



Safe manning of merchant ships: an approach and computer tool

Alapetite, Alexandre; Kozin, Igor

Published in:
Maritime Policy and Management

Link to article, DOI:
[10.1080/03088839.2016.1276305](https://doi.org/10.1080/03088839.2016.1276305)

Publication date:
2017

Document Version
Peer reviewed version

[Link back to DTU Orbit](#)

Citation (APA):
Alapetite, A., & Kozin, I. (2017). Safe manning of merchant ships: an approach and computer tool. *Maritime Policy and Management*, 44(3), 323-335. <https://doi.org/10.1080/03088839.2016.1276305>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Safe manning of merchant ships: an approach and computer tool

Alexandre Alapetite, Igor Kozine¹

alexandre@alapetite.fr, igko@dtu.dk

Technical University of Denmark, Department of Management Engineering,
Produktionstorvet 424, DK-2800 Kongens Lyngby, Denmark

Abstract

In the shipping industry, staffing expenses have become a vital competition parameter. In this paper an approach and a software tool are presented to support decisions on the staffing of merchant ships. The tool is implemented in the form of a Web user interface that makes use of discrete-event simulation and allows estimation of the workload and of whether different scenarios are successfully performed taking account of the number of crewmembers, watch schedules, distribution of competencies and others. The software library “SimManning” at the core of the project is provided as open source. The tool is conceived as a support for the maritime authorities, certifying bodies and shipping companies to assess whether a ship is safely manned.

Keywords

Discrete-event simulation; Web-based simulation; Manning; Staffing; Merchant ships

1 Introduction

Safe manning of merchant ships is a hot topic in the international debate. According to international regulations, all ships must be safely manned and the minimum staffing for each ship is described in the Minimum Safe Manning Document (IMO 1999). A ship is only allowed to sail if the minimum staffing is fulfilled. The Document defines safe manning as a function of the number of qualified and experienced seafarers necessary for the safety of the ship, crew, passengers, cargo and property, and for the protection of the environment. The ability of seafarers to meet the requirements is also dependent upon conditions related to training, hours of work and rest, occupational safety, health and hygiene and the proper provision of food.

A ship’s staffing can be determined either by the scale manning principle or by the goal-based principle. The first is determined by regulation, which merely stipulates that manning should occur depending on type and size of ship, while the goal-based manning evaluates the specific conditions of each of the functions that needs to be performed safely without violating any rule or regulation.

At present, the shipping industry is in tough international competition where even small margins count. Operational expenses for fuel, port charges, etc., are the same for all ship owners operating in the same markets. However, staffing expenses are very different from country to country, and they

¹ Corresponding author

are the only larger expenses that are variable. Therefore, these expenses have become a vital competition parameter and the need for optimising the staffing has become very important. For military vessels, there is a similar pressure to rationalise staffing (Tyson 2006).

The goal-based principle has a potential of optimising staffing expenses given the safety level on the ship is properly maintained. To support making decisions based on this principle, a tool is needed that can provide an objective foundation and documentation to assess whether a ship is safely staffed. More information on the need of such a tool and optimising ship staffing is given in Bruzzone et al. (2007), Wetteland et al. (2000), and in Tyson (2006).

On a general note, staffing a ship can be considered an optimisation problem with the objective function “minimum staffing expenses” or “minimum number of crewmembers” and the restrictive condition that the staff provides safe operation of the ship in the long run. This optimisation problem is multidimensional with a large number of options to choose from. Some options to mention are the size of the crew, number of certified qualifications for the crew, automation of certain tasks, moving some tasks on-shore, having a dual-officer on board, changing the duration of the watch shift, etc. Intuitively, the influence of some of the mentioned variables on safe operation is quite obvious, but the quantification of the influence on safety is more complex. Nevertheless, there exist some measurable parameters that are known to influence the safety of operations directly. One of the pivotal parameters is the workload. Excessive workload and lack of rest and, as a consequence, fatigue are major contributors to inadequate safety on board (MacDonald 2006). In recognition of this fact, the minimum hours of rest on board are stipulated in international (EC 1999) and national regulations (DMEBA 2002). Thus, the main objective function of this optimisation problem is the minimum number of rest rules violations that are defined by the respective regulations. On Danish commercial ships, minimum hours of rest on board can be summarised as follows:

- The minimum hours of rest must not be less than 10 hours in any 24-hour period for a seafarer of 18 years or older. Hours of official rest may not be divided into more than two periods, one lasting at least six hours, and the interval between these rest periods must not exceed 14 hours.
- The total number of rest hours in any seven-day period must not be less than 77 hours.
- Seafarers must not work on Sundays and holidays if the work can be postponed.

As long as all needed tasks are performed as prescribed and safely, the fewer violations of rest rules take place on board, the better the staffing is. To solve the problem of optimizing the staff on board, one needs to perform a comprehensive analysis of tasks carried out during all the phases the ship goes through (sea passage, arrival, departure, cargo handling, and anchoring).

