Naval Fleet Schedule Optimization Using an Integer Linear Program

Megan Widmer[a], Michèle Fee[b] and François-Alex Bourque

Defence Research and Development Canada, Centre for Operational Research and Analysis,

Keywords: Schedule, Fleet, Integer Linear Program, Optimization.

Abstract: To inform decisions about future fleet planning, a way to model asset availability over time is needed. To accomplish this, a tool was developed that generates optimized fleet schedules from simplified operations and maintenance cycles. By repeating these cycles and offsetting them from asset to asset, it is possible to generate schedules that meet a set of fleet availability requirements. Target schedule characteristics were encoded in an Integer Linear Program (ILP) and solved using the PuLP python package with the COIN-OR branch and cut solver. To evaluate the effectiveness of the approach, fleet schedules for notional asset fleets were generated and compared qualitatively to those made using a genetic algorithm (GA) based tool that is currently in use. The ILP tool was found to produce schedules that met the requirements more consistently than the GA.

1 INTRODUCTION

Understanding when assets in a fleet are available is a fundamental component in future fleet planning. A method that models optimized fleet schedules that conform to a set of requirements is therefore essential. Such schedules are used, for example, to inform fleet size required to meet operational ambitions and high-level fleet crewing and training requirements.

Due to the nature of the operational demand and the logistics of ship maintenance, the fleet schedules are modelled from repeating operations and maintenance cycles (OPCYCLEs), which differ from class to class according to the assets' maintenance requirements and crewing limitations. These cycles track when each asset is available to perform operations and by offsetting the start of each cycle from asset-to-asset, it is possible to generate fleet schedules that meet a set of availability objectives.

The optimization of this kind of periodic or cyclic schedule is common to a range of fields, including medical (Ferrand et al., 2011; Burke et al., 2004) and military applications (Verhoeff et al., 2015; Raffensperger and Schrage, 1997) to track personnel and equipment availability. In these applications, the use of regular, repeating cyclic patterns to track overall availability lends a predictability to the final schedule that is beneficial in the context of operations planning. Optimizing schedules built from these cyclic patterns has been done with a range of approaches, including heuristic methods (Fee et al., 2019) and mathematical optimization (Ferrand et al., 2011; Verhoeff et al., 2015). In particular the use of linear programs is well documented (Ferrand et al., 2011; Verhoeff et al., 2015) since the requirements can often be expressed in terms of linear constraints. Solutions are often formulated as a mix of hard and soft constraints, where hard constraints are used to capture requirements and restrictions, while soft constraints capture preferences. A hard constraint describes a scenario where a solution is not feasible if that constraint cannot be met, such as was used with Deris et al. (1997) with maintenance requirements. A soft constraint uses a slack variable that tracks a penalty when the constraint is not found, but which is helpful in solving schedules with some flexibility of the outcome. Ferrand et al. (2011) for example uses both hard and soft constraints in optimizing physician work schedules, where the hard constraints are the work requirements, and the soft constraints are the physicians work preferences. Integer Linear Program (ILP) was chosen to perform the optimization in this work because heuristic methods do not guarantee identification of a global optimum and can be difficult to tune. The ILP tool uses a combination of hard and soft linear constraints to bind a solution space and an objective function to determine the optimal arrangement of each asset's cycles in relation to each other.

[a] https://orcid.org/0009-0005-1378-8532
[b] https://orcid.org/0000-0002-6971-7689

Widmer, M., Fee, M. and Bourque, F.
Naval Fleet Schedule Optimization Using an Integer Linear Program.
DOI: 10.5220/0012272200003639

In *Proceedings of the 13th International Conference on Operations Research and Enterprise Systems (ICORES 2024)*, pages 47-57
ISBN: 978-989-758-681-1; ISSN: 2184-4372

The ILP tool was built using the PULP Python package (Foundation, 2023). The objective for the tool was to create optimized schedules for single- or multi-class fleets at a steady state of operations. The constraints were developed to be modular in order to be easily adapted to account for a range of different availability requirements, including those that arise from a fleet split across separate home ports, as was demonstrated here. To evaluate the effectiveness of the tool, fleet schedules were generated for asset fleets with notional maintenance cycles. These will be compared qualitatively to those produced using a genetic algorithm-based fleet optimizer (Fee et al., 2019) that is currently in use to support navy planning.

2 METHODOLOGY

To generate fleet schedules, an operations and maintenance cycle, or OPCYCLE, is used as the main building block. These provide a semi-flexible mapping of the asset's readiness states over time. An asset's readiness state is the measure of its availability to perform operations. The schedule for a single asset is made by repeating the OPCYCLE sequence over a span of time. By choosing the offset between each asset schedule within a fleet, it is possible to distribute available and unavailable states across the fleet schedule to meet a set of requirements.

2.1 OPCYCLE and Readiness States

For the purposes of this report, the following readiness states are used:

- Extended Readiness (ER): The asset is in deep maintenance (usually in dry dock), so it is not considered employable and is not normally crewed.
- Restricted Readiness (RR): In this readiness state, the asset is in shallow maintenance.
- Normal Readiness (NR): This is a period when the asset can conduct limited domestic operations.
- High Readiness (HR): This is a period when the asset is in the highest state of readiness and so it can deploy on expeditionary operations and conduct the full spectrum of combat operations;

Each readiness state involves a different level of crew training as well as availability of different equipment on board. Remaining at the highest state of readiness is taxing on both the crew and the asset and so can only be maintained for a specific number of months.

Figure 1 shows an example of a typical OPCYCLE. Each block represents a time period where the asset is at a particular readiness state and this sequence shows the transition from lowest readiness to highest and back again.

Each asset starts in deep maintenance (ER) and then begins its trajectory towards higher readiness states. To start preparing the asset for operations, it undergoes an intermediate RR stage. Once the asset is out of the RR state, it will be at NR by default and may be raised to HR at any time before the next ER or RR period, whichever is first. Once the HR deployment is completed, the asset will either return to NR or, if it is due for maintenance, enter the scheduled ER or RR period.

