FastPress Project: Alpha Report

November 2024

Faculty Science Ltd, 160 Old Street, London, EC1V 9BW, UK | info@faculty.ai
Company Registration No. 08873131 UK | VAT Registration No. GB 202 8674 18

Contents

01 Executive Summary

This report outlines the work completed by Faculty during the Alpha phase of the FastPress project. Overall the project aims to identify areas where AI could enhance / automate a gas network planning analysts workflow. Through the initial Discovery phase, the team identified the opportunity for a tool to be developed that could automate the resolution of static gas scenarios, leveraging the SIMONE hydraulic modelling tool and eventually be integrated with existing NESO systems to streamline scenario resolution and enhance operational efficiency. This new tool developed by Faculty is called the Base Scenario Solver (BSS).

The primary objective of this phase of work was to validate the feasibility of the BSS tool, and the different solution components of it that were identified during the Discovery phase, at proof-of-concept (PoC) level. By the end of the phase, both Faculty and NESO should have a good enough understanding of the technical and conceptual functionality of the tool to determine whether it is possible to build and integrate, be sufficiently performant and be utilised by analysts. The results of this work are presented in Section 3.

Additionally, an exploratory workstream was initiated to assess the potential application of data science techniques for identifying pipes that could be targeted for removal (Pipe Removal Recommender). While this investigative work is still at a relatively early stage, it seems promising and the results of the testing warrant further exploration in future phases. The relevant detail is outlined In Section 4.

In parallel with the technical development of the Base Scenario Solver and the Pipe Removal Recommender tool, Faculty worked on the design and iteration of front-end wireframes. These wireframes underwent multiple cycles of user testing and feedback, allowing us to refine the interface based on practical insights and user preferences. This process ensured the evolving design aligns with user expectations and supports intuitive interaction with the tool's core features. A summary of this work is provided in Section 5.

We conclude the report by proposing a roadmap for continuation of the project. This is comprised of:

- A Beta phase for the Base Scenario Solver tool, which will continue to build on the work developed during this phase, refine the existing features to ensure their performance and therefore their adoption by users, and begin to bring the tool into production, and;
- An additional Alpha+ phase which will allow for further exploration of the Pipe Removal Recommender tool, developing it and other associated tools/features into an overarching Holistic Pipe Removal Pathway Planning capability. This capability will transition analysts away from manually evaluating the impact of removing single pipes and provide them with the ability to generate long term, sequential pipe removal strategies to meet strategic goals.

02 Introduction, Context and Report Objectives

02.1 Wider context

Following the passing of the 2023 Energy Act, a legislative framework for a new, publicly owned Independent System Operator and Planner, known as the National Energy System Operator (NESO) was introduced. NESO will take on responsibilities across electricity, gas and hydrogen, including all the existing functions of NGESO, in order to take an enhanced whole system approach to planning and operating the energy sector. This is required to help plan and deliver the integrated system needed to secure the UK's energy security, net zero and affordability goals. It will operate as an independent body, free from other commercial energy interests and from the operational control of government.

In addition to its role and responsibilities in the electricity sector, NESO will take on responsibility for gas strategic planning, gas forecasting, and market strategy functions. As set out in the Department for Energy

Security and Net Zero (DESNZ) Future System Operator policy consultation and update[1], NESO is expected to coordinate a strategic planning gas options assessment process, in which:

1. NESO will publish an independent view of gas network capability and its statement of system needs.

2. National Gas Transmission (NGT) (and potentially other parties in the future) will produce a response to the needs set out by NESO which may include a proposal of options to satisfy these needs.

3. NESO will produce a form of gas options assessment/advice document that will include analysis and recommendations/advice to Ofgem on the proposals under step 2.

4. NESO will submit its analysis/advice on proposals and options (from step (3)) to Ofgem for further decisions on funding.

It is intended that NESO will also undertake medium-/long-term gas supply and demand forecasts, as an output of the Future Energy Pathways process. These assumptions and forecasts are likely to feed into the strategic planning process, as well as being communicated to wider industry. Finally, to complement existing security of supply standards focussed on supply infrastructure capacity, NESO will deliver a new medium range (at 5 and 10- year time intervals) Gas Supply Security Assessment on an annual basis. This assessment will consider gas supplies according to availability, reliability, and deliverability across specific time horizons.

In order to meet these objectives, NESO will have a number of teams of analysts who are responsible for gas network planning. Gas network planning is the process of designing, developing, and managing the infrastructure required for the transportation and distribution of natural gas from production sites to consumers. This involves several critical tasks such as forecasting gas demand, ensuring a reliable and safe supply, optimising the network's performance, meeting commercial obligations and maintaining regulatory compliance.

02.2 Project context

The FastPress project is a 24-week innovation project designed by NESO to understand how artificial intelligence (AI) can be integrated into the workflows of gas network planning analysts. The project, delivered by Faculty, has been divided into two distinct phases: a Discovery phase which was delivered from June - August of this year (2024) and an Alpha phase, which this report concludes, delivered from September - November.

During the Discovery phase, the Faculty team conducted user research and mapped existing analyst's workflows to understand the current tooling in use and to identify pain points and problem areas, in order to ultimately determine suitability for possible AI applications. The findings from the Discovery phase, summarised in this report, uncovered several key challenges within the current workflow of gas network planning analysts:

1. ***Manual Process for Solving SIMONE Scenarios***: Despite initial automation through the Scenario Creation Tool (SCT) and the use of a standardised master network, solving SIMONE scenarios remains a largely manual process. Analysts spend significant time and effort manually intervening in scenarios, which diminishes overall efficiency.

2. ***Repetition of Work Across Analyst***s: Analysts typically export network configuration settings in a custom file format after reaching a solution. This results in repeated work, as different analysts might tackle the same scenario independently, leading to inefficiencies and increased likelihood of errors.

3. ***Subjectivity and Lack of Standardisation***: Adjustments to configurations heavily rely on individual analyst knowledge and expertise, making the process subjective and unstandardised. This variability can result in inconsistent outcomes and a lack of coherence in the methods applied across the team. It also makes training new analysts more challenging.

[1] https://www.ofgem.gov.uk/sites/default/files/2024-03/Main%20document%20-%20Statutory%20consultation%20on%20National%20Energy%20System%20Operator%20licences%20and%20other%20impacted%20licences.pdf

4. ***Restricted Scope of Analysis***: The time-intensive nature of running and solving scenarios limits the number of configurations and scenarios that can be explored. Consequently, the scope of analysis is constrained, inhibiting NESO's ability to assess a wide range of scenarios thoroughly.

5. ***Absence of Intelligence Collection on Unsuccessful Configurations***: Current practices involve saving only the parameters for successfully solved scenarios, leaving a knowledge gap regarding failed configurations. This lack of documentation prevents learning from past attempts and refining future strategies.

6. ***Inconsistent Data Governance***: Mixed levels of data governance have been identified, introducing vulnerabilities where incorrect input/output versions may be used, risking data integrity and reliability.

7. ***Lack of Validation Methods***: Analysts currently lack a validation method to ensure input data accuracy, creating potential for undetected errors that compromise analysis validity.

Based on the above problem areas, a series of proposed features were conceptualised and developed by the Faculty team in order to address the identified challenges. The full list of proposed features can be found in the FastPress Discovery Phase report and are presented below in Figure 1. These features were refined and prioritised in collaboration with analysts and key stakeholders, and discussions on which features to progress to the Alpha phase of the project focused on sequential planning of features which would enable development of a basic tool for analysts to solve base scenarios. Collectively, these features/solution components comprise the Base Scenario Solver tool. In addition, a parallel workstream of data analysis and development relating to asset (pipe) removal/decommissioning was prioritised for further exploration (Piper Removal Recommender tool).

