Author(s): Dahal, K.P., Aldridge, C.J., McDonald, J.R. and Burt, G.M.

Conference Paper: A GA-based technique for the scheduling of storage tanks

Conference: Congress on Evolutionary Computation (CEC). Washington, DC, USA. 6-9 July 1999.

Publication year: 1999

Publication title: Proceedings of the Congress on Evolutionary Computation

ISBN: 0-7803-5536-9

Publisher: IEEE

Original online publication is available at:
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=785548&isnumber=16970

# A GA-based technique for the scheduling of storage tanks

**K.P. Dahal** k.p.dahal@strath.ac.uk **C.J. Aldridge** c.j.aldridge@strath.ac.uk **J.R. McDonald** j.mcdonald@eee.strath.ac.uk **G.M. Burt** g.burt@eee.strath.ac.uk

Rolls-Royce University Technology Centre, University of Strathclyde
204 George Street, Glasgow, G1 1XW, UK.

**Abstract- This paper proposes the application of a genetic algorithm based methodology for the scheduling of storage tanks. The proposed approach is an integration of GA and heuristic rule-based techniques, which decomposes the complex mixed integer optimisation problem into integer and real number sub-problems. The GA string considers the integer problem, and the heuristic approach solves the real number problems within the GA framework. The algorithm is demonstrated for a test problem related to a water treatment facility at a port, and has been found to give a significantly better schedule than those generated using a heuristic-based approach.**

## 1 Introduction

### 1.1 Problem domain

A real-life problem involving the short-term scheduling of the filling and emptying of tanks in a ballast water treatment facility at a port is considered. The typical layout of the facility is shown in Figure 1. During a given scheduling horizon, ships with ballast water arrive at a berth to take on a cargo of oil. However, ships berthing at a jetty must discharge their contaminated ballast water before they can take on cargo. If ships cannot discharge ballast water due to some constraints in the facility, they must wait until discharging is possible and the ship operators have the right to charge demurrage costs for the time the vessel waits to deballast.

At each jetty station ballast water can be pumped from the ship through the ballast pipeline to one of a number of receipt tanks. The ballast water is then left in the tanks to settle, thereby allowing the oil and water to separate, before the remaining oily-water is run down through further treatment facilities via a run down line. In order to maximise the water quality the running down rate should be at a minimum. In addition to this, the subsequent treatment facility demands continuous and steady flow of the oily-water.

The solution of the problem involves determining the details of the unloading plan for the ships as described later, allocating tanks to store the ballast water of the ships, allocating tanks for running down and determining the running down rates of the facility. This requires minimising delays to ships, maximising water quality by minimising the running down rate and ensuring continuous and steady supply of ballast water to the subsequent process. Furthermore, the solution must satisfy the material balance, physical and operating constraints of the facility. This therefore represents a complex constrained mixed integer combinatorial optimisation problem.

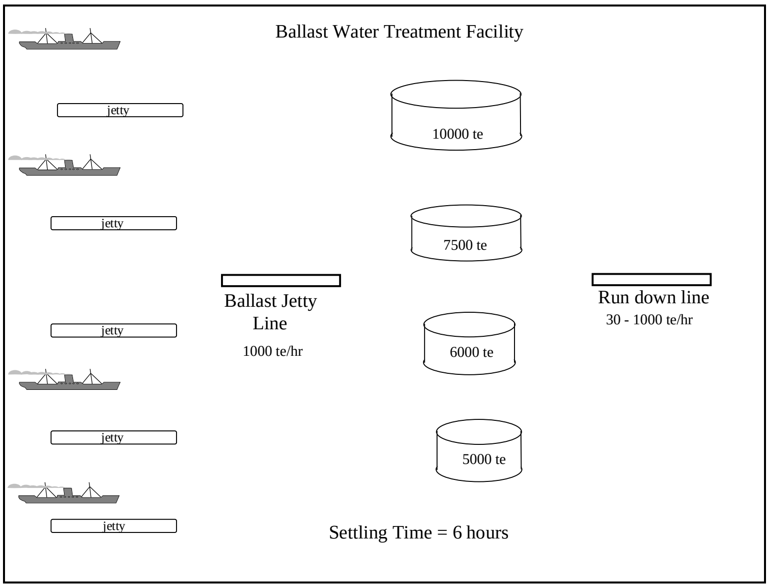


Figure 1: Ballast water treatment facility.