Given the high dimensionality of the problem, its dynamic nature, stochasticity of tasks durations and other events as well as the non-linearity of tasks durations dependent on the phases of the ship’s voyage, one can hardly expect to develop a mathematically well-defined optimisation problem that can be solved analytically. Instead we suggest developing a computer simulation model enabling the evaluation in a dynamic way of all the functions of a ship, and the interaction between staff and these functions. The simulated output should clearly demonstrate if the chosen staffing and workload level on board can comply with the international and national regulations.

This paper reports the choices as well as the development and application of a discrete-event simulation model to support safe manning of merchant ships; it explains in some detail the elements

for the modelling of the manning of ships and describes the development of a dedicated simulation engine for this purpose.

2 Project's ambition

The ambitions and definition of the project were largely driven by some end-users, in particular the Danish Maritime Authority. It was desirable to have a computer tool that would be more precise than current estimations made by their experts. It should be usable both by the Danish authorities, and also directly by their customers (i.e. shipping companies) before filing a certification procedure for a ship. More specifically, the model should be driven by data, allowing users to control the granularity (i.e. level of details). Results should mainly provide statistics regarding the violation of rest rules, and task errors, which is the main outcome to support decision-making. Based on this information, users will manually revise the input parameters such as the size and qualifications of the crew until reaching satisfying results. This iterative process should not be automatized. Finally, there is a focus on internationalisation, with sufficient transparency and hopefully acceptance of the method abroad to pave the way for a better and finer consensus on manning regulations.

There are a number of “real-life” factors that the project does not attempt to account for, and which are likely to impact the work quality and workload. In particular, multiple task interruptions are likely to cause errors, additional workload, and stress, just like excessively long tasks can do. It would be possible to extend our modelling to account for such overheads and related physiological aspects, by introducing appropriate cost functions in the simulation for such undesirable scheduling. Furthermore, a number of aspects are left to the person making the modelling of tasks, such as remembering to include some sickness events and other external conditions that are occasionally incapacitating some crewmembers.

3 Related work

One of the common characteristics of an operator's task in human-machine systems is the need to perform a number of concurrent or consecutive activities within a time interval. To be able to predict operators' performance, several engineering and psychological theories and models have been developed. Reviews of models of multiple task processing and related topics can be found in Wickens (1992) and Liu (1997). In general, engineering theories of human behaviour are concerned with describing gross human behaviours rather than the cognitive or psychomotor mechanisms that underlie them Pew and Mavor (1998). Psychological theories and models of multiple task processing are distinct from engineering in that they are more concerned with understanding and representing the mechanisms underlying behaviours. Theories and models of multiple task processing can be viewed as an extension of theories and models of attention. A prominent theory in this regard is the single-channel theory of selective attention, on which many engineering models of human performance have been founded. The models view human performance as a single server queuing problem, or multiple task sequencing problem, in which a set of tasks or diverse sources of information are queued for service from the single-server human information processing system Liu (1997). The behaviour of systems of this kind can either be modelled by queuing theory or simulated by some modelling software that can imitate the behaviour of a queuing system. Task analyses organised by task sequence are the basis for the modelling that is often referred to as task network modelling Laughery, Lebiere, and Archer (2006), Laughery and Corker (1997). Applications of queuing

systems and task network modelling to human performance modelling can be found in DiVita, Morris, and Osga (2005), Kozine (2007), Liu (1997), Mitchell and Cummings (2005), Wickens (1992).

The above approaches to the modelling of human performance are not specific to staffing optimisation though they can be used to predict workload, timely performance of tasks within a time available, personnel utilisation, etc. These outputs of the modelling in turn can be used to support staffing. Unlike the above approaches to staffing based on models of human performance, there exists a group of rule-based methods providing guidelines for manning safety-critical systems in different domains. For example, document [0] describes the extensive instrument for assessing the manning levels in the control rooms of chemical installations and by using it, bottle necks in personnel arrangements can be pro-actively detected and handled. A completely different staffing evaluation methodology is elaborated in the form of a check list for the chemical industry (Reniers et al., 2007).

In some cases a staffing problem can be well stated mathematically as an optimisation problem. For example, in Heimerl and Kolisch (2010) the problem of staffing multiple projects is modelled as a mixed-integer linear program with the objective function of minimum labour costs. As a prerequisite, it is assumed that a project schedule is given and the required number of units of each skill for each project phase is known. As far as the crewing of ships is concerned, we can see that simulation of human performance in discrete-event simulation (DES) computer environments becomes the core of the supportive tool. Most of the applications found in the literature concern war ships (Wetteland et al., 2000, Srivastava, 2005, Tyson, 2006, and Arher, Lewis and Lockett, 1996) while the manning of civil ships with computer simulation tools is certainly in the incipient phase. Perhaps the most comprehensive existing computer tool for optimal crewing of ships is Integrated Simulation Manpower Analysis Tool (ISMAT) though dedicated to US Navy ships (Tyson, 2006).