2.1.1 OPCYCLE Repeats

If the schedule length is longer than the OPCYCLE, the cycle is repeated. For example, for the OPCYCLE described in Figure 1, to generate a schedule of 30 time units long, the OPCYCLE would be repeated twice. If the schedule length is not a multiple of the length of the OPCYCLE, as shown in Figure 2 for a fleet schedule of 20 time steps, partial OPCYCLE repeats are used. Any time steps beyond t=19 are ignored.

2.1.2 Sub-Operational Cycles

Since it is taxing on both the equipment and the crew to maintain an asset at a higher state of readiness for long periods of time, the operational cycles are often broken up into smaller sub-segments, referred to here as sub-OPCYLEs. Each sub-cycle contains a single HR period and is delineated by short, shallow maintenance periods rather than the longer, deep maintenance found at the start of the cycle.

Each sub-OPCYCLE includes a single sequence of RR, NR, and HR readiness states, where the time that the asset spends at NR and/or HR as well as the placement of the HR state may vary. In Figure 3, the two sub-OPCYCLEs are labelled.

2.1.3 Placement of the HR State

Following the RR stage in each sub-OPCYCLE, the asset may enter the HR state or transition to NR. Once at NR, the asset may be raised to HR at any time before the start of the next maintenance period (ER or RR). In Figure 4, the first sub-OPCYCLE of the OPCYCLE from Figure 1 is examined. In Panel (a), the HR state starts after the RR state ends, while in Panel (b), (c) and (d), the start of the HR period is offset by one, two and three time steps respectively.

For example, in Figure 1, the length of the HR offset in the first sub-OPCYCLE is zero (directly after the RR period) and in the second sub-OPCYCLE

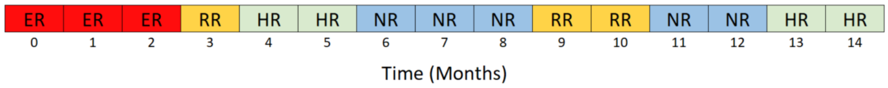

Figure 1: Notional example of an OPCYCLE.

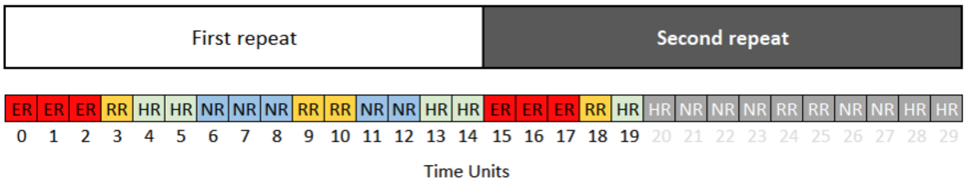

Figure 2: An illustration on how the repeats of an OPCYCLE work using a fleet schedule of 20 time units. The gray readiness states that begin at $t = 20$ show those outside the fleet schedule.

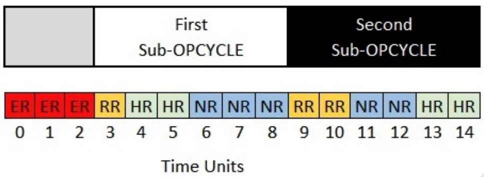

Figure 3: Illustration of sub-OPCYCLEs.

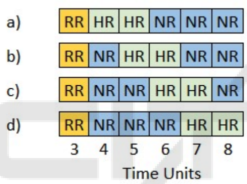

Figure 4: Examples of HR period offset.

is two (two blocks away from the RR period). Breaking the OPCYCLE up in this way allows for the independent placement of the HR period in each sub-OPCYCLE, as seen in Figure 1, which will be an integral part of the subsequent formulation since the flexibility allows readiness requirements to be more easily met.

2.1.4 Wrap Around Feature

For a schedule at steady state, when an OPCYCLE is offset from asset-to-asset, it is necessary for the OPCYCLE to wrap around to the start. For example, in Figure 5, a single asset schedule of 20 time units made from the OPCYCLE in Figure 1 is offset by 5 time units (the ER block begins at 5). The first OPCYCLE runs from time step 5 to 19, and then repeats from time step 20 to 29. The last 5 blocks of that second OPCYCLE wrap around to the beginning of the schedule and appear in time steps 0 through 4 of the schedule (labeled as “second repeat”).

2.2 Fleet Schedule

Combining individual asset schedules, it is possible to construct a fleet schedule, such as that shown in Figure 6. In this representation of a fleet schedule, each row represents a single asset schedule where the coloured tiles represent the readiness states. This figure shows an example of a fleet schedule for a notional fleet of five assets using the OPCYCLE described in Figure 1. To the right of the fleet schedule is a table showing the offset of each asset schedule.

2.2.1 Sub-Fleets

Occasionally it is necessary to track readiness states by grouping of assets, such as those with a common home port or those within a larger fleet belonging to different classes of asset. To accommodate this, the tool includes the idea of sub-fleets. For example using Figure 6, the fleet of five assets can be split between two different sub fleets, Port A and Port B. For the notional example, assets 0-2 are based in Port A and assets 3-4 are based in Port B. So, the fleet schedule now has two sub-fleets that can have their own optimization objectives.

2.3 Characteristics of a Good Schedule

In a naval context, there are many distinct characteristics that make up a good, optimized schedule. They are:

1. A consistent number of assets at ER for all assets at each time unit;
2. A consistent number of assets at HR for all assets at each time unit; and
3. A consistent number of assets at HR for the assets in each sub-fleet.