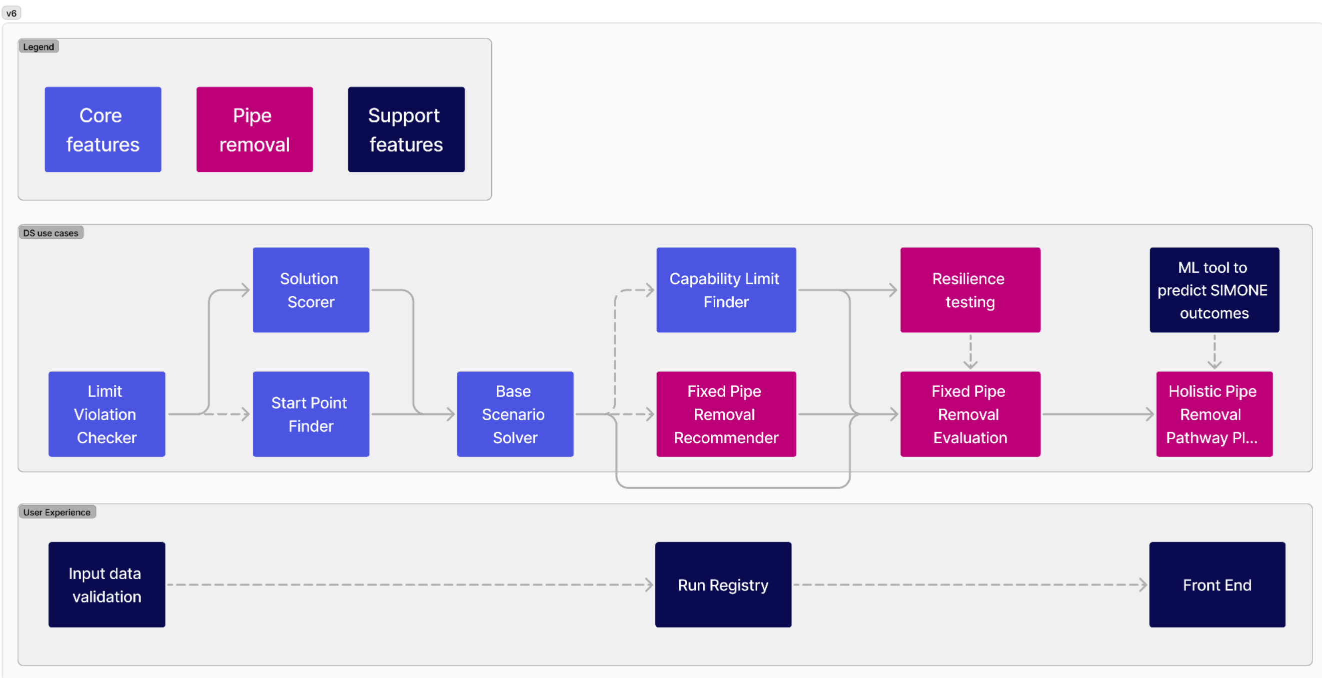


Figure 1: Roadmap of proposed feature development generated during FastPress Discovery phase.

Subsequently it was determined that the following features would be progressed to the Alpha phase of the project. For clarity, whilst features 1-3 are required to deliver the Base Scenario Solver tool, they are useful in their own right in isolation:

1. ***Limit Violation Checker:*** a function that checks against limits on the network to make sure all conditions are met for a scenario solution and flags when SIMONE limits have been violated. We are intending for this feature to produce a pass/fail result.
2. ***Solution Scorer***: a function that provides a continuous score for how well a solution performs based on the distance of assets from their limits - helping the overall scenario solver tool assess the quality of solutions in an automated fashion.
3. ***Start Point Finder:*** a function that compares the new scenario to a database of previously solved scenarios, identifies the one with the most similar entry and exit flow configurations and produces an asset configuration for this scenario as a starting point for the new solution.

4. ***Base Scenario Solver:*** the ultimate capability that the above features support, this tool can search the compressors, regulators and valves (CRV) solution space for the best configuration of a network for a given scenario. It relies on an optimisation algorithm to search the space of possible configurations until it can find an optimal solution, and will be accelerated by starting the search from the output of the Start Point Finder feature.
5. ***Fixed Pipe Removal Recommender***: a separate tool that intends to use past solution data to identify the importance of pipes and recommend those that are appropriate for removal.

The features will all form part of an overall tool, which has the following high level user requirements:

- The tool should help Gas Networking Planning Analysts solve static base scenarios, run exit capability analysis on static scenarios and run entry capability analysis on static scenarios
- The tool should standardise assessment criteria of solved scenarios
- The tool should provide traceability and auditability of all SIMONE scenario runs
- The tool should allow/enable sharing of results within teams.

A full list of the user requirements and functional/non-functional requirements can be found in Appendix 09.1 and Appendix 09.2.

The intention for the Alpha phase was to develop these features into a PoC tool, essentially testing and determining each feature's feasibility by building a minimal and simplified version of it. The primary objective of this was to verify whether these features and the overall tool have practical potential to assist the analysts in their workflows, and to determine whether they could be successfully implemented and integrated within existing NESO workflows if developed further. This testing was also intended to determine the core functionalities and identify/address critical technical challenges, as well as to test assumptions and get a good understanding of the potential risks and obstacles to further development.

Whilst the overall aim of the FastPress tool is to automate the overly manual tasks undertaken by gas analysts, it is crucial that the proposed mechanism for interacting with the tool does not contradict the analysts' mental model of how such a process should work. This is reflected in the change shown in the following diagrams, contrasting the previously identified process with the proposed new process enabled by the FastPress tool. The key difference between the two processes is the significant reduction in time required to solve a static scenario.

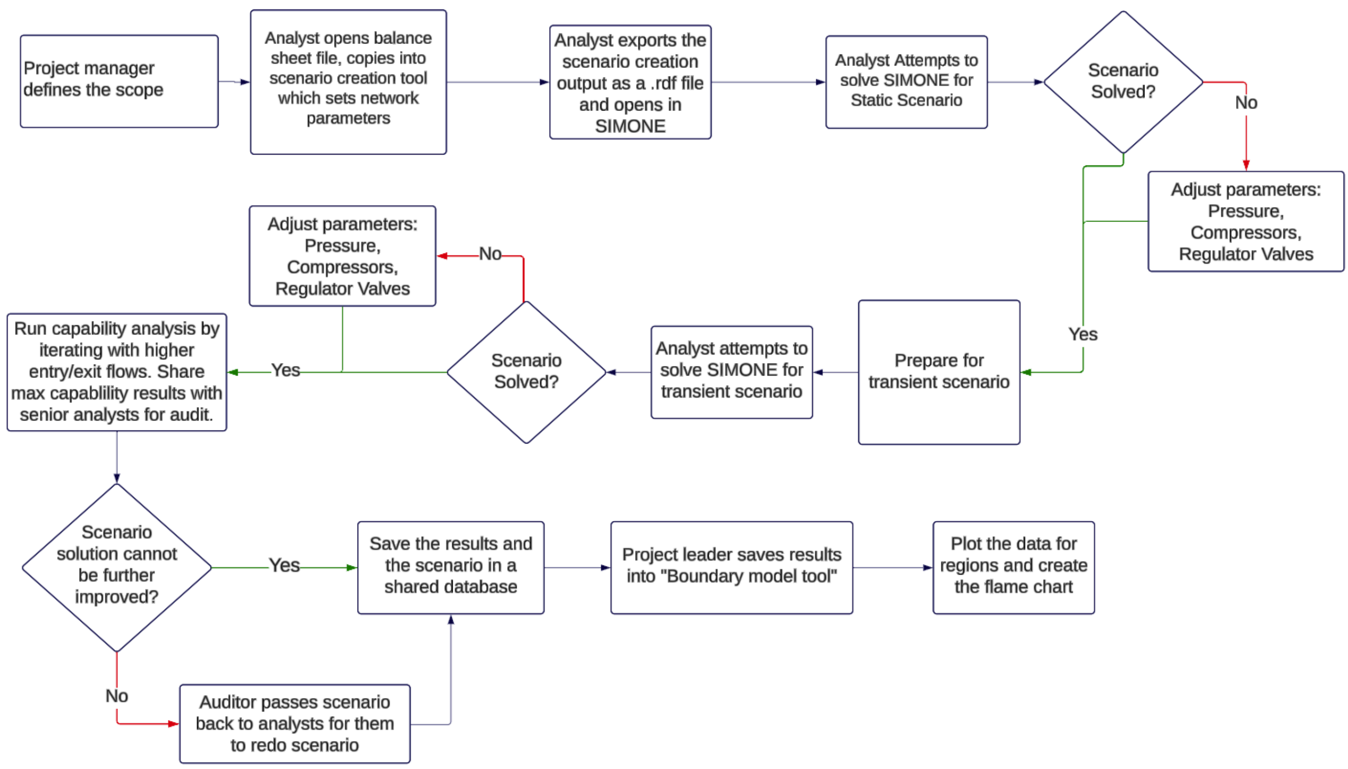


Figure 2: As-is user process diagram (i.e. without the FastPress tool)