This is a typical type of scheduling problem posed in chemical, oil and water facilities which is vital to solve in order to operate the facility optimally.

### 1.2 Solution techniques

Over the years, there have been many mathematical programming methods and heuristic-based techniques reported in the literature for solving such problems [1,2]. General solution methods include those based on mixed-integer programming, branch-and-bound techniques and dynamic programming. The main drawback of these techniques is that the number of combinations of states which must be searched increases exponentially and becomes computationally prohibitive -known as 'curse of dimensionality'. Furthermore, these techniques are poor in handling the non-linear objective and constraint functions. Hence, several assumptions are generally made to make the problem solvable using reasonable computational resources [3]. The heuristic-based techniques use a trial-and-error method to evaluate the objective function in the time interval under examination. This is troublesome and time consuming as it requires significant operator input [4].

In order to overcome the above limitations, genetic algorithm (GA)-based solution techniques have been more recently implemented for solving complex scheduling problems [5-8]. This paper presents the application of a new GA-based solution approach with a heuristic rule-based component for solving the aforementioned scheduling problem. The technique decomposes the problem into integer and continuous (real number) problem elements. The GA string characterises the integer problem and the real number problems are solved within the GA by using the rule-based heuristic component. The results obtained by using the technique for a test problem based on a realistic scenario are promising and are better than those found using a heuristic approach alone.

The paper is organised as follows. In section 2 the scheduling problem is formulated. Section 3 presents an overview of the proposed approach. The test problem is described in section 4. The performance of the solution technique and the results obtained are discussed in section 5, while conclusions are noted in section 6.

## 2 Optimisation Problem Formulation

Given the configuration of the facility as well as the arrival times and contents of ships and the equipment capacity limitations, the problem becomes one of determining the ship unloading plan and the schedule of filling, settling and emptying tanks as described by the following objective and constraints.

**Objective:** Minimise the sum of costs associated with the waiting time for ships due to the filling constraints, costs involved with excessive run-down rates and costs associated with the non-uniformity of the run-down rates.

**Constraints:** The following are the operating rules and constraints that have to be adhered to in this problem:

1. The “first come, first serve” principle applies for the ship unloading. The first ship to arrive discharges at the highest possible discharge rate.
2. Each ship arrives at and leaves from the jetty station only once throughout the scheduling period.
3. A ship must unload all of its ballast before leaving the jetty station.
4. The ballast jetty line discharge rate cannot be greater than its capacity.
5. Only one tank can be connected to the ballast jetty line at a time.
6. The tank being filled must not be in the running down stage.
7. A tank cannot be filled to more than its capacity.
8. A tank must stand stationary for at least the given settling time after filling up and before running down.
9. Only one tank can be in the running down stage at a time.
10. The instantaneous change in the running down rate must be less than the specified limit.
11. A tank continues to run down until it is emptied.
12. The running down rates must be within the specified range.

## 3 The Proposed Solution Approach

### 3.1 Decomposition of Problem

The proposed GA-based solution technique decomposes the scheduling problem into three sub-problems as shown in Figure 2. Here sub-problems 1 and 3 are continuous (real

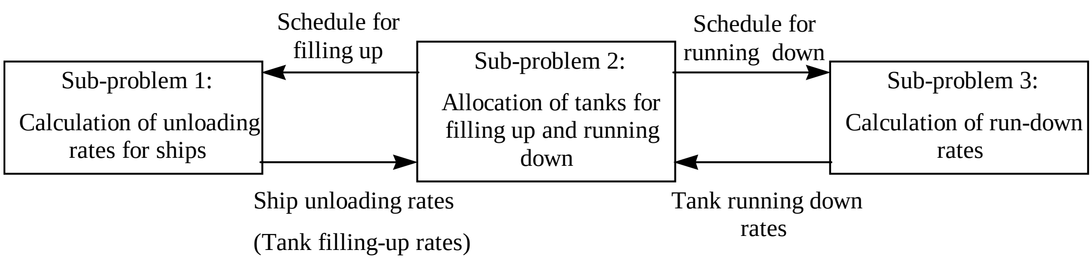


Figure 2: Three sub-problems of the tank scheduling problem.

number) programming problem whereas sub-problem 2 is an integer programming problem.