While the structure length and placement of the ER periods in the OPCYCLE captures the assets’ main-

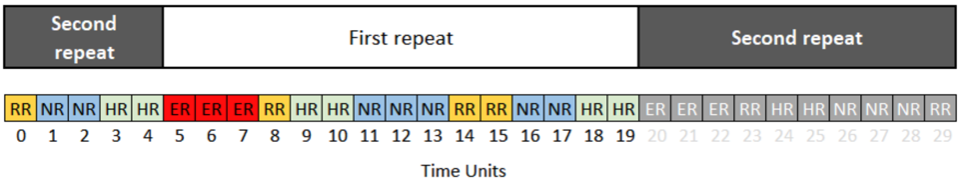

Figure 5: A demonstration of the wrap around feature of an OPCYCLE that has an offset of 5 time units. The gray readiness states that begin at time step 20 show those outside of the fleet schedule.

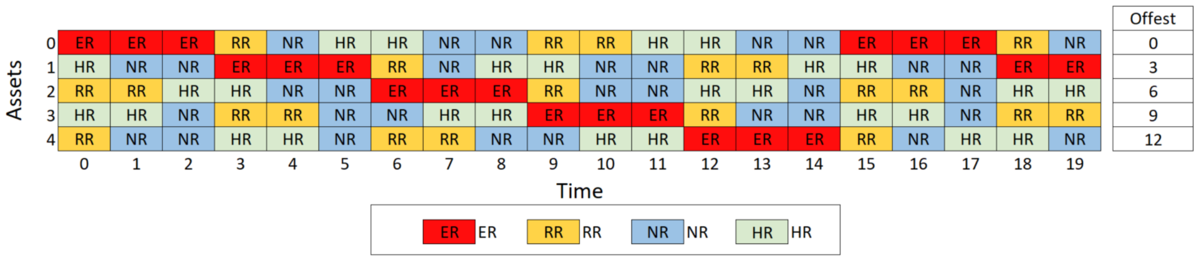

Figure 6: Fleet schedule for five assets spanning 20 time units.

tenance needs, this consistency needed for at the schedule-level stems from the maintenance facility's workload balancing requirements. If there are multiple maintenance facilities for a given fleet of assets, this first characteristic is imposed at the sub-fleet level rather than for all the assets.

The second and third characteristics are derived from the underlying operational demand that the asset fleet is expected to fulfil. With the exception of fleets tied to specific seasonal operations, the bulk of the operational demand is across the fleet as well as at the sub-fleet level is consistent throughout the year. As a result, an attempt is made to maintain the number of assets at $\acute{HR}$ consistent over time in the whole fleet as well as each sub-fleet.

These three characteristics are included in the constraints used in the mathematical formulation of the fleet scheduling problem.

2.4 Integer Linear Program

The following section describes the implementation of the integer linear program (ILP) used to optimize fleet schedules.

2.4.1 Indices

Below are all of the indices used in the formulation of the ILP tool.

- i a sub-fleet
- a an asset
- t a time step
- k an OPCYCLE offset
- h an OPCYCLE repeat
- d a sub-OPCYCLE
- l a HR period offset

2.4.2 Sets

Below are all of the sets used in the formulation of the ILP.

- I the set indexing the sub-fleets
- A the set indexing the whole fleet of assets
- A_i the set indexing the assets in sub-fleet i
- T the set of time steps
- K the set of possible values by which an OPCYCLE is offset relative to the start of the fleet schedule
- H_a the set indexing the OPCYCLE repeats of asset a in the fleet schedule
- $D_{a,h}$ the set indexing the sub-OPCYCLEs of asset a and for each OPCYCLE repeat h
- $L_{a,h,d}$ the set of possible values by which a HR period is offset relative to the RR period of sub-OPCYCLE d for each OPCYCLE repeat h and for each asset a

The fleet schedule from Figure 6 can be used to better interpret the sets and indices defined. Starting with the set T that represents the 20 time steps that make up the schedule length that can be represented as $T = \{0, \ldots, 19\}$.

The five assets in Figure 6 are represented by the set $A = \{0, 1, 2, 3, 4\}$. The set I represents the whole fleet (indicated by zero) and two sub-fleets that correspond to Port A (indicated by one) and Port B (indicated by two), so the set is $I = \{0,1,2\}$. There are three sets for A_i the total fleet $A_0 = A$ for the Port A, $A_1 = \{0,1,2\}$ and for the Port B, $A_2 = \{3,4\}$.

The set K has 20 possible OPCYCLE offset starting points which can be represented as $K = \{0,...,19\}$. Since the OPCYCLE is 15 time units long and the fleet schedule is 20 time units long, there are two repeats of the OPCYCLE represented by the set H_a. The first repeat is full and the second repeat slightly shortened, and depending on the offset the formulation will remove the excess time steps after the offset has been applied. Since the fleet schedule only contains one OPCYCLE and each asset has the same number of repeats, $H_a = \{0,1\}$, $\forall$ a. In this set, zero represents the portion of the fleet schedule shown as the "first repeat" in Figures 2 and 5 and one as the "second repeat" in those figures.

Similarly, from Figure 1, there are two sub-OPCYCLEs that are present and since each asset has the same OPCYCLE, $D_{a,h} = \{0,1\}$, $\forall$ a and h. For this example, the sub-OPCYCLE indicated by zero includes the potions highlighted in Figure 4 ($t = 3$ to $t = 8$ in the OPCYCLE). The $D_{a,h} = 1$ refers to the second sub-OPCYCLE between $t = 9$ to $t = 14$ in Figure 1.

Finally, the set $L_{a,h,d}$ represents the possible relative offset of each HR. Using the OPCYCLE in Figure 1, the set $L_{a,h,d}$, $\forall$ a and h can be determined. The single OPCYCLE contains two sub-OPCYCLEs so there are two sets for both values of d:

$$L_{a,h,d} = \begin{cases} \{0,1,2,3,4\} & \text{if } d = 0 \\ \{0,1,2,3\} & \text{if } d = 1 \end{cases}$$

The positions for the first sub-OPCYCLE ($d = 0$) 1, 2, 3 and 4 are shown in Figure 4. Since the second sub-OPCYCLE ($d = 1$) is shorter, there are fewer permissible values for $L_{a,h,d}$.

2.4.3 Variables

Below are the variables used in the formulation of the ILP.

