AN OFFSHORE ENERGY SIMULATION THROUGH FLOW NETWORKS: *CEL* WITHIN THE *MSP CHALLENGE 2050* SIMULATION GAME PLATFORM

Kevin Hutchinson, Harald Warmelink, Wilco Boode, Carlos Santos, Igor Mayer
Academy for Digital Entertainment
Breda University of Applied Sciences
Monseigneur Hopmansstraat 1, 4817 JT Breda,
The Netherlands
E-mail: hutchinson.k@buas.nl

KEYWORDS
Offshore energy simulation, maritime spatial planning, flow network, maximum flow problem, Dinic's algorithm.

ABSTRACT

This paper presents the design of the offshore energy simulation *CEL* as a flow network, and its integration in the MSP Challenge 2050 simulation game platform. This platform is designed to aid learning about the key characteristics and complexity of marine or maritime spatial planning (MSP). The addition of CEL to this platform greatly aids MSP authorities in learning about and planning for offshore energy production, a highly topical and big development in human activities at sea. Rather than a standard flow network, CEL incorporates three additions to accommodate for the specificities of energy grids: an additional node for each team's expected energy, a split of each node representing an object into input and output parts to include the node's capacity, and bidirectional edges for all cables to enable more complex energy grid designs. Implemented with Dinic's algorithm it takes less than 30ms for the simulation to run for the average amount of grids included in an MSP Challenge 2050 game session. In this manner CEL enables MSP authorities and their energy stakeholders to use MSP Challenge 2050 for designing and testing more comprehensive offshore energy grids.

INTRODUCTION

MSP Challenge 2050 (henceforth MSP Challenge) is a novel simulation game platform designed to aid learning about the key characteristics and complexity of marine or maritime spatial planning (MSP). MSP is a process conducted by different governments surrounding a sea basin such as the North Sea, ending up in a spatial plan for each country's areas therein, i.e., their territorial waters and exclusive economic zones. More specifically, it is a process by which a country 'analyse[s] and organise[s] human activities in marine areas to achieve ecological, economic and social objectives' (European Union 2014), ending in a spatial plan. This spatial plan is essentially a highly annotated map of the area with spatial designations for specific human activities and marine protection measures for the medium-term future, often a period of 5-10 years. MSP Challenge was first conceived and developed as a computer simulation game in 2011 and has been applied in sessions with MSP authorities, stakeholders and students many times since (Mayer et al. 2014, 2013; Stolte et al. 2013). Since early 2016 it has been further developed at Breda University of Applied Sciences within the context of the EU projects and consortia *NorthSEE, Baltic LINes* and *SIMCelt*. It has now become a platform allowing for all sorts of simulation game sessions: in different sea basins, with different data sources, and with different simulation models running in the background.

A very topical and big development in human activities at sea is offshore energy production. MSP authorities all over the world are highly concerned with finding sites for the development of offshore renewable energy (mostly wind farms) to help achieve economic and sustainability targets set in various (inter)national policy agreements (Kafas et al. 2018; Borrmann et al. 2018; McGowan 2018). For MSP authorities it has therefore become crucial to dive into their country's overall energy production and the intricacies of offshore renewable energy systems development in the sea basin involved.

At present offshore energy production is a highly complex endeavour involving the development, maintenance and possibly upgrade, and future decommissioning of mostly wind farms. The creation of individual wind farms and their onshore connection to an electricity grid is already complicated, as it involves finding suitable sites (e.g. shallow waters and generally high wind speeds) and the selection of suitable technologies (e.g. turbines, pylons, cables, transformers), while these technologies continue to develop (e.g. bigger turbines, new turbine designs and constraints). The complexity arises when taking into consideration the development of large transnational grids of multiple energy systems such as traditional fossil fuel and wind energy, as well as the diverse consequences to other human activities and marine life.

Simulation gaming greatly helps players understand and deal with a complex endeavour such as offshore energy production (Bekebrede, Lo, and Lukosch 2015; van Bilsen, Bekebrede, and Mayer 2010; Mayer 2016). An offshore energy production simulation game would allow MSP authorities to develop and test offshore energy systems with their energy stakeholders in a safe environment. Moreover, given the spatial relevance, it would be very useful if that simulation game could work within the MSP Challenge platform. The added value of the MSP Challenge platform is that it already offers the framework of the MSP process, i.e., the process of collectively developing, reviewing, amending and approving one's plans. It also offers the option of planning all sorts of other human activities, such as shipping infrastructure, and marine protection measures, such as no-shipping or no-fishing zones. Within the MSP Challenge platform, players could thus design offshore energy systems in an integrated manner.