Given a solution to sub-problem 2 there exists a unique solution to sub-problem 1 which complies with the operating rules represented by constraints (1-4). The results of sub-problem 2 include a schedule for filling up tanks and a schedule for running down tanks. Sub-problem 1 determines ship unloading rates in accordance with the schedule, while sub-problem 3 similarly calculates the tank running down rates using a set of rules and constraints (10-12). The heuristic approach proposed here is based on minimising the running down rates and maximising their uniformity subject to the physical constraints of the facility.

Given the nature of this problem an integrated approach has been developed. The GA solution string adopted here consists of only the decision (integer) variables for sub-problem 2, while sub-problems 1 and 3 are solved within the evaluation function using a rule-based approach. Sub-problem 2 is a combinatorial problem and consequently represents the most difficult among the three sub-problems. This therefore forms a natural target for a GA.

### 3.2 Implementation

The GA string uses integer encoding, which represents the index of the filling-up and running-down tanks for each interval of time. The use of integer encoding instead of binary encoding reduces the size of the GA search space. Each scheduling interval is represented by two integers, one identifying the tank being filled and one the tank being run down. If $iF_t$ and $iR_t$ are the indices of the filling-up and running-down tanks at time t respectively, and T is the number of times in the scheduling period, then the GA string is given by

$$iF_1, iR_1, iF_2, iR_2, \ldots\ldots\ldots\ldots IF_{T-1}, iR_{T-1}, iF_T, iR_{T.}$$

For N number of tanks, $iF_t$ varies from 0 to N, and $iR_t$ varies from 1 to N. This type of representation automatically satisfies constraints (5) and (9) of the problem.

The merit of the schedule represented by the GA string is calculated by an evaluation function as shown in Figure 3. Given the tanks deemed to be filling-up and running-down, a set of rules is used to calculate the ship unloading rates (and therefore the tank filling-up rates) and the run-down rates. The unloading plan for the ships is calculated considering the physical and operational constraints (1-4). The strategy adopted for calculating the running down rates is to implement the maximum possible change as early as possible in the rundown process. This strategy recognises (11) and (12).

The remaining constraints (5-10) of the problem are considered by introducing penalty functions in the evaluation function. These penalty functions take into account not only the fact that constraints are violated but also the degree of those violations by using linear functions. In addition, the evaluation function includes penalty functions for non-preferred operation of tanks. For example, it is preferred to have a tank remain in the filling-up stage until it is full, and to avoid topping up a settled tank.

The evaluation value (indicated in Figure 3) for a GA string is the weighted sum of the objective value, the penalty value for the violation of the constraints and the penalty value for the non-preferred operation of tanks. The weighting coefficients are chosen so that the violation of the constraints generally gives greater penalty values than the objective values and penalty values for the non-preferred operation of tanks. The evaluation value of a string gives an inverse indication of the overall quality of a solution. The lower the evaluation value of a string, the better is its quality. For feasible solutions, the evaluation value is the sum of the objective value and the penalty values for the non-preferred operation (if any).