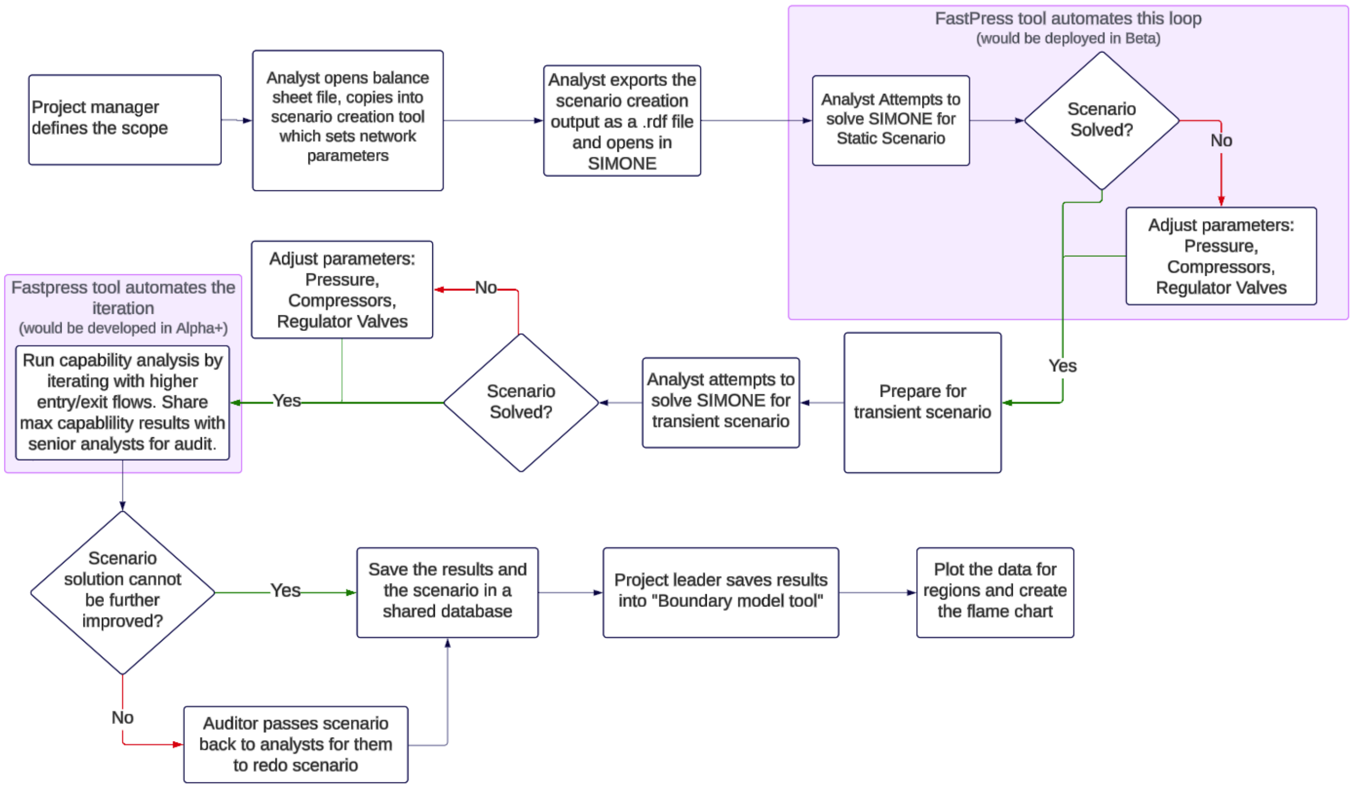


Figure 3: Updated user process diagram (i.e. with the inclusion of the FastPress tool)

02.3 Report objectives

The objective of this report is to outline the work that Faculty have undertaken during the Alpha phase of the project, with the aim of:

- Outlining the functional and non-functional requirements of the tool that we have determined and developed through user interaction with the PoC features
- Explaining the features that have been developed, the status of each feature, the function of the feature in relation to the overall tool, the lessons learnt during development and the areas for further development and improvement
- Demonstrating the user interface that we have designed for the tool and the process of developing this with users
- Explaining key structural decisions we have made for the tool and the rationale for these
- Exploring the roadmap for further development of the tool, including the items that are outstanding for resolution and the process for embedding and integrating the tool within NESO workflows.

Overall this report should provide the reader with a good overview and understanding of the proposed tool and its features, how users will interact with the tool in future and how they will benefit from its further development.

03 Base Scenario Solver

The goal of the Base Scenario Solver (BSS) is to automatically "solve" static base scenarios, where solve is defined as a scenario that would pass Standard Operating Procedure (SOP) audits. This is a complex, multistep process that can take up a large amount of analyst time, especially when focusing on more challenging problems. Due to this complexity and the difficulty of finding a solution, the algorithmic BSS solution requires support by several other algorithmic solutions, that emulate or complement analyst behaviour (*i.e.* the Limit Violation Checker, Solution Scorer and Start Point Finder)

03.1 Methodology and approach

To create a PoC that effectively demonstrates core capabilities without overcomplicating the tool, several strategic decisions were made to narrow the solution's scope. Prior to commencement, the FastPress project was restricted to static scenarios, which was an important first step. This moderation in scope not only makes the problem more manageable and easier to handle with existing domain expertise, but it also accelerates the process of obtaining results. Static scenarios are quicker to experiment with, unlike transient scenarios that require more time to solve. Furthermore, addressing static scenarios is an essential preliminary step before transitioning to a transient solver.

The goal of this PoC is to address base scenarios, specifically focusing on solving them without delving into capability analysis. The intention is to tackle the "solving base scenario" component of the network capability analysis, which is the most significant and time consuming part of that workflow. This involves solving scenarios by adjusting key elements like compressors, regulators, valves (CRV), control linepack, and supply flows using EADI PSET settings. The PoC does not perform entry/exit capability analysis or any regional flow redistribution; however, it is capable of accepting scenarios that have been manually modified by a gas analyst through flow redistribution.

To achieve these objectives, several components are necessary:

- Limit Violation Checker: this plays a critical role in reproducing the checks done by SIMONE and gas analysts to determine if a scenario is successful or has failed.
- Solution Scorer: provides the algorithm with feedback on its performance to ensure alignment with the correct solution path.
- Start Point Finder: reduces the search space by identifying favourable starting scenarios.
- Optimisation Algorithm, specifically a Genetic Algorithm: employed to explore the search space effectively.

03.2 Discussion of key features

03.2.1 Limit Violation Checker

The Limit Violation Checker (LVC) is an essential component that underpins the ability to determine when a scenario can be classed as ‘solved’, serving as a foundational element for the Solution Scorer feature that is necessary for the Base Scenario Solver. To this end, a solved scenario is defined as a scenario that successfully comes to a hydraulic solution in SIMONE and does not breach any operational limits on the network. These limits are not features that are included in the physical simulation. They include checks surrounding asset limits—such as ensuring that pressure levels remain within safe margins and that compressor units do not overheat—as well as commercial requirements, like guaranteeing the delivery of gas above a specified minimum pressure.

The current process for limit checking is a combination of automated checks carried out in SIMONE and manual inspections by gas analysts. Most limits are encoded in the master network.[2] These limits can easily be flagged in the SIMONE UI by gas analysts using a ‘limits violated’ tool that presents the assets that are outside of their permitted range of values. Some checks are not covered by this table e.g. whether or not compressor units are running within their envelopes[3], which is typically checked by opening a separate window within the UI and manually inspecting the envelope graph. The remaining limits are checked by manually comparing asset pressures, temperatures, and flows from the SIMONE UI to separate documents describing more nuanced limits and/or limits that otherwise are not included in the master network.

The LVC is structured around a sequence of checks:

1. Successful execution of the scenario in SIMONE.
2. The actual linepack—a measure of the gas volume stored within the network—remains within one million cubic metres of the target linepack, where the target linepack is a function of the total demand across the network.

[2] The ‘master network’ refers to the most up-to-date representation of the transmission network as SIMONE network files. These are produced by NGT and undergo regular updates and audits to ensure their performance.

[3] A compressor envelope is a predefined window of flow rates and head in which the compressor unit operates safely.

3. Supply flow matches offtake flow to two decimal places when using units of million cubic metres per day.
4. Confirmation that no assets are breaching their associated limits.

The asset-level checks are carried out by reproducing both the checks that are typically carried out in SIMONE and the checks that are currently carried out by manual comparison to documentation. It is important to integrate even the checks that are already automated in SIMONE into the LVC in order to support automatic collection of results by the tool. In addition, by embedding these checks within the LVC, we maintain flexibility for future developments that are based on checks against limits, such as the Solution Scorer. The integration of SIMONE asset checks relies on the limits being accurate in the network used, so it is important that the correct version of the network is used and users should be aware that changes they make to the limits in the scenario will be propagated to the LVC.

The SIMONE-based asset checks include ensuring that nodes maintain pressures between their maximum and minimum limits and that temperatures do not exceed maximum thresholds. When distinct strict safety limits and softer limits for optimal network resilience—such as pressure cover limits at terminal entry points—are involved, the LVC defines a 'failure' based on any breaches of the strict limits. However, the softer limits are documented to assist in the ongoing evolution of features building upon the LVC, such as the Solution Scorer. In addition to the node limits, compressor units are checked to ascertain they are operating within their safe operating envelopes: their operation status must not be either 'choke' or 'surge' as recorded by SIMONE.

The currently-manual checks that are now included in the LVC include demand-dependent agreed operating pressures which allow for lowering specific node minimum pressure limits when national demand falls below a certain threshold. There are also asset-dependent constraints like compressor flow, RPM, and pressure limits, and ensuring that the Lockerley compressor station operates within its designated inlet pressure boundaries, as well as verifying that regulators operate in authorised modes and adhere to flow and pressure guidelines.