The review of the literature has shown that there is an established theoretical basis for modelling human performance that in turn can serve as a foundation for optimal staffing of merchant ships. Nevertheless, a deeper analysis of the developed approaches, models and computer tools has shown their conceptual limitations and case-oriented nature that do not allow us to re-use them directly for our purpose.

To overcome this we have developed a new taxonomy for task relations and task parameterisation that are rich enough to take account of all possible task combinations (sequencing, concurrency, interrupting, resuming, etc.) and allowable time slots for their performance taking place on merchant ships.

4 Approach to simulating the manning of merchant ships

The basic process of human action is thought of as a queuing system, the operating characteristics of which are largely determined by two statistical properties, namely the probability distribution of tasks' inter-arrival times and the probability distribution of service times. As mentioned above, models of queuing systems can be developed within the framework of queuing theory. Nevertheless, due to queuing theory's rather limited abilities, which result from a limited number of assumptions, the use of a DES environment is the approach of our choice to model the behaviour of the complex real world systems.

4.1 Specificities of the project

After analysing end-users' needs and related work, a number of characteristics that make the project different from more conventional approaches and models have been identified. *First*, the model must be highly parameterizable (i.e. based as much as possible on customizable parameters): all tasks, crews, and scenarios must be defined by the end-users, and therefore their number and properties greatly vary from one simulation to another. Very little information can thus be pre-programmed in the model. End-users are domain experts, but not at all versed in statistics, simulation, or computer programming. *Second*, the qualifications are different for every crewmember, and each task has unique values for its many parameters. In traditional DES terminology, this translates to the fact that the capabilities of the resources are only partially overlapping, and that entity instances are very different from each other. *Third*, there are only few hard-wired relationships between tasks: most tasks can be executed independently and in a random order, although competing for the same resources (the qualified crewmembers), and controlled by other parameters such as the current navigation phase and the time of day. There is therefore no need to have a complex multi-level task network, as presented in some other studies (Wetteland et al., 2000, and Tyson, 2006). Furthermore, this level of detail would have required a significant amount of additional information, and a complexity (especially for the users during data input) that would have been outside the scope of the project. *Fourth*, the model must be able to query and modify the properties of any task in the system at any time, in order to:

- Interrupt tasks that are running out of the allowed daily time window, or due to the effect of other parameters such as weekends, weather, and change of navigation phase.
- Optimally assign incoming tasks to crewmembers according to their qualifications and workload, even if some of them were previously busy working on another task.
- Re-assign tasks to other qualified crewmembers to balance the workload across them.
- Manage the case of duplicate instances, when a new instance of a task gets active, while the previous one was not yet completed. For some tasks, such duplicates are forbidden.

Fifth, the model must be able to implement tasks requiring several crewmembers (e.g. a safety meeting where several crewmembers must be present at the same time), and parallel tasks (distinct tasks that must be executed by different crewmembers simultaneously, including interruptions and resuming, e.g. "ballast operations" during "cargo handling"). Although not implemented yet, another future functionality will be the possibility for a given crewmember to work on a number of tasks at the same time (multitasking, e.g. correcting sea maps while navigating).

Furthermore, some tasks can be triggered by external conditions that are stochastically predefined in the input data. For example, "low visibility", "piracy danger", and "narrow waters" require extra look-out on the bridge.

4.2 Overview of the approach

A Web-based simulation approach (WBS) was chosen for a number of reasons (Byrne, Heavey, and Byrne, 2010), such as the possibility for several users to share data and work collaboratively on different models, the ease of deployment (only a modern Web browser is needed, on any platform) as well as maintenance, and the controlled access for a commercial solution (stated at FORCE Technology website: <http://www.safecrewing.dk>).

Regarding the simulation itself, the model runs over time imitating a ship passing through different phases such as *sea passage*, *arrival*, *port*, and *departure*. Other phases may be defined by users like *anchoring*, *weather bound*, etc. The sequence of phases and their durations (constant or random) is given in a scenario. For each phase a set of predefined tasks are active. Each task is characterized by a number of parameters defining when a task can be performed, its duration (usually random), its priority, etc. If several crewmembers are qualified, the workload is then taken into account. Some data about workloads on crewmembers are then collected and corresponding statistical measures are assessed. By examining the output, conclusions can be drawn on which crewmembers are overloaded, what tasks can be moved to other crewmembers or possibly on-shore (MacDonald, 2006), what qualifications are lacking, and so on. The intent of the tool is not to automatically determine the best crew, but to assess how well a given crew will handle a scenario by reporting performance criteria, in particular – as stated before – statistics regarding violations of rest rules and task errors, both of which must be near zero.