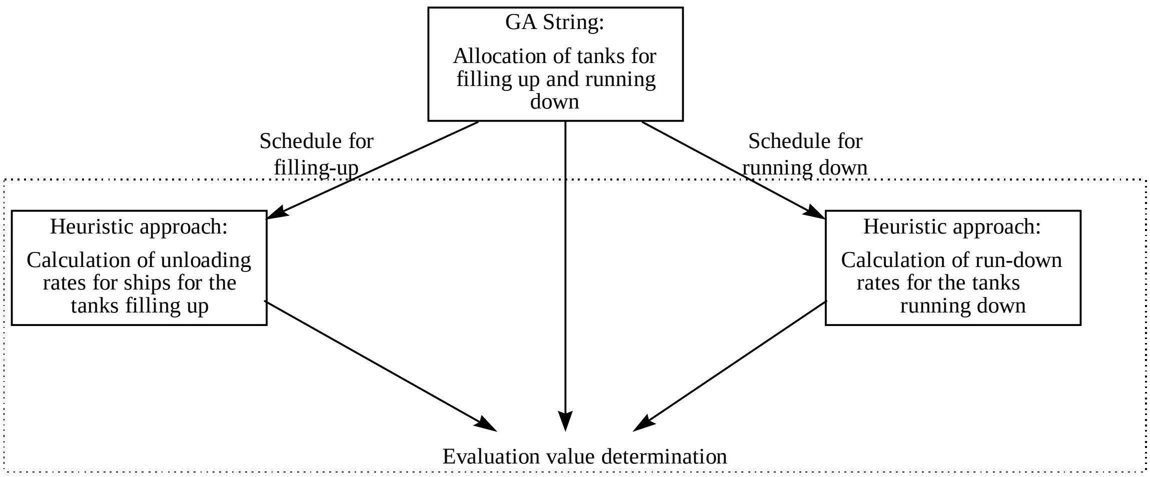


Figure 3: Structure of evaluation function.

### 3.3 GA Operators

The simple GA generates the initial population pool by sampling the search space at random and uses standard crossover and mutation operators for the reproduction process. However, it is also possible to start the GA search process from an initial population which is generated by considering some domain knowledge. For this problem, constraint (6), which describes that a tank filling up cannot be in the running down stage ($iF_t \neq iR_t$), could be respected during the generation of initial solutions. However, if standard crossover and mutation operators are then employed, this constraint may be subsequently violated during the GA reproduction process. Therefore, special n-point crossover and random mutation operators (restricted operators) should be designed which produce new solutions that do not violate constraint (6). The approach presented in this paper uses heuristic knowledge in generating the initial population and the crossover and mutation operators respecting (6) as restricted operators.

## 4 Case Study Description

The proposed solution technique has been applied to a test problem involving four tanks and representing most of the features of a genuine problem. Four ships are to be unloaded during the scheduling period of 24 hours. The maximum flow rate of the ballast jetty line is 1000 te/hr. Here 'te' represents tonne (1000 kg). The tanks must settle for at least 6 hours after the last filling stage before the contents may be run down to the subsequent treatment facility. The running down rate of the facility can range from 30 to 1000 te/hr. The running down rate of the facility at the end of the previous scheduling period is 300 te/hr. The ship and tank details are given in Tables 1 and 2.

| ship | arrival time | initial volume (te) | max unloading rate (te/hr) | waiting cost (units/hr) |
|---|---|---|---|---|
| 1 | 1 | 3000 | 325 | 9.00 |
| 2 | 7 | 8200 | 650 | 24.60 |
| 3 | 13 | 2890 | 650 | 8.67 |
| 4 | 23 | 1400 | 325 | 4.20 |

Table 1: Data for ships.

| tank | capacity (te) | initial volume (te) | initial settled time (hr) |
|---|---|---|---|
| 1 | 10000 | 4650 | 0 |
| 2 | 7500 | 3600 | 6 |
| 3 | 6000 | 6000 | 4 |
| 4 | 5000 | 4800 | 2 |

Table 2: Data for tanks.

The following factors were used in the formulation of the objective function of the problem:

Cost per unit excess run down rate = 0.0035 units/te/hr
Cost associated with non-uniformity of run-down rate= 0.015 units/te/hr

The situation described for this case study represents a tight scheduling situation for the facility, when the available free space in the tanks is smaller than the ballast volume receipts.

## 5 Test Results and Discussion

### 5.1 GA Performance Analysis

In order to identify the best performing GA structure, operators and parameters, comparisons were made of the performances of GAs using steady state and generational population updating approaches, using standard and restricted operators, and varying key GA parameters. The GA parameters which were varied include the crossover probability, mutation probability, the number of crossover points and population size.

The GAs were implemented using the RPL2 program [6] and run on a Sun Sparcstation 1000. A total of ten GA runs have been performed for each case. The total number of trials (iterations) for each run is fixed to 112,500 which is defined by analysis of convergence of the GA technique after a number of experiments. The standard tournament selection method has been applied to choose parents from the population pool for genetic manipulation. The elitism operator has been applied in all cases.