In the development of the LVC, certain checks were excluded, primarily due to considerations of development time versus potential benefit at this stage of the tool's evolution. One of the omitted checks is the exit point pressure cover. Exit point pressure covers are calculated to account for uncertainties around network behaviours and to ensure sufficient operational resilience surrounding these uncertainties. These pressure covers are dependent on the compressors in use and on both the regional and national demand. These dependencies complicate the incorporation of exit point pressure covers into the LVC, to the extent that including the pressure cover calculations would have extended the LVC development timeline. Although crucial for a fully deployed tool, exit point pressure covers were deemed relatively insignificant for the outputs of a PoC, as their inclusion would merely introduce another limit for an automatic solver to manage.

Similarly, the multi-junction checks were not included. Multi-junctions, which are collections of valves and compressor stations at network junctions, have associated approved configurations that dictate the allowed combinations of settings for the constituent assets. These configurations rely on assumptions documented externally, as SIMONE does not include built-in checks for this purpose. Attempting to implement multi-junction level checks presented two significant challenges. First, converting information on approved configuration from assumptions documents into tabular data required substantial input from gas analysts, which was not justifiable for the LVC alone. Second, despite the finite number of approved configurations, each configuration may have multiple functionally equivalent versions, differing in valve settings without affecting the multi-junction's behaviour. Accurate LVC functioning would necessitate comprehensive data on all possible permutations, demanding considerable analyst time. An attempt to implement checks using existing partial data resulted in frequent erroneous failure flags due to incomplete multi-junction coverage, diminishing LVC's utility. Thus, the decision was made that the time investment required to include multi-junctions within the LVC outweighed the potential benefits at this stage of development.

In conclusion, the development of the LVC has reached a successful milestone with the establishment of a working PoC. This version effectively demonstrates the feasibility of automating the detection of limit violations across network scenarios. However, several enhancements would be necessary to transition from this preliminary stage to a fully operational tool. A complete version would benefit from more rigorous auditing with gas analysts to ensure the accuracy and reliability of the checks performed. In addition, the checks that were excluded during this phase would need to be integrated, with the pressure cover calculations as a priority for ensuring the accuracy of the LVC. The integration of the multi-junction checks can be more flexible, as the usefulness of this check could depend on the workflow of gas analysts: even a version that only includes the default setup of each configuration may still provide a benefit.

Additionally, the tool would likely need modifications to enhance its robustness, particularly by simplifying its architecture to facilitate transitions to updated networks over time. This could involve tighter integration with the data contained in the master network, such as employing characteristic curves for the modelling of regulators. Such integration would not only improve the efficiency of limit data collection but also bolster the tool's adaptability to changing network parameters. Through these refinements, the LVC can be developed into a more robust and versatile solution for managing network operations.

03.2.2 Solution Scorer

The Solution Scorer is a key feature that is used to evaluate the quality of a scenario solution by comparing its results, whether the scenario passes or fails, against the defined scenario limits and requirements. The score is determined by evaluating how near its limits the scenario is operating and its proximity to achieving targets for linepack and supply/demand balance.

The utility of the Solution Scorer as a stand-alone tool lies in its ability to offer deeper insights than a simple binary pass/fail assessment for each condition. By identifying scenarios where assets are functioning close to their limits, it underscores potential vulnerabilities or a lack of resilience—even in scenarios that technically pass. Such insights are valuable in understanding not just whether a solution works, but how robust and sustainable it is under varying conditions. In the context of the Base Scenario Solver (BSS), the Solution Scorer is essential. The BSS requires a continuous metric for performance tracking instead of a binary pass/fail outcome in order to effectively navigate towards optimal solutions

The approach for implementing the Solution Scorer centres on measuring the proximity of each scenario to its operational limits, thereby assigning scores that reflect the resilience of the solution. Scenarios with assets that remain below operational limits receive favourable scores, improving as the asset properties are further from these boundaries, signifying greater resilience. Conversely, scenarios with assets that breach limits incur poor scores, with the severity increasing in relation to the extent of the breach.

The scoring methodology mirrors the steps employed in the LVC, but replaces the binary pass/fail outcomes with a continuous scoring model. The three key scoring steps include:

1. achieving the linepack target;
2. ensuring the equal supply and demand gas flows, and;
3. conducting asset-level checks against their respective limits.

Two key technical approaches define the Solution Scorer. Firstly, there is the score calculation function, which must be capable of running for multiple types and combinations of limits. Secondly, the solution scores are scaled using a Maximum Theoretical Error (MTE), providing a framework for consistent scoring across passing and failing scenarios.

The following criteria were used to define the score calculation:

- For each asset property (e.g. the pressure at a node), the scoring function should take as an input the value of the property and the limits associated with that asset property.
- The function has to handle comparing the property value to a minimum limit, a maximum limit, and cases where both minimum and maximum limits exist.
- The function also has to handle soft limits (i.e. limits indicating that the asset is not operating at a resilient level but where a breach does not have a severe enough impact to justify a failing scenario) in addition to strict limits. In the case where a soft limit is present in addition to the strict limit, the asset property that breaches the soft limit should score worse than if that soft limit was not present and the property value was therefore only being judged by its distance from the strict limit.

Given these conditions, an algorithm composed of four functions (Figure 4) was used, where a higher score indicates a worse solution and zero is the best possible score. The functions are:

1. An exponential function where the score increases the closer a property gets to its strict maximum limit (green solid line).

2. An exponential function where the score increases the closer a property gets to its strict minimum limit (blue solid line).
3. A linear function where the score increases the closer a property gets to its strict maximum limit from its soft maximum limit (green dashed line).
4. A linear function where the score increases the closer a property gets to its strict minimum limit from its soft minimum limit (blue dashed line).

The scoring algorithm applies the functions with the following conditions where the score is calculated using:

- Function 1 when only a strict maximum limit exists for the property,
- Function 3 when both a strict maximum limit and a soft maximum limit exist for the property and the property value is within the range enclosed by those limits,
- Function 2 when only a strict minimum limit exists for the property,
- Function 4 when both a strict minimum limit and a soft minimum limit exist for the property and the property value is within the range enclosed by those limits.

Where both minimum and maximum limits exist, the functions are applied according to the above conditions and the maximum value is taken. This approach means that where an asset property has both minimum and maximum limits, it scores best when it is far from both limits.

When applying the scoring calculation, the property values and limits are normalised such that the score is 1 for a property that is exactly at a strict limit.

Figure 4: An illustration of the scoring functions used as part of the Solution Scorer. A higher score indicates a worse performance, as a property value that is closer to its limits indicates a less resilient scenario.

The MTE is used to ensure that a failing scenario always scores worse than a passing scenario. Given how flexible the Solution Scorer is and the number of different conditions being compared, it would otherwise be possible to get behaviours like a scenario scoring better when it causes a failure at a single asset versus many assets being close to their limits

The MTE approach addresses this by scaling the individual scores (i.e. for the linepack condition, the supply/demand equality condition, and the score for each asset property) such that, when the scores are summed to generate a full scenario score, the score for a failing scenario is always worse than the score for any passing scenario. Given that the scoring function ensures that each individual component score is below 1 for a pass and above 1 for a failure, the worst score possible for a passing scenario is where every component score is at 1. Summing the number of component scores then generates the MTE for a scenario. The MTE is then added to any component score that is above 1, resulting in any individual failure having a higher (i.e. worse) score than a scenario with every asset operating at its limits.

In conclusion, the Solution Scorer is a functional tool that supports a more nuanced assessment of solution performance and facilitates the BSS by providing detailed assessments of scenario solutions. Although it has already reached a functional level, it could benefit from further user research to tune the scoring e.g. to upweight certain assets or requirements over others (i.e. assign greater penalty scores to certain components of the network). This upweighting approach could also be tuned through experimentation to improve the performance of the BSS, in which case the weights used in the Solution Scorer would effectively be hyperparameters of the BSS. Additional features could also be built into the Solution Scorer beyond scoring based on proximity to limits: for example, the scorer could reward using fewer compressor stations or using electric compressor units.

03.2.3 Start Point Finder

The Start Point Finder (SPF) addresses the inefficiencies in solving scenarios using random population settings within the BSS, despite its theoretical capability to do so. Given the vast search space of possible settings, leveraging prior knowledge of previously successful scenarios can significantly enhance solution speed and performance. The SPF is specifically designed to identify scenarios that closely resemble the current problem. Each scenario is characterised by its intake and offtake flows as well as its network architecture. This allows an analyst or the BSS to use those scenarios as a starting point, rather than relying on one default set of scenario settings. The SPF functions effectively as a standalone tool for analysts and for integration with the BSS.

To determine similar scenarios, it is essential to establish a numerical representation for each scenario. By comparing these representations, we can identify scenarios that share similarities. This method draws parallels with Natural Language Processing (NLP) techniques, which find semantically similar pieces of text. In the SPF, the numerical representation of a scenario is derived from its supply and demand parameters, which are the basis for finding appropriate settings to solve the scenario. Various methods exist for comparing the similarity between two vectors, and for simplicity and efficacy, we have chosen cosine similarity, a common approach in semantic similarity[4]. Utilising this method, the SPF can identify and return the top most similar scenarios for a target scenario, along with the settings of these successfully solved scenarios. These settings are the useful starting point for analysts or the BSS.