In this paper we therefore answer the question *how an offshore energy simulation could be designed and implemented within the MSP Challenge platform, allowing players in a multiplayer setting to spatially plan and implement offshore energy production in an attempt to reach a predefined energy demand or target*.

We answer this question by explaining the process of designing and implementing the offshore energy simulation *CEL* as a flow network. The entire team, that extends beyond the authors of this paper, went through this process over a period of almost a year. The team consulted with key energy experts within the *NorthSEE* and *Baltic LINes* consortia and networks at different stages along the way. We conclude with the main opportunities and limitations that the current design of CEL introduces.

OFFSHORE ENERGY DESIGN IN MSP CHALLENGE

The MSP Challenge simulation game platform has a multi-player client-server architecture, where the server processes inputs from the connected clients, feeds input into any connected simulations (notably an ecosystem simulation), receives output from these simulations, and feeds back data to the connected clients. All simulations running in the background have a discrete-event architecture. Each discrete event represents one simulated month and ideally takes only about a second to run. The time between each discrete event is defined by the facilitator or game master and depends on how long he/she wants the entire session to take. MSP Challenge simulates the planning process in periods of 10 years each, up to a maximum of 40 years, during which players play in country teams to design and implement MSPs, analyse the outcomes and further consequences, and make new plans accordingly.

The addition of an energy simulation is meant to allow MSP Challenge players to design more comprehensive energy production plans, within the wider MSP context. MSP Challenge players have always been able to designate areas for energy production such as wind farms, and then connect them to shore via an electricity cable. With this next development step, we wanted to have the platform calculate how much energy the players are actually generating, transporting and consuming with their energy production areas.

To simplify this both for the players and the development team, we do not consider the network as a whole but divide it into separate grids. Grids consist of sources that generate power, cables and transformer stations that transport power, and sockets that consume power. Each grid is an independent part of the network, meaning they are not connected and can in no way influence each other. Each grid's energy output is added up and fed back to the players as the total amount of energy created in their network. Like all spatial designations in MSP Challenge, each grid element (e.g. a wind farm or electricity cable) can belong to a different team. This allows teams to not only co-develop a grid, it also forces them to specify how much power each team adds to or receives from the grid.

Determining the energy distribution is part of the grid design process. These energy distributions only specify what different teams expect to get from a grid, not what they actually get. Players may set the expected amount of energy to a large amount, such as the maximum amount that a wind farm can generate. However, if the cables or transformers in the grid cannot handle this, the energy they actually receive will be lower. Determining and placing cables and transformers that allow the right amount of energy to pass through without wasting too much capacity is a big part of the challenge for the player.

Players create these grids in the game client. The grid designs are then sent to and stored on the game server. When the game's time progresses another month, the game needs to know how much energy teams get from their grids. This is where the energy simulation CEL comes in, which is the focus of the remainder of the paper.

CEL'S ARCHITECTURE

In this section we explain the CEL architecture by specifying what we feed into it as input, what we would like to get out of it as output, and why we chose to approach the simulation as a flow network.

Specifying CEL's input: grids

As explained, CEL's input is a set of individual grids drawn by MSP Challenge teams defining the entire energy network. We need to define of what a grid consists more formally to ensure that the flow of energy through them can be simulated, both technically and realistically. Grids consist of four parts:

1. Input: Sources, such as wind farms
2. Output: Sockets
3. Medium: Cables and transformers
4. Distribution: A list of expected outputs per team that is part of the grid.

Each of these elements can only be in a single grid at a time. When a cable is added between two grids, their contents are combined and they become a single grid. We resolve the energy simulation for each grid separately and then combine the results to present the total amount of energy produced and consumed.

Sources, sockets, cables and transformers are all objects in the world. They are limited by a capacity that is specified in the configuration of the game (thus not in CEL, but in the MSP Challenge platform). When creating a grid, the relevant teams need to specify how much energy they expect to get, and thus how the energy is distributed. The energy that teams expect to get from a grid is not simply a fixed value that they purely set themselves. The value is subject to two rules:

1. Each team's expected energy has to be lower than or equal to the sum of the capacity of their sockets in the grid.
2. The sum of all teams' expected energy combined needs to be lower than or equal to the total energy available in the grid.