The results obtained using generational (GN) and steady state (SS) GAs with standard two-point crossover and mutation operators are presented in Tables 3 and 4 respectively. The tables show the number of GA runs out of a total of ten runs done for each case which found feasible solutions to the test problem. The crossover probability (CP) and mutation probability (MP) were varied in the range of 0.2-1.0 and 0.005-0.1 respectively, while the population size (PS) was fixed to 150. It can be seen from Tables 3 and 4 that both the GN and SS GAs are sensitive to the variation of the operator probabilities. The performance of the SS GA is also shown to be better than that of the GN GA in terms of the reliability of finding feasible solutions to the problem. As a result only the SS GA was considered for further experiments.

Table 5 shows the results obtained using the SS GA with the restricted operators for population initialisation, two-point crossover and mutation operators. The table shows the number of GA runs out of ten for which feasible solutions were found for the same variation in CP, MP and PS as in the previous table. Comparing Table 5 with the SS GA results presented in Table 4, it is clear that the GA with the restricted operators generally gives a better performance than the GA with the standard operators.

In order to observe the effect of the number of crossover points on the performance of the GA, tests were done with the restricted one-point, three-point and four-point crossover operators with CP and MP in the range of 0.2-1.0 and 0.01-0.1. On the basis of the number of GA runs that found a feasible solution, the GA with the one-point crossover operator was found to give a better performance. Seven out of ten GA runs with this operator when CP=0.2 and MP=0.1 found feasible solutions to the test problem.

| MP<br>CP | 0.005 | 0.01 | 0.05 | 0.1 |
|---|---|---|---|---|
| 0.2 | 0 | 0 | 1 | 1 |
| 0.4 | 0 | 1 | 2 | 2 |
| 0.6 | 0 | 0 | 2 | 0 |
| 0.8 | 0 | 0 | 0 | 0 |
| 1.0 | 0 | 1 | 1 | 1 |

Table 3 : Number of GN GA runs out of ten which found feasible solutions with standard operators.

| MP<br>CP | 0.005 | 0.01 | 0.05 | 0.1 |
|---|---|---|---|---|
| 0.2 | 2 | 0 | 1 | 1 |
| 0.4 | 1 | 1 | 3 | 2 |
| 0.6 | 1 | 1 | 2 | 3 |
| 0.8 | 1 | 0 | 2 | 2 |
| 1.0 | 0 | 0 | 3 | 2 |

Table 4 : Number of SS GA runs out of ten which found feasible solutions with standard operators.

| MP<br>CP | 0.005 | 0.01 | 0.05 | 0.1 |
|---|---|---|---|---|
| 0.2 | 0 | 1 | 4 | 2 |
| 0.4 | 1 | 2 | 3 | 1 |
| 0.6 | 1 | 3 | 1 | 4 |
| 0.8 | 1 | 3 | 0 | 3 |
| 1.0 | 1 | 1 | 4 | 3 |

Table 5 : Number of SS GA runs out of ten which found feasible solutions with restricted operators.

Finally the GA was applied to the test problem with varied population sizes in the range of 50 to 200 using the one-point crossover with CP=0.2 and MP=0.1. The results shows that PS=150 gives the best results in terms of the reliability of finding feasible solutions.

Table 6 summarises the GA design identified from the above experimentation which gives the largest number of GA runs (out of ten) that found a feasible solution to the test problem.

| | |
|---|---|
| GA updating approach | Steady state |
| Parent selection method | Tournament |
| Operators | Restricted (one -point crossover) |
| Crossover probability | 0.2 |
| Mutation probability | 0.1 |
| Population size | 150 |
| # runs out of ten that found feasible solutions | 7 |
| Average (over ten runs) evaluation value of best solution found in each run | 262.87 |
| Evaluation value of the best solution | 69.43 |
| Objective value of the best solution | 67.68 |
| Computational time for one run on Sunsparc workstation | 250 s |

Table 6 : Summary of the best GA performance.

### 5.2 Schedules from GA and Heuristic Approaches

Figure 4 shows the unloading plan for ships given by the best GA solution. The plan for the operation of tanks is depicted in Figure 5.