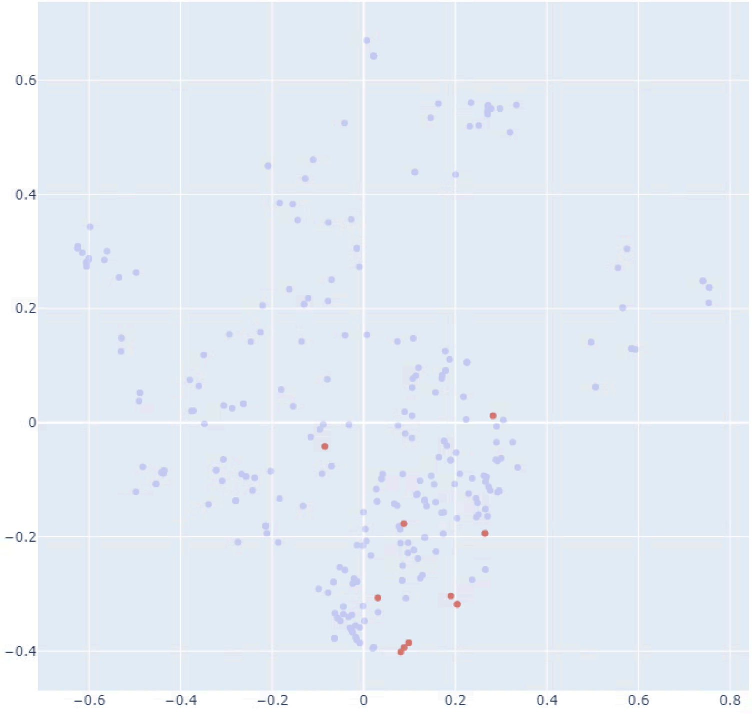


Figure 5: Start Point Finder in action: The closest 10 scenarios (red) selected by the SPF for a target scenario from the pool of all available solved scenarios (grey). Each point represents a scenario, mapped to a two-dimensional representation using the Multiple Correspondence Analysis (MCA). MCA only performs dimension reduction across categorical variables, so does not fully represent the scenarios - however it is useful to show that the SPF selects broadly similar solutions.

There is potential for improving the current implementation of the SPF by exploring alternative approaches for comparing numerical representations. Such alternatives may better accommodate the mixed data types—both binary and continuous variables—present within the scenarios, thereby enhancing the quality and similarity of the

[4] huggingface.co. (n.d.). What is Sentence Similarity? - Hugging Face. [online] Available at: https://huggingface.co/tasks/sentence-similarity.

returned scenarios. Alongside this, curating the database of solved scenarios available to the SPF can help it yield better results, as it currently has scenarios from a variety of networks whose settings may not be as useful.

Regardless of identified improvements, from our work developing this PoC feature we are able to conclude that by using a standard vector similarity measure, the SPF is capable of retrieving useful settings.

03.2.4 Base Scenario Solver

The purpose of the BSS was to find a combination of settings capable of successfully solving a scenario, essentially tackling an optimisation problem. To address this, we reviewed various optimisation algorithms, including Reinforcement Learning, Gradient Descent, Particle Swarm optimisation, and Monte Carlo Tree Search. Ultimately, we chose to use a Genetic Algorithm due to its flexibility and its ability to efficiently handle large search spaces and non-differentiable target functions.

Methodology
A Genetic Algorithm is a type of evolutionary algorithm that mimics the natural processes of hereditary selection in nature. Its terminology borrows heavily from the field of biology. We have included a brief glossary relating the common terms surrounding Genetic Algorithms to the specific implementation in the context of solving SIMONE scenarios.

Glossary of terms:

Name	***Evolutionary Meaning***	***SIMONE interpretations***
Population	*A group of individuals*	*A group of scenario settings*
Genome	*The attributes of one individual*	*One scenario settings file*
Gene	*One feature in a genome*	*A single setting in a scenario settings file*
Mutation	*Randomly modifying a gene*	*Randomly modifying a setting in a settings file*
Crossover	*Creating a new genome from two parents in the population*	*Creating a novel settings file from selecting parts of two existing settings files*
Fitness	*How well a gene does*	*The Solution Score of a settings file*

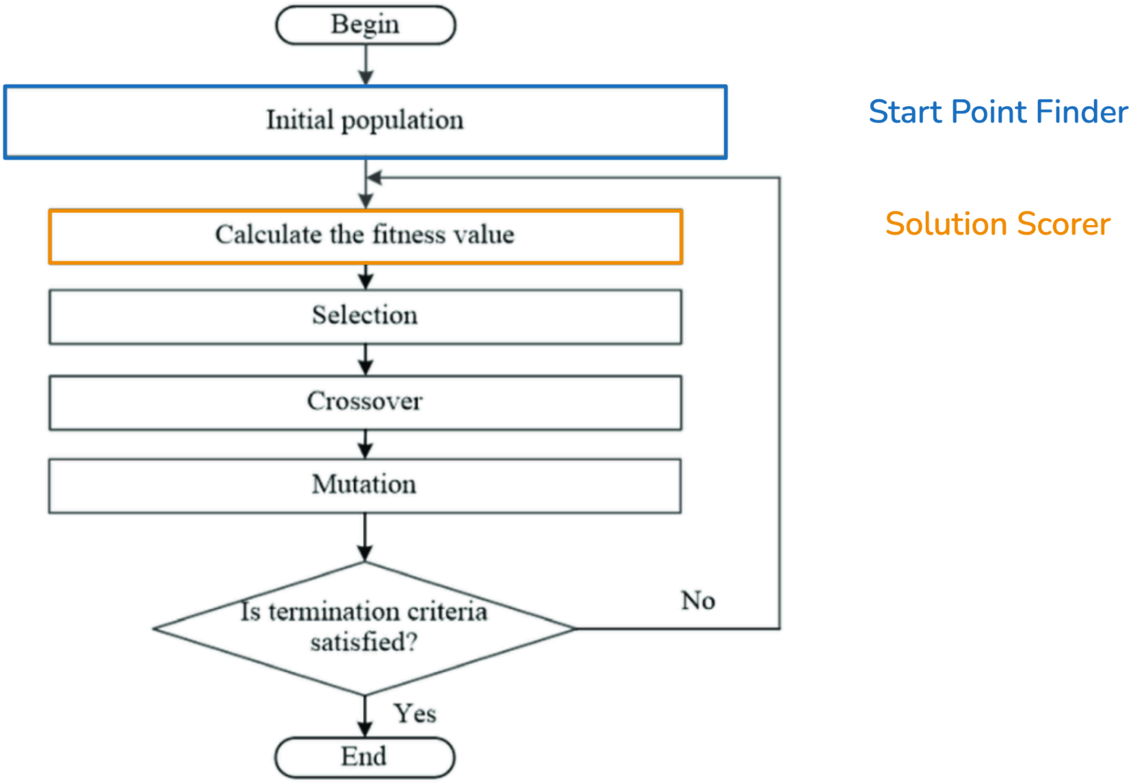


Figure 6: Flow diagram of the basic Genetic Algorithm process

The basic flow of a Genetic Algorithm begins with the creation of an initial population of solutions, referred to as "genes," which typically consists of a set of randomly generated asset configurations. Each of these solutions is evaluated for its effectiveness using a fitness function, which assigns fitness values accordingly. During the selection process, solutions are chosen based on their scores, typically with the highest scoring solutions progressing to the next step. Next, crossover is applied by mixing successful genes to produce new, potentially more viable solutions. Additionally, mutation is introduced by randomly altering the values of some members within the population. The algorithm continues to loop through these steps until a termination criterion is met, which could be as simple as surpassing a predetermined limit or employing more complex methods, like early stopping when no improvement is observed in a set number of iterations. Throughout this process, it is possible to monitor the performance of the population, providing a live view of the solution's progress and enabling informed decisions or adjustments to be made as needed.

Our approach to implementing the Genetic Algorithm was a modular build - we developed several interchangeable options for each functionality to effectively come up with the best approach and include dummy methods for testing and rapid development.

Multi-junction Support: One significant modification necessary in our implementation of the Genetic Algorithm was addressing the need to include only valid multi-junction configurations in all settings, a constraint that required the development of custom mutation and crossover approaches. Without special care, the mutation and crossover algorithms would generate multi-junction configurations that, while leading to physically valid solutions, have not been approved as safe according to the network planning assumptions documentation, meaning they would not be acceptable to analysts. To tackle this the mutation and crossover algorithms were modified to always output valid multi-junction configurations, while maintaining their role of creating diversity (see below). In contrast to the LVC, it was possible here to integrate multi-junction configurations without needing to document all possible functionally-equivalent versions of the configurations: the mutation and crossover functions can be restricted to only use the default version of each approved configuration.