5 Implementation of the approach

Given the experience and specific needs reported above, it was decided to start a new implementation in-house from scratch – this reasoning is backed-up by e.g. Robinson (2005) – using test-driven development associated to modern programming techniques and tools. Another driving force was the requirement of relying solely on products which licence is affordable and compatible with the chosen Web-based simulation approach (Byrne, Heavey, and Byrne, 2010).

5.1 Web interface

The project primarily runs as a Web portal written in HTML 5 with ASP.NET 4.0, on which end-users can log on to create and manage their ships, as well as run simulations, as seen in Figure 1. It is compatible with any modern Web browser, also on mobile devices (some experience was gained from e.g. Dokas and Alapetite, (2006)).

The computation of the simulation itself can either take place server-side, or client-side in a Microsoft Silverlight plug-in embedded in a Web page (similar to what can be done in Java). The former approach allows graphical animations during the simulation. Thus, our Web-based solution can be classified (Byrne, Heavey, and Byrne, 2010) as an approach using a local or remote simulation, hybrid visualisation, and local animation. A Web-based documentation is also used, thanks to a wiki system (Byrne, Heavey, and Byrne, 2010).

5.2 Logical data structure

Arguably, the most important achievement of the project is its data structure, capable of modelling the problem. The data structure is used behind the scene when the end-user is inputting the data, and then by the simulation engine is running the scenario and producing relevant outputs. The central difficulty was to define the essential concepts and isolate the relevant parameters. On the one hand, the concepts and parameters should be reasonably easy for end-users to grasp, acknowledging that they may be experts in the maritime domain but not necessarily skilled in statistics and informatics. On the other hand, the data structure should contain at least the minimal information needed for running the simulation, as well as being structured in a convenient and

efficient way. It is also essential to facilitate the comparison of the results across simulations and therefore limit the number of parameters.

| ID | Task name | Task type | Date / Repetitions | Daily hours | Duration / Interruptions | Priority | Crew members |
|-----|---------------------------------------|-------------------|----------------------------------------|---------------------------|---------------------------------------------------------------------------------------------------------------------|----------|--------------|
| 037 | Accounts - Crew | !Non-watchkeeping | Frequency: 30 ≤ 30d ≥ 31; offset: auto | TimeWindow: 07:00 - 19:00 | duration: 60 ≤ 90min ≥ 120; task: ContinueOrResumeWithoutError; phase: ResumeOrDropWithError; end: DropWithoutError | 200 | 1 |
| 106 | Accounts - Provision counting etc | !Non-watchkeeping | Frequency: 25 ≤ 30d ≥ 45; offset: auto | TimeWindow: 07:00 - 19:00 | duration: 60 ≤ 90min ≥ 120; task: ContinueOrResumeWithoutError; phase: ResumeOrDropWithError; end: DropWithoutError | 200 | 1 |
| 082 | Anchoring Deck Ratings (drop anchor) | !Non-watchkeeping | RelativeStartFromStartOfPhase: 0 | TimeWindow: 00:00 - 00:00 | duration: 20 ≤ 30min ≥ 60; task: ContinueOrResumeWithoutError; phase: Obligatory; end: DropWithoutError | 400 | 1 |
| 083 | Anchoring Deck Ratings (heave anchor) | !Non-watchkeeping | RelativeStopFromEndOfPhase: 0 | TimeWindow: 00:00 - 00:00 | duration: 20 ≤ 30min ≥ 60; task: ContinueOrResumeWithoutError; phase: Obligatory; end: DropWithoutError | 400 | 1 |

Figure 1: A screenshot of a part of the Web interface, showing the top of the task list for a ship

Subsequently, this data structure is described in the logical order typically followed by end-users when defining a new scenario for a new ship on the Web user interface. The concept of *ship* is at the heart of the data set. A ship can contain an arbitrary amount of *tasks*, *crews*, *crewmembers* (e.g. “The captain”), *phase types* (e.g. “Sea passage”), *phases* (e.g. “Sea passage from Amsterdam to London”), and *scenarios*, as defined below. The information is stored in XML files.

A *ship* has a *name* and a set of tasks. A ship has a collection of all possible tasks that may occur at any point of the scenario and for any crew. Noticeably, the size, type, and age of the ship are not part of the input data, since these aspects are indirectly taken into account when defining the list of tasks.