Note that the medium is not present in these rules. The expected energy specifies what teams would get if the medium was perfect, i.e., has the right capacity to carry all produced energy all the way to the socket. In practice the medium might not accommodate the necessary amount of energy to reach all teams' expected values.

Specifying CEL's output: energy KPIs

Having defined more formally what type of data CEL receives as input, we can now specify CEL's output. To define the output we first consider what data we want to display to the players in the game client. We feed back the following variables to the teams as key performance indicators of the energy grids and overall energy network:

- All objects (cables, transformers, sockets and sources) displaying the amount of energy that passed through them.
- An overview per grid, showing how much energy all teams participating in the grid received from it.
- An overview per team and per game session, showing the amount of energy produced, consumed and shared.

To show this information the minimum output we need from CEL is the amount of energy that passes through all objects in the network. All other data can be derived from this. However, it would be easier in programming terms if the output also includes how much energy each team got per grid.

A flow network

The CEL design is based on the simulation objective: *given a grid, find the maximum amount of energy that can flow from the sources to the sockets, with each team's sockets collectively limited by the output expected by the team as defined in the grid design.*

The first part of this phrasing reveals our design foundation – considering energy grids as flow networks – rendering our problem similar to the maximum flow problem (Ahuja 2017; Schrijver 2002). Indeed, others have also approached energy grids as flow networks (Fang et al. 2018). If we can represent these particular energy grids as flow networks, there are a host of existing algorithms to calculate the maximum flow.

However, there are several important differences between a traditional flow network and an energy grid:

- Traditional flow networks have one source and one terminal (or sink), while grids can have multiple sources and sockets (when conceived as terminals).
- Traditional flow networks are directed graphs, and therefore have unidirectional edges, while energy cables (when conceived as edges) are bidirectional.
- Nodes in traditional flow networks do not have limited capacities, while transformer stations, sockets and sources in energy grids do (when conceived as nodes).

DESIGNING CEL AS A FLOW NETWORK

In this section we discuss how we chose to conceive our energy grids as flow networks, thereby resolving the aforementioned issues.

Initial representation

Let us see how we can represent an energy grid as a flow network step by step. Grid A (Figure 1) is used as an example throughout this section. It consists of two sources generating 11 energy units in total that are connected to a transformer, which is in turn connected to two sockets. The capacities of all elements are indicated. In this example we assume that the two teams specified the distribution of the 11 energy units as follows: Team 1 gets five energy units and Team 2 gets six units. Upon further examining the capacities in this example, in practice Team 1 could indeed get its requested five units, but then Team 2 would only get three units. This is how we want the system to work.

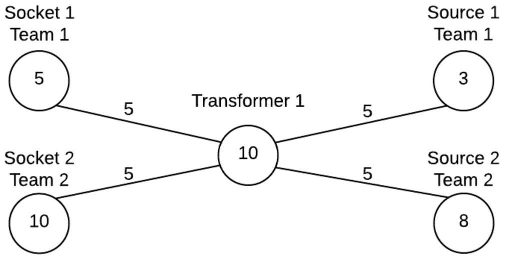

Figure 1: Example Grid A

If we convert this directly to a flow network, we get what is represented in Figure 2. The source *s* and terminal *t* were added, all edges became directional, and all nodes (energy sources, transformer and sockets) lost their inherent capacity. The capacity of the energy sources and sockets are used as the capacity of the edge connecting them to the source and terminal. The capacity of the transformer is lost, which is a key problem. If we were to run this network in a maximum flow algorithm, the results would probably not match our expected outcomes.

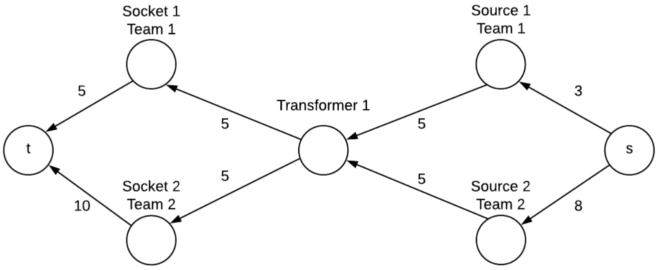

Figure 2: Initial Flow Diagram of Grid A

Allowing for energy distribution

The flow network of Figure 2 is not properly taking the energy distribution set by the two teams into account. The distribution essentially specifies a combined capacity for all sockets of a team in the grid. In the example both teams only have a single socket. Therefore the problem could be solved by setting the capacity of the edges to the terminal to the expected value of that socket's team. However, if a country had multiple sockets this approach would not work, as the flow network would not be able to keep track of the total amount of energy that each team is expecting. To accommodate this option, an intermediary node of each team's expected energy is added (Figure 3).