Population initialisation: The main method for finding the starting population is the Start Point Finder. For testing and validation purposes there are random initialization approaches included, but these were not used in the running of the tool. This enables us to start from a set of known solutions that are the closest match for the scenario at hand, and therefore present our best guess.

Selection: This process selects the genomes to be parents based on their fitness score. In our case, the score was derived using the Solution Scorer, calculating a continuous score for each member of the population, allowing us to compare how well each genome / setting performs.

Crossover: During crossover, the presence of multi-junctions posed a significant challenge, as uniformly mixing settings from two parents could easily result in invalid multi-junction configurations. To address this issue, we first introduced a multi-junction maintaining crossover approach, which preserves the multi-junction structure of one parent by uniformly mixing setting but never modifying the original parent's valve settings or swapping compressors or regulators between on and off, ensuring that a valid configuration is maintained (assuming that the source data in the SPF is using only approved configurations). However this method did not generate diversity in the genome, so we developed a more detailed approach, where each multijunction is randomly chosen from one of the parents, promoting diversity among the multi-junction configurations of the children's genomes. Other variables outside of those covered by multi-junctions are also randomly selected from a parent.

Mutation: Several approaches were taken to ensure effective mutation within the Genetic Algorithm. For multi-junctions, the approach involves randomly selecting a multi-junction and applying a random approved configuration. If this configuration includes a compressor or regulator set to ON, a control mode is randomly selected along with the control mode setpoint. For continuous mutations, the algorithm considers a set number of genes, deciding with a given probability whether to introduce a mutation to a continuous value within the genome. If a mutation is introduced, it randomly selects a compressor station or regulator value to mutate, avoiding valves as they do not possess continuous properties. A random scaling factor is chosen within a set range and applied to the continuous setpoint property of the selected asset. Additionally, the mutation of the EADI PSET value and supply node flow values is also permitted. The categorical approach for mutation covers the mutation of the non-continuous settings of assets that are not part of any multi-junction: it randomly selects a non-multi-junction asset, and flips it between on or off (for valves), or between off, bypass, and different control modes (for compressors and regulators). When turning a compressor or regular on, a random control mode setpoint is used to match the control mode selected.

Population Immigration: This advanced evolutionary technique is used to control the diversity of the population pool through elite and random immigration strategies. Elite immigration ensures that the best solutions identified by the algorithm are retained by maintaining a list of top-scoring solutions, which is updated with each iteration and added to the population in every round. This approach prevents the algorithm from diverging from or forgetting previously identified good solutions. On the other hand, random immigration, involves randomly selecting from known solutions that were not included in the starting population. This strategy is specifically designed to maintain diversity within the population.

Re-rolling: This unique feature, originating from the specifics of SIMONE settings, addresses the challenge of a large part of the search spaces consisting of invalid settings combinations, meaning solutions often fail during mutation or crossover. The re-rolling approach aims to overcome this by repeating the process that generated the failed genome with a new random seed, thereby creating an alternative version of the unsuccessful attempt and attempting to solve it. (I.e. if the product of a crossover - the child - is an invalid combination of settings, then re-rolling will repeat the crossover process with the same parents to gain a sibling that could still be viable) This process can be repeated multiple times in the event of successive failures. Essentially, it functions as a mini-tree search within the Genetic Algorithm, continuing until a valid combination of settings is discovered.

Progress tracking: During a run, we track both the best solution and the median solutions while also comparing them to those from previous runs. This approach enables us to determine which settings perform better and provides insight into the mapping of categorical settings within the reduced MCA space. By observing these mappings, we can identify convergence towards a specific region of the solution space, enhancing our understanding of the algorithm's efficiency and direction.

Hyperparameters: Given the extensive number of features we implemented, there are a large number of hyperparameters available. We list them in Appendix 9.7.

03.3 Results and performance

There were a series of milestones that the development of the BSS aimed to hit:

1: The algorithm is able to run with basic/dummy versions of tooling built in: This was achieved early in the project, and showed that the Genetic Algorithm framework was functional and open for extension with more complex methods.

2: The algorithm is able to consistently improve on the initial solutions generated by the Start Point Finder: Over the course of a BSS run, we track several metrics related to the score of solutions, including the best score achieved in each generation and the median score of the population. These give us an overview of how well the Genetic Algorithm does at improving the population over generations.

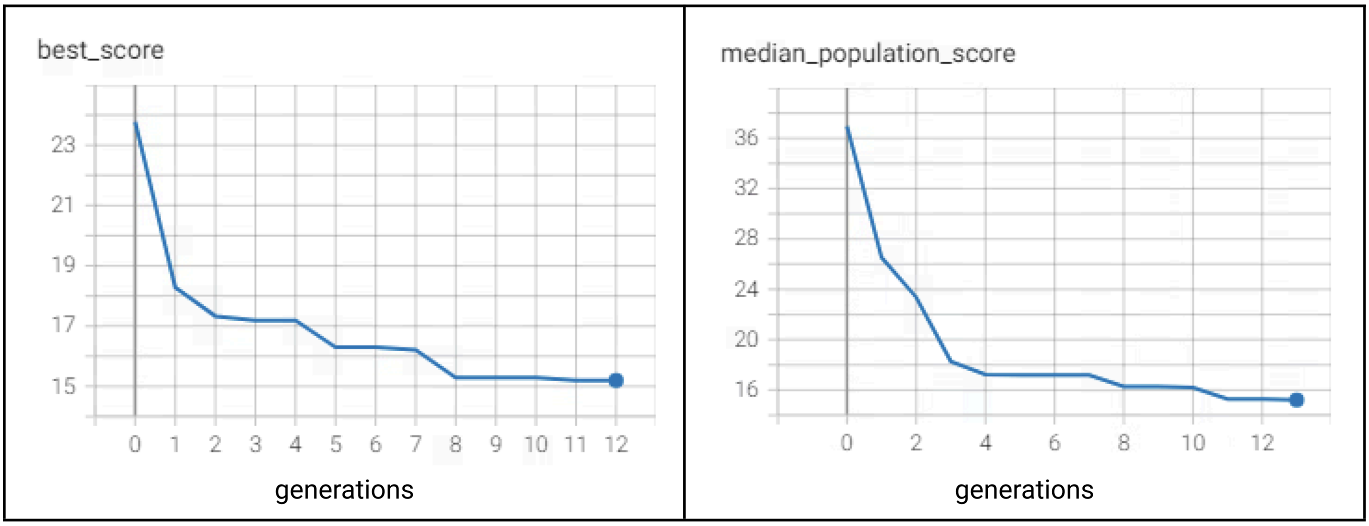


Figure 7: The improvements in the best score (LHS) and the median score of the population (RHS) over the course of a 13 generation Genetic Algorithm run. Lower scores are better. This is a typical profile for a BSS run.

The Start Point Finder returns a set of results which make up the starting population, or generation 0 (Figure 7). We find that the BSS is always able to find a better scoring solution than the best solution returned by the SPF, and can also always improve the median score of the population. This indicates that the BSS is behaving as intended and generating increasingly fit populations over time. However, the solutions tend to plateau around a best score before reaching a fully solved state. This indicates that the solver has reached its limits and can't find a better solution. This could mean that in fact no better solution exists, or that the parameters of the solver are not well calibrated, leading to improper early stopping.

3: The algorithm is able to fully solve a scenario: While this was not achieved, the BSS was able to produce good intermediate results that were similar to the later stages of an analyst's work to solve a scenario. The results are still promising as they show that the BSS can, within the limits that are provided, improve on a solution. Ways to further improve the BSS are discussed in section 3.5.

Case studies:

Case study 1: Best Overall Score:

The best score achieved by the BSS for a test scenario was for a peak day test scenario. It achieves a very close linepack with only three minor supply node violations and no compressor issues. While still not a flawless solution, it demonstrates the BSS's capability to take an unseen test scenario and significantly improve the settings in an attempt to reach a solution. This is a promising result and indicates that a full solution is likely viable with some further modifications to the algorithm.

Limits of the best BSS solution

Assessment	BSS solution
Violated limits	3 pressure limits violated at offtake nodes
Supply/demand balancing	12.67 mismatch
Linepack at target value	1.88 mismatch
Compressor stations	No stations in local bypass, choke or surge

Case study 2: Improvement over SPF

A key reference point for the performance of the BSS is the performance of a solution generated by the SPF (i.e. a solution generated using the settings from previously-solved scenarios with similar supply/demand flows). By comparing the output of the BSS to the best-scoring solution generated using the SPF (where the SPF is used to create the initial population for the BSS), the improvements provided by the Genetic Algorithm can be seen clearly.

Comparison of SPF and BSS solutions for the same scenario.