A *task* has a *name*, and a *type* (e.g. watch-keeping, rest, critical event, external condition) used to prevent some meaningless combination of parameters. It should be noted that almost everything that is happening is internally managed as a task. “Rest” is also a task (and not only the absence of work) in order to ensure that there is proper competition between normal work and rest, and that crewmembers will not work 24 hours a day. This kind of competition between tasks is primarily governed by a *priority* parameter. When two tasks are set to start at the same time for the same crewmember, the task with the higher priority will win, while the other will wait. A task has some essential parameters controlling its *interruption policy*, that is to say what happens when a task is

interrupted by another task, by a phase transition, or at the end of the scenario. In the case of interruptions by other tasks, possible policies include stopping the task, continuing only if other crewmembers can take the task immediately, or resuming later. For phase transitions, values include “obligatory” (a phase cannot complete before the task is finished), “continue” (the task will continue in the very next phase of the scenario if needed and if allowed by the upcoming phase type), and “resume” (the task will resume in the next possible phase of the scenario if needed). There are also different variants controlling in which situations errors will be reported. A task has a number of attributes related to when it can be performed and for how long. It can, for example, be *forbidden* on Sundays. Finally, tasks can have relations with other tasks, such as to ensure that two tasks will always be performed in parallel, or to allow a task to have *slave* tasks that will be active until their *master(s)* complete. A number of interesting but less important task relations have been identified and are kept for later follow-ups of the project, such as the concepts of predecessor, of trigger (to start another task without further dependency), and of interruption.

A crew has a *name* and a list of crewmembers. Each ship can have any number of distinct crews in order to experiment with a different number of crewmembers and qualifications, allowing concepts such as “dual-officers”. A crewmember has a *name* and, for each possible task of the ship, a value telling whether the crewmember is qualified for the task or not, as visible on Figure 2.

| Tasks \ Crew members | 1. Captain Edit | 2. Chief officer Edit | 3. First officer Edit | 4. Second officer Edit | 5. Chief engineer Edit | 6. First engineer Edit | 7. Second engineer Edit |
|---------------------------------------------------------|--------------------|--------------------------|--------------------------|---------------------------|---------------------------|---------------------------|----------------------------|
| 21. Rest Sunday Crew 7 | | | | | | | 100 |
| 22. Navigational watch 00-06 | 50 | 100 | | | | | |
| 23. Navigational watch 06-12 | 100 | 50 | | | | | |
| 24. Navigational watch 12-18 | 50 | 100 | | | | | |
| 25. Navigational watch 18-24 | 100 | 50 | | | | | |
| 26. Captains supervision with navigation and daily work | 100 | | | | | | |
| 27. Check of safety equipment | 50 | 100 | | | | | |
| 28. Passage Planning | 50 | 100 | | | | | |

Figure 2: Excerpt from the definition of a crew

In Figure 2, a range of values from 1 to 100 will influence the choice of which crewmember will take the job (0 when not qualified). However, these values are not currently used to infer anything on work quality or efficiency. The general assumption is that crew members are either qualified or not to perform a task, and if qualified, they should be able to perform the task correctly.

A phase has a *name* (e.g. “Sea passage from Copenhagen to Amsterdam”) and a *phase type* (e.g. “Sea passage”, “Arrival”, “Port”, “Departure”...). A phase has a predefined *duration*, similar to tasks, which is to be understood as the minimal duration. The definitive duration of a task cannot indeed be predicted before running the simulation, as some delayed obligatory tasks may prolong the phase.

Finally, a scenario has a *name* and a list of phases, to which it is possible to append more phases or even existing scenarios.

When running the simulation, a user will have to select a ship, one of the ship's crews, and one of the ship's scenarios. The number of replications (number of times the same scenario is run) can be tuned to balance the quality of the outputs, at the expense of a longer computation time.

5.3 Open source dedicated simulation library

The new software library (DLL) is written in C# .NET 4.0, a popular and modern high-level multi-paradigm general-purpose programming language. The library is also compatible with Mono 2.8+, making it usable on platforms such as Linux and Mac OS X in addition to Microsoft Windows. It is divided into two parts: a generic discrete-event simulation engine (core), and a software module containing the logic specific to the maritime domain. Other modules could be developed for various domains thanks to an API (application programming interface).

The core library without the maritime module is released as open source (<http://simmanning.codeplex.com>) under the name "SimManning" with documentation and examples. Its focus is set on the simulation of staffing in domains that have sequences of work phases, which is not limited to maritime applications. The library supports both deterministic and stochastic models, and the calculations are reproducible, providing that the random number generator is initialised with the same seed. The library is largely relying on object-oriented programming, although some other paradigms such as functional programming are used in many places too (in particular for data filtering).

When two events are set at the exact same time (which is very common), it is needed to resolve time ties (Henriksen, 2009). Thoroughly developed task taxonomy allows comparing the values assigned to different task attributes and giving a higher priority to one of the two in any conceivable situation.

Additional features include a programming interface providing events upon task assignments, phase transitions, various errors (rest rule violation, forbidden task interruption, task errors, etc.).

5.4 Maritime software module

The maritime software module extends the generic simulation library described above (using its API) to implement domain-specific logic for the maritime case (it is not a part of the open source distribution). This includes the calculation and collection of the data needed for the statistics presented below, in particular regarding rest rules and different types of workload.