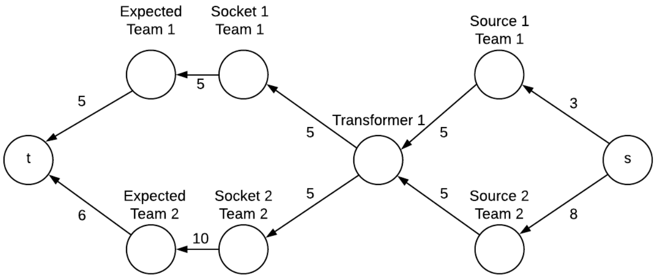

Figure 3: Grid A Limited by Energy Distribution

Introducing node capacity

The energy sources can receive a limited amount of energy from the flow network's source, and the energy sockets can send a limited amount of energy to the terminal. However, how much energy flows through the nodes themselves is not limited in our example flow network. As already mentioned this is particularly a problem for the transformers.

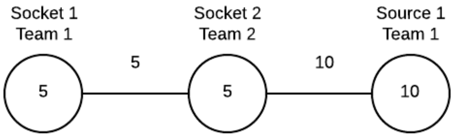

Figure 4: Example Grid B

Consider Grid B (Figure 4), a new example where the distribution of the 10 energy units is as follows: Team 1 gets five energy units and Team 2 gets five units. In our currently proposed design we would have to represent this with the flow network of Figure 5.

In Figure 5 both teams would receive the total amount of five energy units they each expect (at the nodes Expected Team 1 and 2 respectively). However, Socket 2 would be noted as passing on 10 energy units (rather than only receiving that amount), even though its capacity is only five units. This same problem applies to energy sources; they could pass on more energy than their capacity should allow.

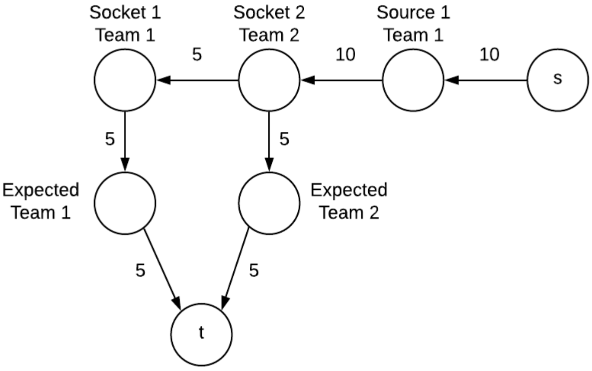

Figure 5: Grid B as a Flow Network

Adding node capacity solves this particular problem. To add node capacity to the flow network we split all nodes into two parts: input and an output. The input part of the node has an edge to the output part with the capacity of the original node. Figure 6 shows how Grid B would be affected by the introduction of node capacity to the flow network.

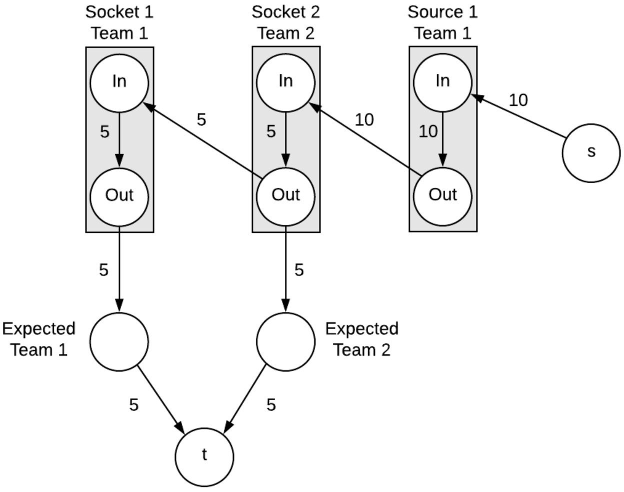

Figure 6: Grid B with Split Nodes

Applying node capacity on our original Grid A leads to Figure 7. This figure combines the added node of each team's expected energy to accommodate for energy distribution (Figure 3) with the socket's input-output distinction for node capacity.