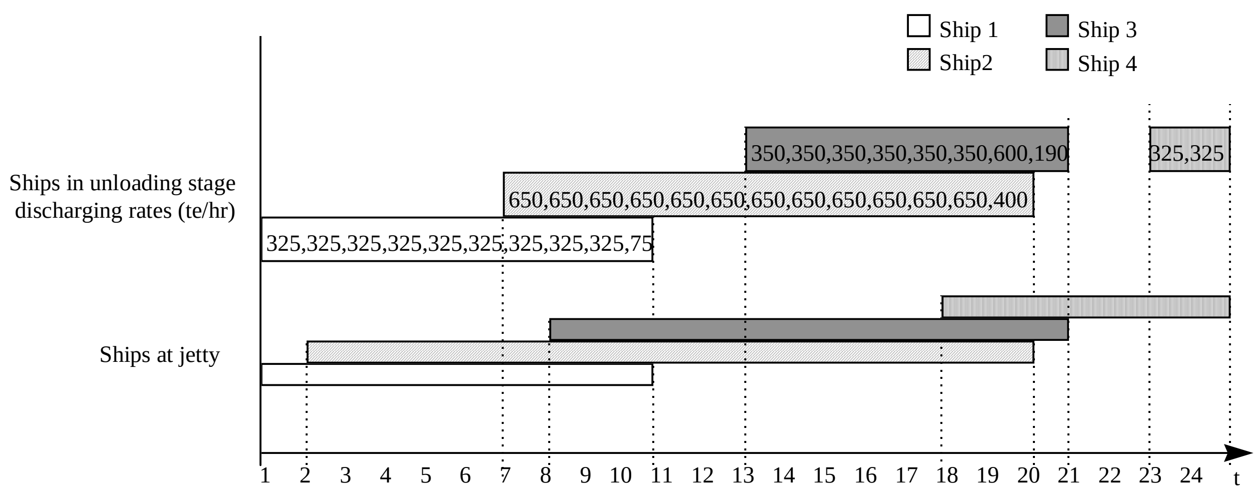


Figure 4: Ship unloading plan given by GA solution and heuristic schedule.

The numerical values for the ship unloading rates, tank filling-up rates and tank running-down rates are also shown in these figures. The GA solution is feasible and has an evaluation value of 69.43.

For the test problem, a heuristic schedule of filling and emptying tanks has been developed using a common operational practice, in order to compare with the schedule given by the GA. The operation of the facility is based on heuristic rules which mainly focus on unloading ships as early as possible and maintaining a constant run down rate of the ballast water. If there is a choice in selecting a tank for filling up, the smallest tank which will take the complete contents of a ship (if possible) is chosen, leaving larger tanks free to receive the next ship-load of ballast. For running down, the heuristic selects the smallest tank settled, in order to have an empty tank available earlier. The strategy for changing the run-down rates is applied as described in the previous section. Figure 4 shows the unloading plan for ships given by the heuristic schedule, while the plan for the operation of tanks is depicted in Figure 6. The heuristic solution is feasible and has an evaluation value of 73.94.

The best GA solution for the test problem has the same ship unloading plan as the heuristic schedule shown in Figure 4. This is not altogether surprising, as the 'first come, first served' principle has been embedded into the GA, and the heuristic and best GA solutions both give the shortest waiting times for the ships, and hence the lowest possible demurrage costs.

The plan for the operation of tanks in the GA schedule (Figure 5) can be seen to differ slightly from the heuristic plan (Figure 6). The selection and time allocation for filling-up tanks in both schedules are the same. For running down the heuristic schedule allocates the first 9 time intervals for

