-pit mining. *Computers and Operations Research*, 115(March 2020):104539.

Bellifemine, F., Caire, G., and Greenwood, D. (2007). *Developing Multi-Agent Systems with JADE*. John Wiley & Sons, Chichester, UK.

Chang, Y., Ren, H., and Wang, S. (2015). Modelling and optimizing an open-pit truck scheduling problem. *Discrete Dynamics in Nature and Society*, 2015:745378.

Costa, F. P. D., Souza, M. J. F., and Pinto, L. R. (2005). Um modelo de programação matemática para alocação estática de caminhões visando ao atendimento de metas de produção e qualidade. *Rem: Revista Escola de Minas*, 58(1):77–81.

Glover, F. (1989). Tabu Search - Part I. *Orsa Journal on Computing*, 1(3):190–206.

Icarte, G. and Herzog, O. (2019). A multi-agent system for truck dispatching in an open-pit mine. In Lottermoser, B., editor, *Abstracts of the Second International Conference Mines of the Future 13 & 14 June 2019, Institute of Mineral Resources Engineering, RWTH Aachen University*, Aachen, Germany. Verlag Mainz.

Icarte, G., Rivero, E., and Herzog, O. (2020). An Agent-based System for Truck Dispatching in Open-pit Mines. In Rocha, A., Steels, L., and van den Herik, J., editors, *Proceedings of the 12th International Conference on Agents and Artificial Intelligence - Volume 1: ICAART*, pages 73–81, Valleta, Malta. SciTePress.

Koç, Ç., Bekta, T., Jabali, O., and Laporte, G. (2016). Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research*, 249(1):1—-21.

Krzyzanowska, J. (2007). The impact of mixed fleet hauling on mining operations at Venetia mine. *Journal of the Southern African Institute of Mining and Metallurgy*, 107(4):215–224.

Liao, T. W., Chang, P. C., Kuo, R. J., and Liao, C. J. (2014). A comparison of five hybrid metaheuristic algorithms for unrelated parallel-machine scheduling and inbound trucks sequencing in multi-door cross docking systems. *Applied Soft Computing Journal*, 21(2014):180–193.

Newman, A. M., Rubio, E., Caro, R., Weintraub, A., and Eurek, K. (2010). A review of operations research in mine planning. *Interfaces*, 40(3):222–245.

Ozdemir, B. and Kumral, M. (2018). Appraising production targets through agent-based Petri net simulation of material handling systems in open pit mines. *Simulation Modelling Practice and Theory*, 87(2018):138–154.

Patterson, S. R., Kozan, E., and Hyland, P. (2017). Energy efficient scheduling of open-pit coal mine trucks. *European Journal of Operational Research*, 262(2):759–770.

Warden, T., Porzel, R., Gehrke, J. D., Herzog, O., Langer, H., and Malaka, R. (2010). Towards Ontology-Based Multiagent Simulations : Plasma Approach. In Bargiela, A., Azam Ali, S., Crowley, D., and Kerckhoffs, E. J., editors, *Proceedings of the 24th European Conference on Modelling and Simulation, ECMS 2010*, pages 50–56, Kuala Lumpur, Malaysia.

Xu, T., Shi, F., and Liu, W. (2019). Research on Open-pit Mine Vehicle Scheduling Problem with Approximate Dynamic Programming. In *2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, pages 571–577, Taipei, Taiwan. IEEE.