$X_k^{(a)}$ a binary variable that indicates whether the OPCYCLE starts at the time step $t = k$. For example, in Figure 6, since the OPCYCLE for asset 1 begins at $t = 3$, $X_3^{(0)} = 1$, $X_0^{(0)} = X_1^{(0)} = ...X_{19}^{(0)} = 0$.

$Y_k^{(a,h,d,l)}$ a binary variable that indicates whether the HR period for each sub-OPCYCLE, d, and each repeat of the OPCYCLE in the fleet schedule, h, is shifted by l time steps from the start of the availability block. In Figure 6, for asset 2 (counting from the start of the OPCYCLE at $t = 6$), the visible HR periods are shifted by 2, 1 and 0. As a result, $Y_2^{(2,6,0,0)}$=$Y_1^{(2,6,0,1)}$=$Y_0^{(2,6,1,1)}$=0. The HR block at $h = 1$,$d = 0$ is not shown since it occurs between $t = 20$ and $t = 29$ and is trimmed off.

Z_t a binary slack variable that counts how many time step have soft constraints that are not met over the time period of the fleet schedule.

2.4.4 Parameters

Below are the parameters used in the formulation of the ILP tool.

$ER_{t,k}^{(a)}$ a matrix indicating when asset a is at ER. It returns one if asset a is at ER at time step t given that its OPCYCLE is offset by k, otherwise it returns to zero.

$HR_{t,k}^{(a,h,d,l)}$ a matrix indicating when asset a is at HR. For an OPCYCLE offset by k, it returns one when asset a is at HR at time step t for each HR period in sub-cycle d of OPCYCLE repeat h for each offset of l, otherwise it returns zero.

M the large integer used with the slack variables

$MaxER_i$ the maximum quantity of assets at ER for each sub-fleet i

$MaxHR_i$ the maximum quantity of assets at HR for each sub-fleet i

$MinER_i$ the minimum quantity of assets at ER for each sub-fleet i

$MinHR_i$ the minimum quantity of assets at HR for each sub-fleet i

For the notional example of Figure 1, since the OPCYCLEs are identical for all assets, the matrices $ER_{t,k}^{(a)}$ are the same for all assets. In other words, $ER_{t,k}^{(0)}$=$ER_{t,k}^{(1)}$=...=$ER_{t,k}^{(4)}$, shown in Figure 7.

In this matrix, a value of one corresponds to the presence of an ER block at a particular time step, t, (row) given a particular offset, k (column). Since each asset can only have one such offset (only one value of k), this matrix is used in the constraints described in

$$\begin{bmatrix}
1 & 0 & 0 & \cdots & 1 & 1 \\
1 & 1 & 0 & \cdots & 0 & 1 \\
1 & 1 & 1 & \cdots & 0 & 0 \\
0 & 1 & 1 & \cdots & 0 & 0 \\
0 & 0 & 1 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 1 & 0 \\
0 & 0 & 0 & \cdots & 1 & 1 \\
1 & 0 & 0 & \cdots & 1 & 1 \\
1 & 1 & 0 & \cdots & 0 & 1 \\
1 & 1 & 1 & \cdots & 0 & 0 \\
0 & 1 & 1 & \cdots & 0 & 0 \\
0 & 0 & 1 & \cdots & 0 & 0
\end{bmatrix}$$

Figure 7: Matrix $ER_{t,k}^{(a)}$ for the notional OPCYCLE shown in Figure 1.

subsection 2.4.5 to track the total number of assets in the ER state.

Figure 8 show an example matrix for $HR_{t,k}^{(a,h,d,l)}$. As with the $ER_{t,k}^{(a)}$, these values are OPCYCLE-dependent and so the matrices is the same for all assets in the notional example. The matrix shown below is for the first repeat of the OPCYCLE ($h = 0$) and correspond to the positioning of the HR period in the first repeat ($d = 0$) where it is not shifted from the end of the maintenance period($l = 0$).

As k is increased (in the matrix, moving to the right), the time step at which the HR states occur is shuffled forward in time and, accordingly, down the rows of the matrix.

In Figure 8, the first two matrices represent the first HR period in the OPCYCLE. As l is increased, as for Figure 8 (a) to (b), the position of the ones, are shifted forward in time or downward in the matrix. The matrices in Figure 8 (c) and (d) represent this mapping for the second sub-OPCYCLE and so the ones start further down in the matrix.

The value of M was chosen to be 100 since it much larger than the fleet sizes that are expected to be used with this tool.

In the notional example provided in Figure 1, the maximum number of assets at ER for each sub-fleet, designated by the variable $MaxER_i$, was calculated using Eq. (1).

$$\begin{bmatrix}
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
1 & 0 & \cdots & 0 & 0 \\
1 & 1 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 1 & 0 \\
0 & 0 & \cdots & 1 & 1 \\
0 & 0 & \cdots & 0 & 1
\end{bmatrix}$$

Figure 8: Example matrices for $HR_{t,k}^{(a,0,0,0)}$.

$$MaxER_i = \left\lceil \frac{\text{ER in OPCYCLE}}{\text{Length of OPCYCLE}} \times \text{Number of Assets} \right\rceil \tag{1}$$

To keep the number of assets close to this maximum bound, the minimum bound for the number of assets at ER for each sub-fleet, $MinER_i$, was set to one fewer than the $MaxER_i$.

Using the notional example for a fleet schedule of length $t = 15$ that was used to calculate $MaxER_i$ the maximum number of assets for each sub-fleet can be calculated using Eq. (1).

$$\begin{aligned}
MaxER_0 &= \left\lceil \left(\frac{3}{15} \times 3\right) + \left(\frac{3}{15} \times 2\right) \right\rceil \\
&= \lceil 0.6 + 0.4 \rceil \\
&= 1.0 \\
MaxER_1 &= \left\lceil \left(\frac{3}{15} \times 3\right) \right\rceil \\
&= \lceil 0.6 \rceil \\
&= 1.0 \\
MaxER_2 &= \left\lceil \left(\frac{3}{15} \times 2\right) \right\rceil \\
&= \lceil 0.4 \rceil \\
&= 1.0
\end{aligned} \tag{2}$$

$MaxER_i$ must have a minimum value of one and so the value for both sub-fleets, $MaxER_1$ and $MaxER_2$ is rounded up to one.