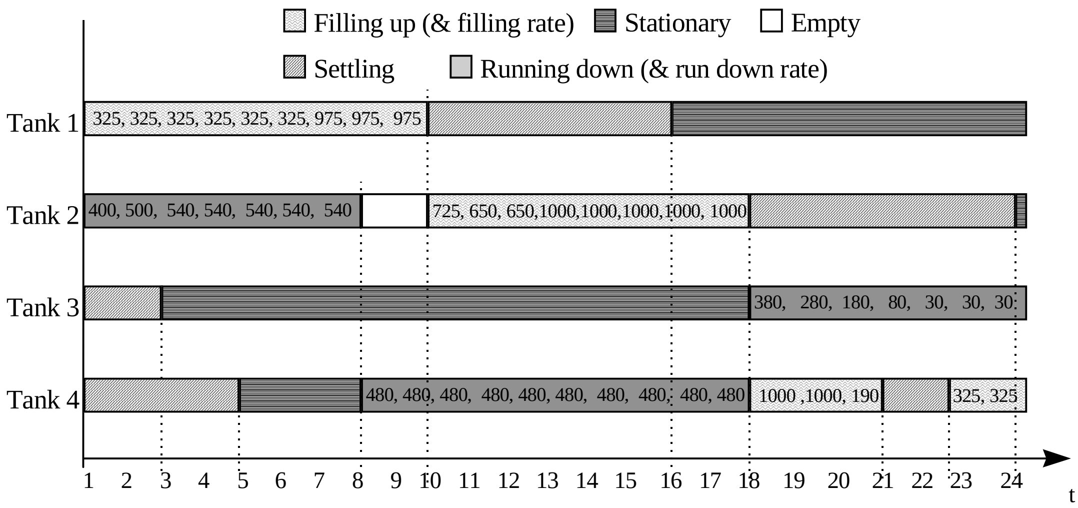


Figure 5: The operation of tanks for the best GA solution.

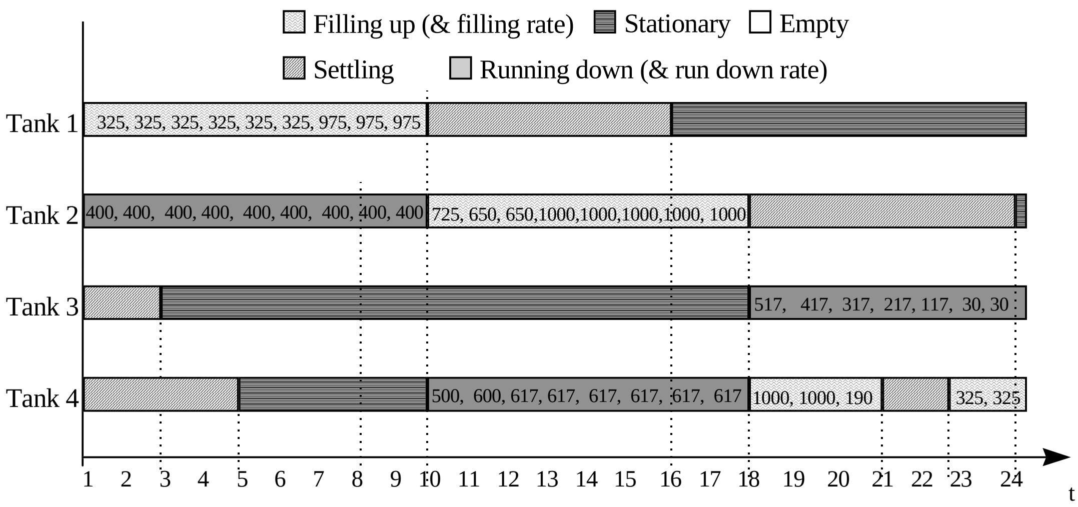


Figure 6 : The operation of tanks for the heuristic schedule.

| Schedules | Costs associated with | | | Total costs (Obj) | Evaluation values |
|---|---|---|---|---|---|
| | ships waiting times | run-down rates | non-uniformity of run-down rates | | |
| Heuristic | 26.01 | 32.63 | 13.55 | 72.20 | 73.94 |
| GA | 26.01 | 30.42 | 11.25 | 67.68 | 69.43 |

Table 7: Comparison of costs of the heuristic and the best GA schedules for the test problem.

tank 2 and the next 8 time intervals for tank 4, whereas the GA schedule allocates the first 7 intervals for tank 2 and the next 10 time intervals for tank 4. As the volume of contents to be run down in tank 4 (4800 te) is greater than that in tank 2 (3600 te), the allocation of more time for running down the larger volume makes the GA schedule better than the heuristic schedule. This improvement is demonstrated by the fact that the highest run-down rate in the heuristic schedule is 617 te/hr in comparison with 540 te/hr in the GA solution.