Assessment	SPF solution	BSS solution
Violated limits	34 pressure limits violated at offtake nodes	2 pressure limits violated at offtake nodes
Supply/demand balancing	17.55 mismatch	8.38 mismatch
Linepack at target value	51.09 mismatch	19.42 mismatch
Compressor stations	1 compressor unit in CHOKE	2 stations in local bypass

By comparing in this way, it can be seen that, although the BSS does not reach a fully solved solution, it significantly improves upon the performance of the SPF. This validates the capability of the Genetic Algorithm in solving SIMONE scenarios, demonstrating that the evolutionary strategy allows the model to discover solution improvements.

Case Study 3: Consistency of results

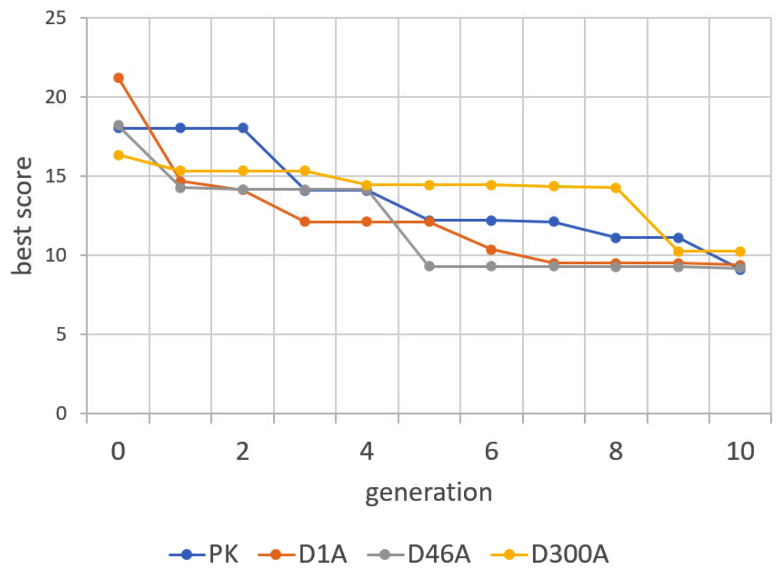


Figure 8: The improvements in the best score of the population over the course of a 10 generation Genetic Algorithm run, shown for four different gas day scenarios across the same gas year. The demand is highest at PK and the earlier gas days, and lowest at the latest gas days. Lower scores indicate a better performance.

The consistency of the performance of the BSS was investigated by running the model against four different gas day scenarios across the same gas year. Given that the demand is highest at the 'peak' (PK) day and the earlier gas days, and demand decreases at later gas days, these results indicate similar model performance across a wide range of demand levels.

03.4 Lessons learned through development of the BSS

The lessons learned from developing the BSS affirm its potential: it has passed the milestone of consistently improving on Start Point Finder results, which represent the current best approach to 'automatically' solve scenarios. This achievement suggests that an automated solver for gas network scenarios is indeed feasible. As part of the BSS development, several key insights have been uncovered that will shape the path forward.

Developing the BSS has involved substantial effort in codifying solution workflows, requiring close collaboration with gas analysts. This indicates that ongoing development will necessitate sustained, tight collaboration with these experts. Beyond the need to establish technical requirements, continued engagement will likely reveal opportunities for improvement as the tool evolves e.g. by encoding pre-existing knowledge about gas network behaviour directly into the Genetic Algorithm.

A crucial takeaway from the project is the need for accurate integration of approved multi-junction configurations. Without this, there is no way for the BSS to reliably produce valid multi-junctions, and it will produce non-viable solutions. Incorporating multi-junctions into a customised Genetic Algorithm enhances search efficiency by eliminating the need to explore every possible combination of valves, compressors, and regulators. It also allows for more sophisticated strategies, such as varying mutation rates in different contexts (e.g. different mutation rates for multi-junction configurations versus assets that are not part of a multi-junction) or implementing staged optimisation processes. This requirement for the integration of approved multi-junction configuration relies on the configurations being translated from the assumptions documentation to a tabular data format by gas analysts; NESO gas analysts have produced these data for the current versions of the approved configurations, but these data will need to be kept up to date with any updates or modifications to the approved configurations.

Another learning from the development of the BSS is the importance of the Start Point Finder as a component of the BSS, as random initialisation of the solver typically yields invalid solutions with no direction for improvement. By ensuring the BSS begins its search within a valid space, the Start Point Finder establishes a high-quality initial population that enhances performance across generations of the Genetic Algorithm.

During the course of this phase of the FastPress project, there was an update to the common model used for scenario modelling. This gave us the opportunity to evaluate how the BSS would need to be modified given changes to the network. It is evident that while the BSS does require updates to remain aligned with changes in the master network, its core design is sufficiently robust and it is unlikely to need fundamental redevelopment for typical common model updates.

The challenges posed by changes to the master network can be divided into two main categories:

- **Changes that will require modifications to components of the BSS in order to provide valid results on the new network.** For instance, recent updates in multi-junctions within the common model would necessitate revisions to the tabular data describing approved configurations, which can be accomplished by simply updating the tabular data in the relevant spreadsheet or CSV format. Similarly, any adjustments to operational limits that are not already incorporated into the master network—such as compressor station flow limits or offtake pressures contingent on national demand—must be mirrored by updating the tabular limit data. However, if the approach to interpreting limits changes, e.g. if a previously-constant head limit on a compressor station now depends on the flow direction through the compressor, the BSS code itself will need modification.
- **Changes that are likely to reduce the performance of the BSS on the new network until updates are made to components of the BSS.** This would materialise as lower-quality final results or increased time to achieve solutions. Changes in network assets, like those in multi-junctions, may reduce the relevance of historical data, perhaps even invalidating it under significant alterations. This reduction

in data quality and quantity affects the Start Point Finder, which forms the foundation of the BSS, leading to lower performance until new data (*i.e.* solved scenarios on the new network) is available. The precise volume of new data required to restore effective BSS performance is not yet clear, but it is likely to show a gradual increase in performance with added data and plateau in performance when the data quality is fully restored.

03.5 Areas for further development

The primary focus in further development of the BSS should be improving the quality of BSS solutions, both in absolute quality of individual results and in consistency of results across scenarios. As it stands, the BSS has generated solutions with relatively good scores (e.g. only showing a few asset-level limit breaches, and has not balanced linepack and supply/demand flows) but it has not yet achieved a fully solved scenario that would pass standard solution audits. There is also an inconsistency in the final scores achieved by the BSS depending on the scenario that is to be solved; it has not yet been determined if this inconsistency is due to genuine differences in the difficulty of solving scenarios or if it is a result of a high sensitivity to the randomness in Genetic Algorithms.

The strategies in consideration for creating these performance improvements include:

- **Tuning of the methods already in use.** This involves running large-scale experiments to determine the optimal 'hyperparameters' (i.e. parameters that affect the way in which the Genetic Algorithm runs) for the BSS.
- **Adding support for live automatic hyperparameter adaptation during BSS runs.** This would allow the algorithm to adapt its strategies in response to the previous algorithm steps e.g. reduce mutation/crossover rates if too many 'invalid' solutions are being generated; increase mutation/crossover rates if the scores are plateauing.
- **Further development of crossover and mutation strategies in collaboration with gas analysts.** This would allow for gas analyst expertise to be translated as pre-existing knowledge into the BSS e.g. by creating mutation and crossover strategies that imitate typical strategies used by gas analysts.
- **Explore 'staged' optimisation.** This would split the 'solving' into stages e.g. optimise the multi-junction configurations first, then tune compressor setpoints, then adjust line pack and supply, and iterate across stages as needed. This has the additional advantage of being more interpretable to analysts as it is more similar to the manual approach and because multiple 'optimised' strategies could be presented as results to the analysts.

In addition to performance improvements, the impact of the BSS would be improved by increasing speed. This could be achieved by integrating parallelisation (discussed further in Section 06 which explores tool structuring) and by exploring different population evolution strategies that track previously-executed scenarios and reduce the number of changes that require re-running scenarios.

For a deployed BSS tool, it will also be necessary to complete the LVC component by building in the exit point pressure cover as discussed in Section 03.2.1 and rigorously evaluating the assumptions built into the LVC in collaboration with gas analysts.

Beyond developing the BSS as-is, it will also be possible to add additional features to the BSS, such as:

- **Solve scenarios subject to user-defined restrictions.** With this feature, gas analysts could provide restrictions such as a given compressor strategy, or a maximum number of compressors, and the BSS will attempt to solve the scenario within those constraints.
- **Additional scoring control** to include more ways to judge the 'quality' of solutions e.g. better scores for solutions that use fewer compressors or use green compressors. It is likely that a feature that rewards using fewer compressors will be a high priority for development of the BSS, as minimising compressor use is a key goal in the gas analyst workflow.