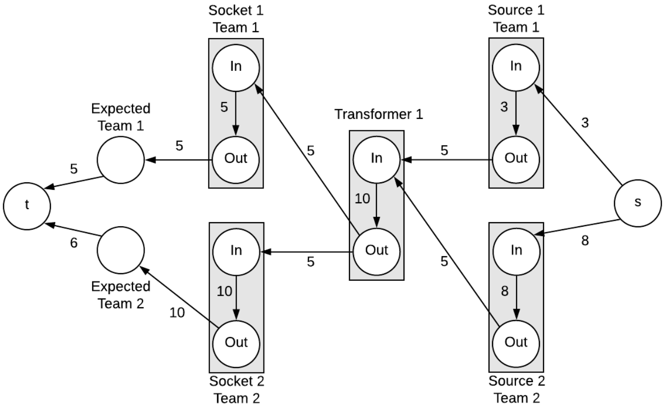

Figure 7: Grid A with Split Nodes

Introducing bidirectional edges

In an energy grid the maximum flow can be achieved by only using directional edges. This is because energy will never flow through a cable in both directions at the same time. Of course, in different time frames energy can pass through a cable in different directions. One could thus argue that all cable edges should be bidirectional. Yet from a flow network perspective, even when an edge is bidirectional, energy will only ever pass through it in one direction at the same time.

This is why, until now, we represented our cables with unidirectional edges and determined this direction beforehand. This was not a problem in the example of Grid A, since it was obvious in what direction energy would pass through the cable. However, when grids become more complex and contain loops, predetermining directionality becomes more complex. Instead of trying to determine directionality beforehand, we simulate bidirectional edges and let the maximum flow algorithm determine which way to send the energy through it.

To accomplish this we duplicate the edges representing cables, with the duplicate having the same capacity but a reversed direction. Because the nodes are split into two parts, this reversed edge would go from an input node to an output node, which is incorrect. The new edge should therefore be moved to go from the output node of its origin to the input node of its destination. Applying this to Grid A results in Figure 8.

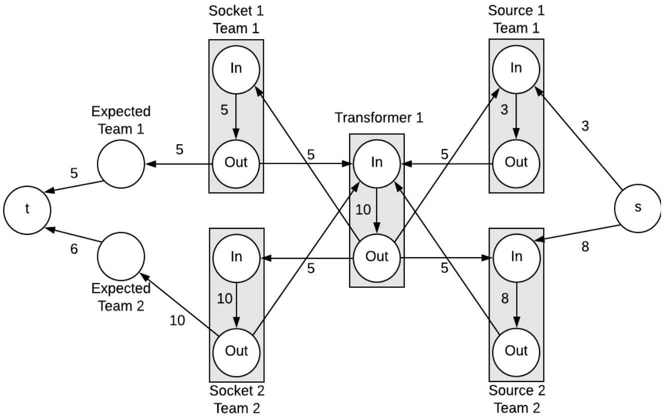

Figure 8: Grid A with Bidirectional Edges

In summary

With these changes we can now represent energy grids as flow networks while still retaining the important properties of an energy grid:

1. To limit the energy per team to the value they have defined in their grid design, we added an additional node of expected energy per team, connecting their sockets to the terminal with an edge with a capacity of their expected value.
2. To represent node capacities we split all the nodes representing objects into input and output parts, connected with an edge with the node's capacity.
3. To represent bidirectional edges we duplicated all the edges representing cables and reversed their direction.

The created flow network can now be used as input for a maximum flow algorithm to calculate the flow of energy through a grid per simulated month in MSP Challenge.

CEL IN PRACTICE

Path prioritization

When putting the presented design into practice, a final consideration that has to be made is how energy is distributed among the appropriate teams if their expected values cannot be reached. In other words: how much does each team actually get when none of their expectations can be met? There are two main options to handle this situation:

1. The energy is spread equally.
2. It is undefined.

The first option has a lot of additional considerations. How is ‘equally’ defined? What happens if energy cannot be distributed equally? Do we prioritize an equal distribution over the maximum flow? This option can also not be implemented by changing the the flow network design. It requires changing the maximum flow algorithm applied.

For MSP Challenge the second option was deemed appropriate. If players plan well, their expected values will be reached. If not, the maximum flow algorithm will simply do its work, determine that the maximum flow does not equal the expected flow and maximum capacity of the energy sources, and present results based on whatever paths the algorithm ended up with. CEL feeds back the number of energy units that every object passes on. Thus the residual energy can be calculated and fed back to the players as ‘wasted’ energy. This information is an incentive for players to analyse and revise their grid design as part of their MSP, serving the very purpose of MSP Challenge.