A similar calculation, shown in Eq. (3), can be conducted to determine the maximum number of assets at HR for each sub-fleet, designated by the variable $MaxHR_i$.

$$MaxHR_i = \left\lceil \frac{\text{HR in OPCYCLE}}{\text{Length of OPCYCLE}} \times \text{Number of Assets} \right\rceil \quad (3)$$

The minimum number of assets at HR for each sub-fleet, $MinHR_i$, was calculated using the value of $MaxHR_i$,

$$MinHR_i = MaxHR_i - 1 \quad (4)$$

Using the notional example for a fleet schedule of length $t = 15$ with five assets, three in one sub-fleet and two in the other, the maximum number of assets for each sub-fleet can be calculated using Eq. (3).

$$\begin{aligned} MaxHR_0 &= \left\lceil \left(\frac{4}{15} \times 3\right) + \left(\frac{4}{15} \times 2\right) \right\rceil \\ &= \lceil 0.8 + 0.5333 \rceil \\ &= \lceil 1.3333 \rceil \\ &= 2.0 \\ MaxHR_1 &= \left\lceil \left(\frac{4}{15} \times 3\right) \right\rceil \\ &= \lceil 0.8 \rceil \\ &= 1.0 \\ MaxHR_2 &= \left\lceil \left(\frac{4}{15} \times 2\right) \right\rceil \\ &= \lceil 0.5333 \rceil \\ &= 1.0 \end{aligned}$$

The result was one for both sub-fleets, which means that a maximum of one asset for each sub-fleet can be at HR at any time unit.

2.4.5 Constraints

Below are all of the constraints that were used in the formulation of the ILP tool.

First Offset is Zero

The first constraint is used to force the first asset in the fleet schedule to start the ER at the first time unit.

$$X_0^{(0)} = 1 \quad (5)$$

All Assets Must Only Have One Assigned Offset

This constraint is used to tell the model that, for each asset, there can only be one starting point for the OPCYCLE.

$$\sum_{k \in K} X_k^{(a)} = 1 \qquad \forall a \in A \quad (6)$$

From the notional example in Figure 6, for each asset in the schedule a constraint is produced, so there are six constraints. The first one is:

$$X_0^{(0)} + X_1^{(0)} + X_2^{(0)} + X_3^{(0)} + X_4^{(0)} = 1$$

Bounding the Number of Assets at ER

For the sub-fleets defined by A_i, the number of assets in ER is bound between $MaxER_i$ and $MinER_i$.

$$\sum_{a \in A_i} \sum_{k \in K} X_k^{(a)} ER_{t,k}^{(a)} \le MaxER_i \qquad \forall i \in I \text{ and } t \in T \quad (7)$$

$$\sum_{a \in A_i} \sum_{k \in K} X_k^{(a)} ER_{t,k}^{(a)} \ge MinER_i \qquad \forall i \in I \text{ and } t \in T \quad (8)$$

From the notional example in Figure 1, consider the whole fleet $A_0 = \{0, 1, 2, 3, 4\}$. There are a total of 40 constraints. Set $MaxER_0$ and $MinER_0$ to 1 and 0, respectively, the first constraints of Equations (7) and (8) are written as

$$0 \le X_0^{(0)} + X_{13}^{(0)} + X_{14}^{(a)} + \ldots + X_0^{(4)} + X_{13}^{(4)} + X_{14}^{(4)} \le 1$$

Bounding the Number of Assets at HR

For the sub-fleets defined by A_i, the number of assets in HR is bound between $MaxHR_i$ and $MinHR_i$. A slack variable, Z_t, is introduced at each time unit t to soften the constraints yielding:

$$-(MZ_t) + \sum_{a \in A_i} \sum_{k \in K} \sum_{h \in H_a} \sum_{d \in D_{a,h}} \sum_{l \in L_{a,h,d}} Y_k^{(a,h,d,l)} HR_{t,k}^{(a,h,d,l)} \le MaxHR_i \qquad \forall i \in I \text{ and } t \in T \quad (9)$$

$$(MZ_t) + \sum_{a \in A_i} \sum_{k \in K} \sum_{h \in H_a} \sum_{d \in D_{a,h}} \sum_{l \in L_{a,h,d}} Y_k^{(a,h,d,l)} HR_{t,k,h,d,l}^{(a)} \ge MinHR_i \qquad \forall i \in I \text{ and } t \in T \quad (10)$$

The big-M method was selected for these constraints to simplify the problem. While it is possible to exclude the big-M and use the slack variable to track the deviation of the number of assets at HR from the $HRMax_i$ and $HRMin_i$, this requires a Z variable for each constraint rather than each time step. Linking the slack variable across several constraints does run the risk of permitting more constraints to be broken for a given timestep without penalty as soon as one was broken, but this was not found to be the case with the scenarios tested. This is likely because of the way the $HRMax_i$ and $HRMin_i$ range was calculated paired with the flexibility given to the HR placement within the sub-OPCYCLEs. This meant these soft constraints were generally easy to meet.