A central feature is the algorithm used for assigning tasks to crewmembers (dispatcher). If a task A with a lower priority p_a is already occupying some of the resources needed by a new task B of higher priority p_b , task A will be interrupted if and only if $\lfloor p_b / 100 \rfloor > \lfloor p_a / 100 \rfloor$ ⁽²⁾ and if no other resources are available. Then, a score is calculated only for the remaining qualified crewmember with the formula: $\alpha Q + \beta P_1 + \gamma P_2 + \delta W_{24} + \varepsilon W_{168}$, where Q is how much the crewmember is qualified for the task (1 to 100), P_1 is the priority of the task to be assigned (1 to 1000), P_2 the priority of the task on which the crewmember currently works – if any –, W_{24} the number of working hours of the crewmember in the past 24 hours, and W_{168} the number of working hours in the past week.

Some trial and error tests were performed with expert users to empirically estimate the coefficients giving the most natural outputs, currently set to: $\alpha=1$, $\beta=1$, $\gamma=-1$, $\delta=-200$, $\varepsilon=-20$. The reasons for the

² Floor function, defined as $\lfloor x \rfloor = \max\{m \in \mathbb{Z} \mid m \leq x\}$, where \mathbb{Z} is the set of integers. For instance, $\lfloor 34.56 \rfloor = 34$

relatively high value of δ are mainly because a major behaviour to obtain realistic simulations is to avoid having any crewmember working too much, and the fact that naturally, Q , P_1 , P_2 are typically quite high to start with. When one or more crewmembers are assigned to a task, the model calculates the maximal duration of time these crewmembers may work on the task before infringing the rest rules, and – if necessary – generates some events in the future to attempt to offload this task to other qualified crewmembers, if any.

5.5 Output of the model

Defining the types of outputs produced by the simulation that are shown to the end-users was a similar dilemma to that regarding the data structure. Simulations can produce an overwhelming amount of statistics, and more logs than are humanly readable. Therefore, the effort here was to identify the most relevant information needed. Results are provided on the Web interface.

First, there should be some general information to get a sense of how the scenario has run. This is provided by histograms showing the average and the distribution of phase durations, and by other histograms showing the distribution of the work of each crewmember in the different task types (e.g. what is the proportion of rest, watch keeping, etc. for the captain during sea passages).

Then the most valuable information comes, which is the main purpose of this simulation, namely an estimation of the possibility for the crew to perform all tasks while complying with regulations about rest rules. For this matter, the average number of rest rules violations is reported for each crewmember, classified by phase type, and type of rest rule (Figure 3).

Legend: Rest rule 1 (1×work≤14h/24h); Rest rule 2A (1×rest≥6h/24h); Rest rule 2B (2×rest≥10h/24h); Rest rule 3 (work≤77h/1wk);



Figure 3: Statistics about the average number of rest rules violations during the sea passages of a given scenario. All should be at zero.

For instance, a value of 0.2 in Figure 3 would indicate that a given crew-member would hit a given rest rule type violation every fifth occurrence of the given phase type. There is no standard definition to measure the severity of violations, but when a crewmember is seriously overloaded during a period, more than one violation will likely occur, in particular violations of different types. Formally speaking and according to regulations, these values must be at zero. Therefore, additional human resources must be provided, or other qualifications and task distributions experimented until rest rules violations are infinitesimal. This should be repeated with a number of different scenarios.

It only makes sense to report this workload-related information in parallel with some information about the quality of work, and this role is taken by Table 1, summarising the average number of task

failures, for instance because tasks were inopportunately interrupted, or could not be completed on time. For example, Table 1 reports that the daily task #26 fails in average 3.4 times during every transition phase. The crew must be revised until reaching a negligible number of task errors.

| Tasks of a given scenario | Phase type | | |
|---------------------------------------------------------|-------------|------------|------|
| | Sea passage | Transition | Port |
| 24. Navigational watch 12-18 | 0.05 | 0.08 | |
| 26. Captains supervision with navigation and daily work | | 3.4 | |
| 27. Check of safety equipment | | 0.9 | |
| 28. Passage Planning | | | 1.0 |

Table 1: Excerpt of a report on the number of task errors on average for each phase types

In order to get a better understanding of how a scenario is actually executed and thus to be able to do some tuning and debugging, more detailed information is provided in the form of a table listing every task assignment, and a plot of the workload for each crewmember across the whole scenario.

5.6 Interpretation and decision making

Given the outputs provided by iterations of simulations, a crew is deemed to be more appropriate than another one for a given ship and scenario when both the rest rules violations and tasks errors are lower, with a goal of reaching zero. Furthermore, a crew should still be robust across several realistic and relevant scenarios, including some with a high frequency of ports (longer sea passages tend to have a lower workload). Crews with negligible violations and task errors will be deemed valid and it will be up to the shipping companies to determine which one(s) is/are best for them, according to their own strategies, considering among other variables the number of crewmembers and the cost of their training and qualifications.