Performance

The presented additions to the flow network design naturally decreases performance when compared to a direct conversion to a flow network (as posed in Figure 2). We should note that this is not a fair comparison, because a direct conversion would not adhere to the properties of an energy grid. Nevertheless, the comparison can give us an indication of the additional performance cost incurred.

The performance of maximum flow algorithms is usually expressed in E and V, where E is the number of edges and V the number of vertices (nodes). Our additions to the standard flow network change these values into E' and V' in the following ways, where T is the number of teams in the grid:

$$E' = 2E + T + V \tag{1}$$

$$V' = 2V + T \tag{2}$$

In MSP Challenge Dinic's algorithm (Dinic 1970) was used for its robustness and good performance. Dinic's algorithm has a complexity *F* (equation 3). With our additions the complexity *F'* needs to be calculated as shown in equation 4.

$$F = O(V^2\ \ E) \tag{3}$$
$$F' = O((V + T)^2\ \ (E + T + V)) \tag{4}$$

In practice, with the average amount of grids in MSP Challenge, the energy simulation takes less than 30ms to run a simulated month. This includes the data requests done to the server. We note that we tested the solution on high-end hardware.

CONCLUSION

In this paper we presented how a dive into the complexities of offshore energy production with the help of experts from the NorthSEE, Baltic LINes and SIMCelt consortia and networks led to an amended flow network simulation called CEL. We chose to represent offshore energy grids as flow networks, with three important additions: an additional node for each team's expected energy, a split of each node into input and output parts to include node capacity, and bidirectional edges for all cables to enable more complex energy grid designs. Implemented with Dinic's algorithm, we have an energy simulation CEL that is suitable for use by MSP authorities and their energy stakeholders within the simulation game platform MSP Challenge. CEL's performance is at such a level, that it can be safely and usefully incorporated into the MSP Challenge platform.

With this design CEL enables MSP authorities and their energy stakeholders to use MSP Challenge 2050 for designing and actually testing more comprehensive offshore energy grids. An additional value of our approach is that we can also use it for other energy infrastructures, notably fossil fuel energy production (offshore oil and gas notably). An advanced use of MSP Challenge could thus concern the decommissioning of fossil fuel energy production balanced with the development of offshore renewable energy production. This way MSP Challenge would be used for offshore energy transition management.

We note several limitations in our approach. Our energy simulation is still a simplification of an offshore energy system. The simulation also does not take external influences on energy flow into account, such as fluctuating wind speeds. We note, however, that we can deal with the latter outside CEL by letting MSP Challenge dynamically configure the maximum capacity of an energy source with each discrete event (thus each simulated month). Still, we have to keep in mind that our goal is not to offer a complete offshore energy electrical engineering design system. MSP authorities need to know enough about the electrical engineering involved to come up with comprehensive and theoretically feasible offshore energy production designs befitting relevant policy objectives and within the wider MSP context. We believe the presented CEL design helps achieve that.

FUTURE RESEARCH

Our next step is to apply CEL in offshore energy MSP Challenge sessions within the North Sea and Baltic Sea regions. We aim to help these regions' MSP authorities and energy stakeholders plan for offshore energy in an integrated manner, thus also considering other human activities as well as the impacts and protection measures for marine life. These sessions are part of the NorthSEE and Baltic LINes projects and will take place at the end of 2018 and beginning of 2019.

The question remains how and to what extent MSP Challenge and CEL will aid MSP processes in the different sea regions we apply them. Moreover, as the MSP Challenge platform continues to develop, the question remains whether and how we should adjust or optimize CEL to improve performance or enable new features. A different maximum flow algorithm could help improve performance, if needed. Related to this question, is the matter of overall MSP Challenge development and support methodology. How should the platform be maintained, further developed and used, and by whom? We aim to answer these questions in the coming years.

ACKNOWLEDGEMENTS

The research leading to these results acknowledges the contribution of the NorthSEE, Baltic LINes and SIMCelt projects, which respectively received funding from the European Union's Interreg North Sea Region (NSR) and Baltic Sea Region (BSR) programs (both part of the European Regional Development Fund), and the European Maritime and Fisheries Fund (EMFF).