From the notional example in Figure 1. For the sub-fleet containing all assets, $A_0 = \{0,1,2,3,4\}$, the constraint equations produced a total of 40 constraints. Setting $MaxHR_0$ and $MinHR_0$ to be two and one respectively, the first constraint of Equations (9) and (10), where $t = 0$, are written as:

$$1 \leq Y_1^{(0,1,1,2)} + Y_2^{(0,1,1,1)} + Y_2^{(0,1,1,2)} + Y_3^{(0,1,1,0)} + Y_3^{(0,1,1,1)} + Y_4^{(0,1,1,0)} + Y_7^{(0,1,0,3)} + Y_8^{(0,1,0,2)} + Y_8^{(0,1,0,3)} + Y_9^{(0,1,0,1)} + Y_9^{(0,1,0,2)} + Y_{10}^{(0,1,0,0)} + Y_{10}^{(0,1,0,1)} + Y_{11}^{(0,1,0,0)} + \ldots + Y_1^{(4,1,1,2)} + Y_2^{(4,1,1,1)} + Y_2^{(4,1,1,2)} + Y_3^{(4,1,1,0)} + Y_3^{(4,1,1,1)} + Y_4^{(4,1,1,0)} + Y_7^{(4,1,0,3)} + Y_8^{(4,1,0,2)} + Y_8^{(4,1,0,3)} + Y_9^{(4,1,0,1)} + Y_9^{(4,1,0,2)} + Y_{10}^{(4,1,0,0)} + Y_{10}^{(4,1,0,1)} + Y_{11}^{(4,1,0,0)} - 100Z_0 \leq 2 \quad (11)$$

Link X and Y Variables

To ensure that the solution for $X_k^{(a)}$ and $Y_k^{(a,h,dl,k)}$ correspond to the same k, the variables are linked using the constraint shown in Equation (12).

$$-X_k^{(a)} + \sum_{l \in L} Y_k^{(a,h,d,l)} = 0 \quad \forall a \in A, k \in K, h \in H \text{ and } d \in D \quad (12)$$

This equation ensures that if the solution for $X_k^{(a)}$=0, then the corresponding $Y_k^{(a,h,d,l)}$ do not contain a value of one. As written, this equation also makes sure that, for an $X_k^{(a)}$ with a value of one, only one offset l is selected for each HR period.

From the example in Figure 1, there are a total of 60 constraints that are generated, one for each HR period in the fleet schedule. An example of the first constraint is provided below.

$$-X_0^{(0)} + Y_0^{(0,0,0,0)} + Y_0^{(0,0,0,1)} + Y_0^{(0,0,0,2)} = 0$$

2.4.6 Objective

The objective of the ILP, shown in Equation (13), ensures that the number of assets at HR for the sub-fleets in question stray minimally from the prescribed target ranges.

$$\text{minimize} \frac{\sum_{t \in T} Z_t}{|T|} \quad (13)$$

2.4.7 Existing Genetic Algorithm Tool

A genetic algorithm schedule optimization tool was developed to generate fleet schedules for a navy in transition (Fee et al., 2019). The tool was intended to characterize the capability gap that would occur given the time-lines involved in replacing an older fleet with a newer one. It has since been adapted to handle steady state schedules. The cost function of the GA included the following terms:

- Penalty for the number of times the schedule failed to stay within a target range of number of assets at ER.
- Penalty for how much the total number of assets at HR at each time deviated from a target number.
- Penalty for fluctuations in the total number of assets at ER measured as the sum of the standard deviation in a rolling 24-time-increment window along the length of the schedule.
- Penalty for fluctuations in the total number of assets at HR measured as the sum of the standard deviation in a rolling 24-time-increment window along the length of the schedule.

Given the non-linear nature of the last two terms, the GA was deemed an appropriate tool at the time. While the schedules were deemed to be better than one could achieve by hand, they did not always conform strictly to the fleet availability requirements discussed above. As a result, an attempt to generate linear constraints that could produce a higher quality result was attempted with the current ILP tool.

3 RESULTS

The schedule optimization tools described above were used to create 12-year schedules for a notional fleets with four, six, eight, ten, and 12 assets. These are used as a basis of comparison between the GA tool and the ILP tool to allow for a qualitative comparison between the algorithms used. For all the fleet schedules, the assets are split evenly between Port A and Port B. These make up two sub-fleets in the example, each having a distinct OPCYCLE shown in Figure 9. The two OPCYCLEs were very similar, but since the deep maintenance facilities are located at Port A, the OPCYCLE for assets assigned to Port B has a slightly longer ER period to account for travel to this facility. These 12-year OPCYCLEs have three sub-OPCYCLEs each with one HR period of 12 months each. A time unit of one month was selected.

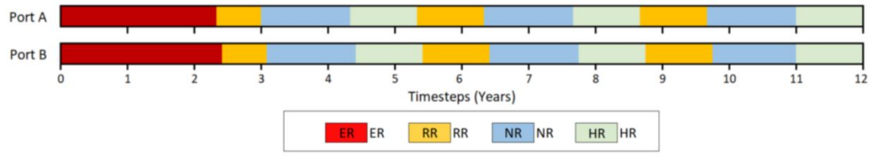

Figure 9: OPYCLEs for assets at Port A and Port B.

3.1 Evaluation of the Fleet Schedule

For the GA and the ILP, fleet schedules were generated for five fleet sizes (four, six, eight, ten and twelve) using the OPCYCLEs from Figure 9. The schedules were compared qualitatively according to their conformity to the characteristics of a good schedule. To facilitate this evaluation, stacked bar graphs were generated that show the number of assets at each readiness state for each month.

The stacked bar graphs are shown for both schedules generated by the GA and ILP tools are shown in Figure 10 to 14.

3.2 Total Assets at ER

Because of the way that the GA implemented the ER availability schedule requirement, it had mixed success with generating schedules where the number of assets did not fluctuate unnecessarily. The GA attempted to maintain the number of assets at ER at a particular target value, effectively $MinER_i$, while limiting variation around this target by penalizing deviations from this value. This method resulted in the number of assets at ER occasionally increasing too high.

For example, in the six- and eight-asset fleet schedules, shown in Figure 11 and 12, the number of assets at ER remained in a range of one assets (between one and two assets) for the whole schedule. In the case of the solutions for the four-, ten- and twelve-asset fleets, found in Figures 10, 13 and 14, respectively, the number of assets fluctuated by three assets. This is most visible in the 12-asset schedule, shown in Figure 14(a). In this case, $MaxER_0$ is three, but the number of assets at ER spiked to four around $t = 20$ and dipped down to one around $t = 120$. These fluctuations in the number of assets at ER places a strain on the maintenance facility, requiring them to employ and discharge staff to adapt to rapid changes in demand.