Furthermore, shipping companies may favour hybrid solutions with different manning strategies depending on the situations, such as sending aboard another officer just before a period known to have a high workload like a series of ports. These hybrid strategies are already common nowadays, and can also be simulated in the software.

6 Validation and illustration

The granularity of ship models is fully defined by the input data provided by end-users. There are indeed virtually no limits in terms of number of tasks, phases, crewmembers, relations, and time precision. It is up to the user to decide at which level of details to stop, depending mainly on the data available, in particular with regard to abnormal situations and emergencies.

In order to ensure that the software implementation matches the expectations of experts in the field, a formal validation was continuously made during the development by automatically comparing the results produced by the simulation engine to the results calculated by hand on 49 unit tests of various complexity, as detailed earlier (test-driven development). In addition, some larger, more comprehensive, and more realistic models were needed to further validate the solution, and to serve as illustration, reference, and documentation. Therefore, a domain expert produced two major datasets for the project after having made a field study – accompanied by a scientist – on-board two commercial ships currently in service:

- *Jytte Bres* is an 89-meter long dry cargo ship built in 1999, with a dwt (dead weight tonnes) of 4748 tons, and typically 7 to 9 crewmembers. The model defined about 100 tasks, with an example of scenario containing 7 ports over an average duration of nearly two months.
- *Torm Ismini* is a 228-meter long tanker ship built in 2004, with a dwt of 74999 tons, and typically 19 to 23 crewmembers. The model defined about 75 tasks, with an example of scenario containing 7 ports over an average duration of one month.

These two ships of different nature supplement each other well to assess the generality of the solution (a passenger ship, which has different work patterns, would be a good candidate for further validation). The results were found satisfactory by experts in the field, although the success is mitigated by the fact that the use of the system is reported as complex by early users of the system, in particular regarding understanding the impact of the input data on the outputs. The number of input parameters was found high, although none of them could really be removed without losing the possibility to implement common real-world situations.

7 Concluding notes

While the developed tool accounts for routine and scheduled tasks on board as well as external stochastic factors that influence the performance (weather conditions, for instance), the occurrence of incidents and accidents and the severity of their negative impact are not part of the tool. According to feedback from experienced seafarers, even “normal” incidents like sickness, food poisoning and small injuries can drastically increase workload on some crew members and become stress factors that both impair safety.

The approach does not also take account of psychological aspects of safe manning. For example, an increase of the watch by two hours from four to six hours would not distinguish how much this adds to the exhaustiveness of the staff and eventually to a chance of committing a safety-related error. At large, the inclusion of human factors (Bruzzone, 2007, and Wetteland, 2000) like performance shaping factors of crew staff (Kozine, 2007), some cognitive modelling, the influence of assigning a task to a crew member with a lower qualification, team work effect, and a richer set of task interactions would provide a better picture to be able to man the ship safer. Some degree of geographical awareness would also be desirable, in order for instance to properly implement the navigation through time zones, as well as some spatial awareness to better simulate the effect of the movement of crewmembers on the ship.

On the user interface side, the system should be made more user friendly, in particular, by providing a library of predefined sets of tasks, and by implementing wizards able to assist in the data input (e.g. with graphical tools such as a world map to define scenarios) as well as data analysis (e.g. identify the root of common problems). Monitoring the events and the assignment decisions in future additional cruises is also a very wishful activity that will certainly provide an invaluable input for improving the model.

8 Acknowledgements

The project was developed in collaboration between the Technical University of Denmark, FORCE Technology (Maritime Division, Denmark), and the Danish Maritime Authority. A major part of the project was financed by the Danish Maritime Fund.

We would like to thank Steen Weber for his participation in the initial phases before retiring from DTU. Several persons from FORCE Technology should be acknowledged as well, particularly, maritime expert Uwe Delfs Jespersen for invaluable contributions, Christian Brandt Lauritzen for playing a major role especially during the first implementation phase, Jens Erik Bay for managing the project, and Captain Carl Thue Rabjerg for his assistance in data collection.