REFERENCES

Ahuja, R. K. 2017. *Network Flows: Theory, Algorithms, and Applications*. 1st ed. Pearson Education.

Bekebrede, G.; J. Lo; and H. Lukosch. 2015. "Understanding Complex Systems Through Mental Models and Shared Experiences." *Simulation & Gaming* 46, No.5, 536–62. https://doi.org/10.1177/1046878115621463.

Bilsen, A. van; G. Bekebrede; and I.S. Mayer. 2010. "Understanding Complex Adaptive Systems by Playing Games." *Informatics in Education* 9, No.1, 1–18.

Borrmann, R.; K. Rehfeldt; A-K. Wallasch; and Silke Lüers. 2018. "Capacity Densities of European Offshore Wind Farms." http://www.vasab.org/index.php/documents/doc_download/1581-capacity-densities-of-european-offshore-wind-farms.

Dinic, E A. 1970. "An Algorithm for the Solution of the Max-Flow Problem with the Polynomial Estimation." *Doklady Akademii Nauk SSSR* 194, No.4, 1277–80.

European Union. 2014. "Directive 2014/89/EU of the European Parliment and of the Council of 23 July 2014 Establishing a Framework for Maritime Spatial Planning." *Offical Journal of the European Union* 2014 (Apr), 135–45.

Fang, J.; C. Su; Z. Chen; H. Sun; and Per Lund. 2018. "Power System Structural Vulnerability Assessment Based on an Improved Maximum Flow Approach." *IEEE Transactions on Smart Grid* 9, No.2, 777–85. https://doi.org/10.1109/TSG.2016.2565619.

Kafas, A; M. Ripken; K. Wright; M. Billet; S. Sangiuliano; E. Ooms; and U. Scheffler. 2018. "Status Quo Report on Offshore

Energy Planning Provisions in the North Sea Region." http://www.northsearegion.eu/media/4930/northsee-offshore-energy-status-quo-main-report-final-version-120418.pdf.

Mayer, I.S. 2016. *Playful Organisations & Learning Systems*. Breda: NHTV Breda University of Applied Sciences. https://www.researchgate.net/profile/Igor_Mayer/publication/304816825_Playful_Organisations_Learning_Systems/links/577bc2eb08ae213761cab129/Playful-Organisations-Learning-Systems.pdf.

Mayer, I.S.; Q. Zhou; X. Keijser; and L. Abspoel. 2014. "Gaming the Future of the Ocean: The Marine Spatial Planning Challenge 2050." In *Serious Games Development and Applications*, 150–62. Lecture Notes in Computer Science. Springer, Cham. https://doi.org/10.1007/978-3-319-11623-5_13.

Mayer, I.S.; Q. Zhou; J. Lo; L. Abspoel; X. Keijser; E. Olsen; E. Nixon; and A. Kannen. 2013. "Integrated, Ecosystem-Based Marine Spatial Planning: Design and Results of a Game-Based, Quasi-Experiment." *Ocean & Coastal Management* 82 (Supplement C): 7–26. https://doi.org/10.1016/j.ocecoaman.2013.04.006.

McGowan, L. 2018. "Offshore Wind: Maritime Sector Briefing Note." http://www.simcelt.eu/wp-content/uploads/D2c_Offshore-Wind-Briefing-Note.pdf.

Schrijver, A. 2002. "On the History of the Transportation and Maximum Flow Problems." *Mathematical Programming* 91, No.3, 437–45. https://doi.org/10.1007/s101070100259.

Stolte, W.; X. Keijser; A. de Kluijver; and A. J. Nolte. 2013. "A Conceptual Pressure-Response and a Simple Food Web Model for the Maritime Spatial Planning Challenge 2050 Serious Game." 1208167-000-ZKS-0007. Delft: Deltares.

WEB REFERENCES

www.balticlines.eu
www.mspchallenge.info
www.northsee.eu
www.simcelt.eu

BIOGRAPHIES

Kevin Hutchinson, Wilco Boode and Carlos Santos are all part of the Breda University of Applied Sciences (BUAS) games and media research and development team. Kevin Hutchinson functions as a core developer of MSP Challenge, while Wilco Boode functions as the lead designer, and Carlos Santos as the technical manager. Harald Warmelink is a senior researcher at BUAS, and functions as project manager of MSP Challenge. Igor Mayer is professor of serious and applied gaming at BUAS, and functions as the project leader of MSP Challenge.