By changing the way this is constrained in the ILP, these fluctuations were minimized. For all the schedules produced by the ILP, the total number of assets at ER are strictly bound between $MaxER_0$ and $MinER_0$. For example, in the ten-asset results as shown in Figure 13, for all months, the total number of assets at ER remains at a value of one or two assets.

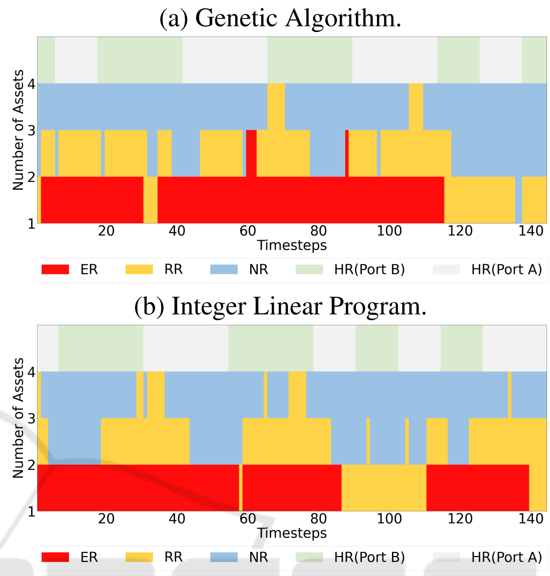

Figure 10: Comparing the schedules of 4.

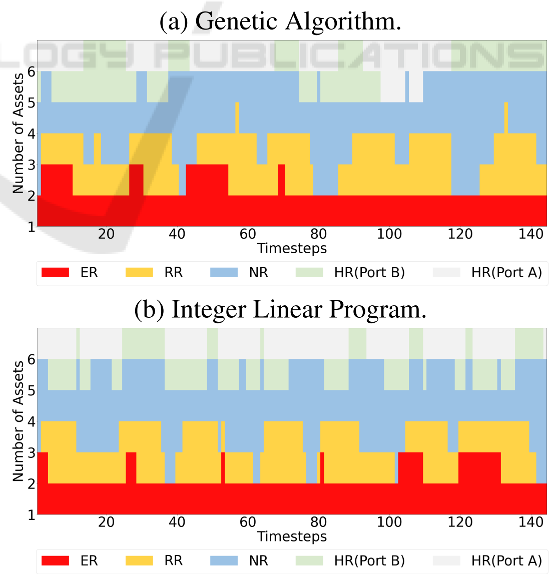

Figure 11: Comparing the schedules of 6.

(a) Genetic Algorithm.

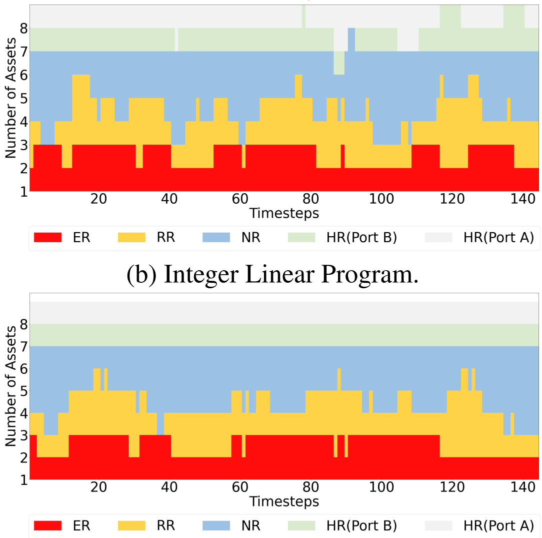

Figure 12: Comparing the schedules of 8.

(a) Genetic Algorithm.

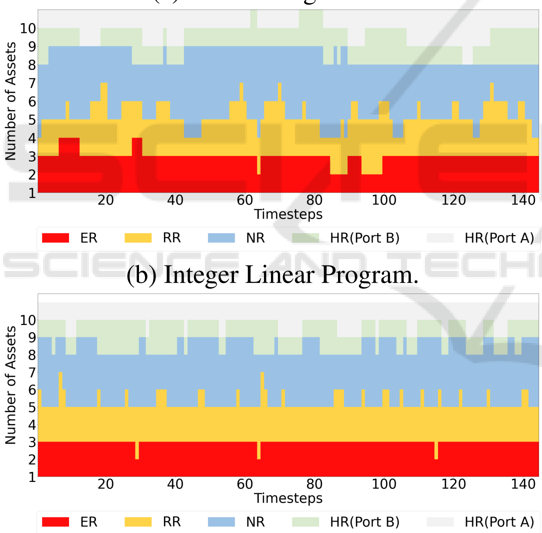

Figure 13: Comparing the schedules of 10.

3.3 Overall Assets at HR

The overall number of assets at HR in the GA generated schedules exhibiting a similar behaviour to the number of assets at ER. The number of assets at HR for most schedules varied by at most one. Exceptionally, the eight-asset fleet in Figure 12 showed a minor inconsistency, where from $t = 89$ to $t = 91$, where there was a spike to three assets, followed by a dip to one from $t = 93$ to $t = 95$. In this instance, it should be possible to fill the deficit with the surplus that was observed.

(a) Genetic Algorithm.