The quality of the heuristic solution and the best GA solution for the test problem is quantified in Table 7. The numerical values in this table show that the schedule found by the GA-based approach is better than the heuristic schedule developed using the current operational practice for the test problem. As described earlier, the case study represents a tight scheduling situation for the facility, in that there is not much choice of tank selection for filling up and running down at the start of the scheduling period. The search space for the test problem is very large, while the feasible solution space is very small. In such a tight situation, it is a significant achievement of the GA to obtain a better schedule.

Unlike other solution techniques, the GA-based technique works with a population of solutions and offers a set of solutions instead of a single final solution. This highlights an advantage in this application. For example, should the operating conditions change such that the best solution is no longer appropriate, the GA-based technique provides alternatives in its final set of solutions which may be selected instead.

## 6 Conclusions

A realistic test problem for scheduling ship unloading and tank filling and emptying for a ballast water treatment facility has been presented, and the application of a GA-based technique to solve the problem demonstrated. The solution technique uses an ‘integrated’ approach, in which the GA string represents the allocation of tanks for filling up and for running down, and a rule-based approach is used to calculate the ship unloading rates and the tank run-down rates within the evaluation function. The GA string has been encoded using integers. Penalty functions have been employed to incorporate the objectives, constraints and operation of the facility in the evaluation function.

The sensitivity of the GA method to different population updating approaches, operators and parameters has been established. Restricted GA operators have been adopted which always respect an essential problem constraint for the generation of initial solutions and the reproduction process. The GA with these restricted operators has been found to be robust. Tests have shown that high quality solutions can be found if an appropriate specification of the GA is selected for the problem.

Furthermore, comparisons have been made between the GA-based approach and a heuristic method based on current operational practice. It has been shown that the GA-based approach finds a better schedule, with a lower and more uniform run-down rate in a reasonable computational time. Although the GA-based approach is not guaranteed to find the global optimal solution, it is a significant achievement to obtain a good solution to a complex problem like that discussed above in a reasonable computational time. The results demonstrate that the GA-based approach forms the basis of an effective scheduling tool.

## Acknowledgements

This work was carried out in the Rolls-Royce University Technology Centre in Power Engineering at the University of Strathclyde in Glasgow. The authors would like to thank Steve Burchell of BP for providing the realistic problem scenario to formulate the test problem. The authors also acknowledge the use of the Reproductive Plan Language, RPL2, produced by Quadstone Limited, in the production of this work.

## References

1. N. Shah, “Mathematical programming techniques for crude oil scheduling”, Computers Chemical Engineering, Vol. 20, Suppl., pp. S1227-1232, 1996.
2. H. Lee, J.M. Pinto, I.E. Grossmann and S. Park, “Mixed-integer linear programming model for refinery short-term scheduling of crude oil unloading with inventory management”, Ind. Eng. Chem. Res. Vol 35, 1996, 1630-1641.
3. R.L. Storch, W. Song and Z.B. Zabinsky, “Scheduling a chemical processing tankline”, in: I.A. Pappas and I.P. Tatsiopoulos (eds.), Advances in production management systems, Elsevier Science Publisher B.V., 1993, 457-468.
4. S. Yang, “Maintenance scheduling of generating units in a power system”, in: X. Wang and J.R. McDonald, Eds., Modern Power System Planning, McGraw-Hill, London, 1994, 247-307.

5. L. Davis, “Hand book of Genetic Algorithms”, Van Nostrand Reinhold, 1991.
6. Reproductive Plan Language, RPL2, user manual, Quadstone Ltd, 1997.
7. K.P. Dahal, C.J. Aldridge and J.R. McDonald, “Generator maintenance scheduling using a genetic algorithm with a fuzzy evaluation function”, Fuzzy Sets and Systems, Vol 102, 1999, 21-29.
8. C.R. Reeves, “Genetic algorithms”, in: C.R. Reeves, Ed., Modern heuristic techniques for combinatorial problems, Blackwell Scientific Publication, London, 1993, 151-196.