References

- Arher, R.D., Lewis, G.W., and Lockett, J. 1996. "Human Performance Modelling of Reduced Manning Concepts for Navy Ships." *Proc. of the Human Factors and Ergonomics Society*, 987-991.
- Brabazon, P., and Conlin, H. 2001. *Assessing the safety of staffing arrangements for process operations in the chemical and allied industries*. London: Health and Safety Executive.
- Byrne, J., Heavey, C., and Byrne, P.J. 2010. "A review of Web-based simulation and supporting tools." *Simulation Modelling Practice and Theory* 18(3): 253-276. doi:10.1016/j.simpat.2009.09.013
- Bruzzone, A.G., Briano, E., Bocca, E., and Massei, M. 2007. "Evaluation of the impact of different human factor models on industrial and business processes." *Simulation Modelling Practice and Theory* 15 (2): 199-218. doi:10.1016/j.simpat.2006.09.020
- DMEBA. 2002. "Regulation on seafarer's rest (In Danish: "Bekendtgørelse om søfarendes hviletid mv."). Regulation 515". *Danish Ministry of Economic and Business Affairs*. <https://www.retsinformation.dk/Forms/R0710.aspx?id=25940>
- Dokas, I.M., and Alapetite, A. 2006. « A view on the Web engineering nature of Web based expert systems." Paper presented at the 1st International Conference on Software and Data Technologies, Setúbal, Portugal, September 11-14.
- DiVita, J., Morris, R., and Osga, G. 2005. "Modeling Supervisory Control in the Air Defense Warfare Domain with Queuing Theory." Paper presented at the 10th International Command and Control Research and Technology Symposium, Virginia, June 13 - 16.
- EC. 1999. "Agreement on the organisation of working time of seafarers concluded by the European Community Shipowners' Association (ECSA) and the Federation of Transport Workers' Unions in the European Union (FST)". *Council of the European Union. Directive 1999/63/EC*. <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31999L0063:EN:NOT>
- Greenstein, J.S., and Rouse, W.B. 1982. "A model of human decision making in multiple process monitoring situations." *IEEE Transactions on Systems, Man, and Cybernetics* 12 (2): 182-193.
- Henriksen, J.O. 2009. "Efficient Modeling of Delays in Discrete-Event Simulation. Advancing the Frontiers of Simulation." *International Series in Operations Research & Management Science* 133: 105-141. doi:10.1007/b110059_6
- Heimerl, C. and Kolisch, R. 2010. "Scheduling and staffing multiple projects with a multi-skilled workforce." *OR Spectrum* 32: 343-368.
- IMO. 1999. Principles of safe manning. *Resolution A.890(21), 21st session*. <http://www.imo.org/OurWork/HumanElement/VisionPrinciplesGoals/Pages/PrinciplesOnSafeManning.aspx>
- Kozine, I. 2007. "Simulation of human performance in time-pressured scenarios." *Proc. IMechE, Part O: J. Risk and Reliability* 221 (2): 141-151, doi:10.1243/1748006XJRR48
- Liu, Y. 1997. "Queuing Network Modeling of Human Performance of Concurrent Spatial and Verbal Tasks." *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans* 27 (2): 195-207, doi:10.1109/3468.554682.
- Laughery, K.R., and Corker, K. 1997. "Computer Modeling and Simulation." In *Handbook of Human Factors and Ergonomics, 2nd ed.*, edited by G. Salvendy, 1375-1408. New York: John Wiley & Sons.
- Laughery, K.R., Lebiere, C. and Archer, C. 2006. "Modelling Human Performance in Complex Systems." Chapter 36 in *Handbook of Human Factors and Ergonomics, 3rd ed.*, edited by G. Salvendy, New York: John Wiley & Sons.

- MacDonald, R. 2006. "Safe Manning of Ships – Yesterday, Today and Tomorrow." *WMU Journal of Maritime Affairs* 5 (2): 163-151.
- Mitchell, P.J., and Cummings, M.L. 2005. "Management of Multiple Dynamic Human Supervisory Control Tasks." Paper presented at the 10th International Command and Control Research and technology Symposium, Virginia, June 13 - 16, paper 176.
- Pew, R.W., and Mavor, A.S. (eds.). 1998. *Modelling Human and Organizational Behavior: Application to Military Simulations*. National Academy Press.
- Robinson, S. 2005. "Discrete-event simulation: from the pioneers to the present, what next?" *Journal of the Operational Research Society* 56: 619–629, doi:10.1057/palgrave.jors.2601864.
- Reniers, G.L.L., Dullaert, W., Ale, B.J.M., Verschueren, F., and Soudan, K. 2007. "Engineering an Instrument to Evaluate Safety Critical Manning Arrangements in Chemical Industrial Areas." *Journal of Business Chemistry* 4 (2): 60-75.
- Srivastava, N., Pietryka, F., Horne, G., and Theroff, M. 2005. "Simulation environment to assess technology insertion impact and optimized manning." Paper presented at the 37th conference on Winter simulation (WSC '05): 1088-1093.
- Tyson, J.S. 2006. "Manning and Automation Model for Naval Ship Analysis and Optimization. Master's Thesis, Virginia Tech, USA.
- Wickens, C.D. 1992. *Engineering Psychology and Human Performance, second edition*. New York: HarperCollins.
- Walden, R.S. and Rouse, W.B. "A queuing model of pilot decision making in a multitask flight management situation." *IEEE Trans. on Systems, Man, and Cybernetics* 8 (12): 867-874.
- Wetteland, C. R., Miller, J. L., French, J., O'Brien, K., and Spooner, D. J. 2000. "The human simulation: resolving manning issues onboard DD21." Paper presented at the 32nd Conference on Winter Simulation, Orlando, Florida, December 10-13: 1402-1406.