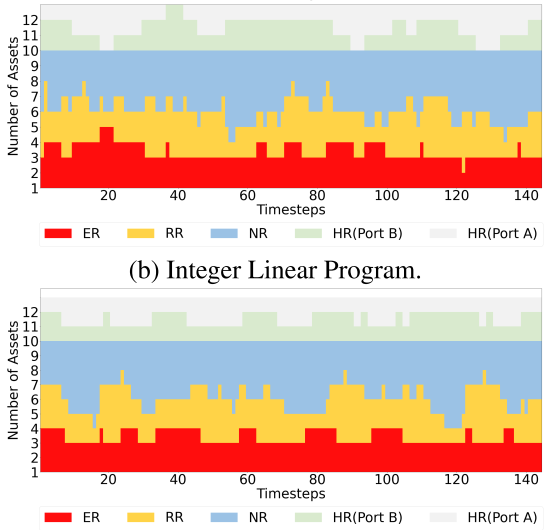

Figure 14: Comparing the schedules of 12.

In the ILP, while the number of assets at HR in the whole fleet is encoded as a soft constraint, restricting the number between a range of acceptable values (between $MinHR_0$ and $MaxHR_0$) for each time increment in the schedule was as successful as it is with the number of assets at ER. For example, it can be seen the ten-asset result as shown in Figure 13 that the total of assets at HR stays between values of two and three for all months.

One side effect of the way in which the GA and ILP handle this schedule restriction is the distribution of the fluctuations observed. In the four-, eight- and twelve-asset fleets, shown in Figure 10, 12 and 14, respectively, it is possible to find a solution where exactly one, two and three assets, respectively, are at HR for the entire schedule. In the six- and ten-asset fleets shown in Figure 11 and 13 where this is not the case, the way in which the number of assets at HR cluster is different.

In the GA tool, the fluctuations are controlled by attempting to reduce the local variability along the schedule by incurring a penalty that is the sum of the standard deviation of a rolling 24-step interval over the length of the schedule.

In doing so, fluctuations in the number of assets at HR are minimized, resulting larger clusters of time when the number of assets are higher or lower. For example, in Figure 13, the intervals where there are three assets at HR are grouped into roughly two clusters, around $t = 30$ to $t = 40$ and $t = 85$ to $t = 144$. In the ILP result for this same fleet size, there are a greater number of smaller groupings.

In the context of a naval fleet schedule, this observation points to an important secondary characteristic of fleet schedules not taken into account in the formulation of both tools. The frequency and duration of changes to availability would have an impact on the operation of a fleet. For the sake of longer term planning, the regularity of availability at different readiness states at a fleet level reduces strain on the supporting infrastructure, such as personnel and maintenance facilities. That being said, there is also likely an optimum duration of any increases in availability that could be leveraged. This optimum would be dependent on the type of operations that the fleet would undertake. For example, having the equivalent of one additional asset frequently available at HR for one week at a time may not provide a measurable benefit, while having a surge of one less often, but for 4-8 months may provide an advantage.

3.4 Assets at HR by Home Port

Since the GA did not aim to control the number of assets at either port in any way, it performed poorly in this respect. Analysing each port separately revealed schedules where the assets at HR on a coast fell to zero at times. In the smaller fleets, specifically the four- and six-asset results, there were large periods of time where only one coast has assets at HR. This was inevitable since the $MaxHR_1$ and $MaxHR_2$ were less than one for these fleets. While this is unavoidable in the smaller fleets, this dip to zero assets was found to occur in the larger fleets as well. For example, in the 12-asset schedule (Figure 14) around $t = 130$, there is a period of time where only the Port A has assets at HR. The $MaxHR_1$ and $MaxHR_2$ are both two, indicating that it should be feasible to maintain at least one asset per home port at all times.

With the ILP, this restriction was added in. As a result, the number of assets at HR in each home port is allocated as expected: these stayed between the ranges of $MaxHR_1$ and $MinHR_1$ (Port A) and $MaxHR_2$ and $MinHR_2$ (Port B) at any given month. Two good examples of this are the six- and eight-asset fleet results. The eight-asset fleet schedule has one asset per coast consistently, while for the six-asset fleet it was always between a range of zero and one which is consistent with $MaxHR_1$ and $MaxHR_2$ values.

4 CONCLUSION

A naval schedule optimization tool was developed using an ILP and tested against an existing schedule optimizer that uses a GA. Both tools used simplified operations and maintenance cycles to generate steady-state schedules that conformed to a set of availability requirements. The tools were compared using 12-year schedules generated for a notional fleets of assets of varying sizes, assigned to two separate home ports.

The ILP tool was formulated with a lot of flexibility in mind. It was devised to allow the user to generate schedules of varying lengths for multi- or single-class fleets with different sub-fleet configurations and OPCYCLEs.

When compared qualitatively, the schedules produced with the ILP conformed to the availability requirements for all fleet sizes more than the GA did. In particular, the results generated with the GA occasionally struggled to minimize fluctuations in the number of assets at HR, both at a fleet-wide level and by sub-fleet. The GA also did not have constraints to regulate the number of assets per port and so often produced schedules where there were no assets on one coast, but plenty on the other port.

Since the tool was flexibly constructed using an off the shelf python package that allows the user to intuitively alter the constraints, this leaves room for future modifications. In particular, the introduction of constraints that regulate the regularity of the fluctuations in the schedule may be worthwhile.

REFERENCES

Burke, E., Causmaecker, P., Berghe, G. V., and Landeghem, H. V. (2004). The state of the art of nurse rostering. *Journal of Scheduling*, 7:441–499.

Fee, M., Caron, J.-D., and Fong, V. (2019). Genetic algorithm for optimization of the replacement schedules for major surface combatants. *Theory and Practice of Natural Computing*, pages 161–172.

Ferrand, Y., Magazine, M., Rao, U., and Glass, T. (2011). Building cyclic schedules for emergency department physicians. *Interfaces*, 41:521–533.

Foundation, P. S. (2023). Pulp 2.7.0. https://pypi.org/project/PuLP. Accessed: 2023-07-20.

Raffensperger, J. and Schrage, L. (1997). A new paradigm for measuring military readiness. *Military Operations Research*, 3(5):21–34.

Verhoeff, M., Verhagen, W., and Curran, R. (2015). Maximizing operational readiness in military aviation by optimizing flight and maintenance planning. *Transportation Research Procedia*, 10:941–950.