03.6 Summary and conclusion on Base Scenario Solver

The development of the BSS aimed to demonstrate the feasibility of automatically solving static scenarios within a reasonable timeframe. While the model has not yet reached the stage of regularly producing fully solved scenarios, it has shown promising capabilities by consistently improving upon the results provided by the Start Point Finder, which represent the current best approach to 'automatically' solve scenarios. This demonstrates the potential for automated solution discovery using the approaches described. To achieve the objective of generating fully solved results, further enhancements are necessary, as outlined in Section 03.5. These improvements will address aspects such as solution quality and operational consistency, ensuring the BSS reaches its full potential.

From the perspective of running in a reasonable timeframe, the model currently plateaus in solution quality within a timeframe of roughly 2 hours on a VDI without any parallelisation[5]. This is very likely to change, as the BSS has not yet undergone any major re-designs for speed improvements, and the speed could be significantly improved by both general improvement to the model and the other speed improvements discussed. However, the BSS could also require additional time when we build in the required improvements to reliably reach solved scenarios. Given the initial concern that the solution space might be excessively large for any effective exploration (let alone exploration on a timescale that allows for the solver to be helpful as part of gas analyst workflows), the current timescales are encouraging and indicate that the BSS is potentially viable as a tool for analyst workflows.

Thus, while further work is required to refine the BSS and attain fully solved scenarios, the foundation laid by this project suggests promising prospects for future development.

[5] Currently, timings vary depending on the technical set up used to run the model. We have found this to be quicker when the model is run on a laptop, and the implementation of parallelisation will enhance this performance. In future, it is the intention that all processing will be performed in isolated environments on cloud compute instances.

04 Pipe Removal Workstream

04.1 Background and context

The current NESO gas infrastructure is constantly evolving in order to take into account changes in gas demand and the emergence of distribution networks for new energy sources across GB. The current gas network configuration evolves to accommodate future growth, repurposing for the transportation of alternative gas, or decommissioning. One of the ways that the network can adapt is through the removal of pipes. However, since there are a large number of pipes on the network and evaluating their suitability for removal is manual and complex, establishing a robust, data-driven methodology for the identification of pipes for removal is crucial for informed future management of the gas network.

During user research sessions, we observed that the existing workflow for pipe removal involves three main phases:

1. identifying the zone requiring modification
2. selecting a segment / subsection of the network to isolate or remove
3. running simulations in SIMONE to validate the feasibility of the suppression

If the SIMONE simulation (step 3) is unsuccessful, analysts must repeat the process to find an alternative suitable pipe segment for removal. The initial phases of the pipe removal workflow rely on subjective decision-making which could introduce potential bias and inconsistencies among analysts. This section explores how the pipe removal workflow can be optimised and standardised, using information accessible from the gas network that was previously unavailable to analysts. This includes topological information, for example how each pipe and node (supply, offset or other subtypes) are interrelated, as well as functional information which captures how the pipes operate and flow in relation to each other.

During our exploratory data analysis, we computed several metrics to evaluate the significance of each pipe within the gas network, leading to the development of a pipe ranking algorithm grounded in multiple topological features. In light of these new metrics and the fact that analysts generally remove groups of pipes, we have developed two methodologies for automatically grouping pipes. The first one relies solely on the topology of the gas network while the second incorporates gas flow dynamics. These methodologies provide richer insights to analysts about the importance of groups of pipes in the network and enable a data-driven approach to the initial steps of the pipe removal workflow.

04.2 Exploratory data analysis (EDA) and technical approach

04.2.1 Static pipe removal

First, we developed a data converter to load gas network data from a SIMONE network file (TE.xml) and transform it into a Python graph. This process involved parsing the file's content and cleaning it to construct a graph representation suitable for subsequent data analysis. In this context, a *graph*[6] comprises nodes that represent entities, and edges connecting between them, as depicted in Figure 9(A).

[6] Wikipedia Contributors (2019). *Graph theory*. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Graph_theory.

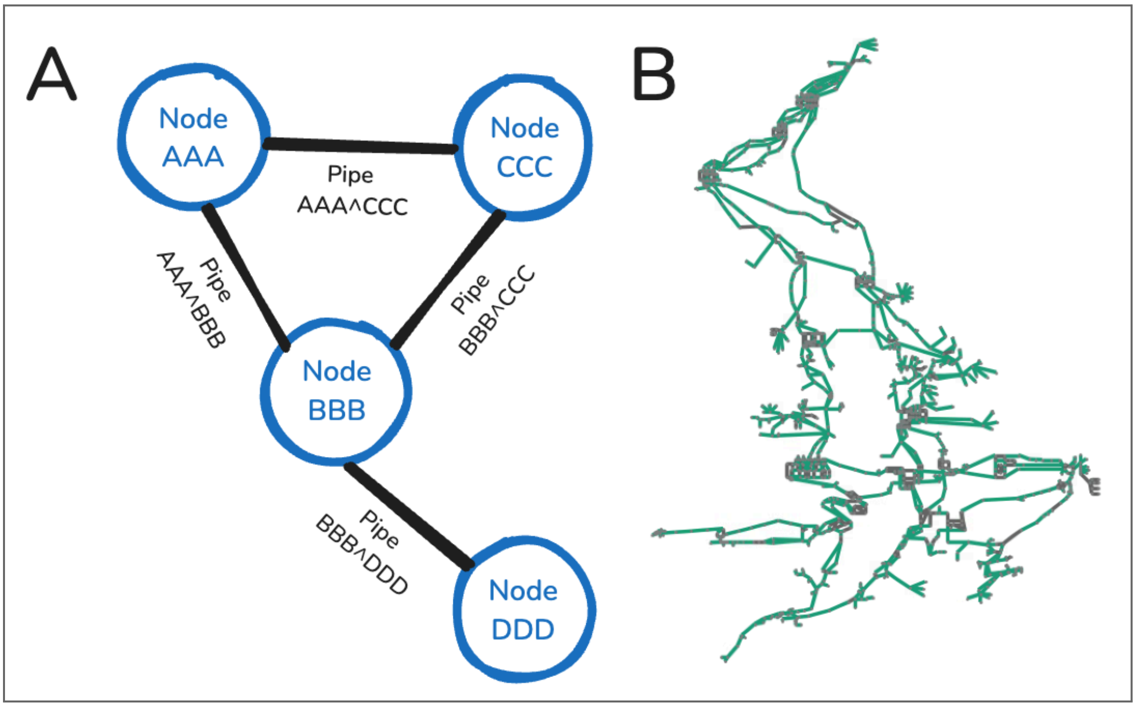


Figure 9(A): Schematic representation of a graph with four nodes (AAA,...) connected by 'edges' - here pipes (AAA^CCC,...). Figure 9(B): Graph of the gas network loaded from the TE.xml file. Nodes are represented at their geographical position. Edges displayed in green are "pipes" while grey edges are other edge types (valves, compressor, resistors).

In the gas network graph (Figure 9(B)), the edges are formed by compressor stations, pipes, valves, control valves, joints, non-return valves, and resistors. These edges connect different types of nodes within the network, such as supply, offset, and other nodes. This graph-based representation enables the computation of various topological metrics, providing insights into the significance of each pipe within the network. To achieve this, we employed a combination of three measures: bridge edges, edge betweenness centrality, and the minimum distance to selected nodes or edges. Each edge had multiple attributes associated with it such as the diameter, length or roughness of the pipe. Similarly, the attributes of each node were their name, the identifier of the pipes they connect to and a list of properties indicating if they are part of supply or offset nodes[7].

The importance of a pipe within a network is influenced by multiple factors, making it difficult to be adequately captured by a single metric. Therefore, we applied various quantification techniques to provide a multi-faceted perspective on the significance of each pipe to the user.

- **Bridges:** since the gas network must remain connected at all points to ensure gas distribution to all downstream nodes, we initially focused on the detection of bridge pipes — those whose removal would split the network into two separate sub-networks (Figure 10 (A)). In the gas network, bridge pipes were located at the periphery of the network in short stem or long pipe segments (Figure 10 (B)).

- **Betweenness Centrality:** Another classical metric to measure the importance of an edge in an undirected graph is the *edge betweenness centrality*[8]. This measure is calculated from the fraction of shortest path from all the possible pairs of nodes in the network traversing the current edge. Thus a pipe traversed multiple times will likely have a high importance in the network (Figure 10 (D)).

- **Distance from supply/offset nodes or bridges edges**: Lastly, a topological distance that can inform us about the importance of a pipe in the gas network is its distance from subsets of nodes or edges in the network. Of particular interest is the distance between each supply/offset node or bridge pipe (Figure 10 (C)).

[7] Note that the TE.xml file provides the information if a node is a supply node directly, however, we imported the identity of the offset nodes from a single scenario to perform the following EDA. This offset node list is hardcoded in the current version of the code but suitable adaptations have been made to ease their loading from an ancillary file in the future.
[8] Brandes, U. (2008). On variants of shortest-path betweenness centrality and their generic computation. Social Networks, 30(2), pp.136–145. doi:https://doi.org/10.1016/j.socnet.2007.11.001.

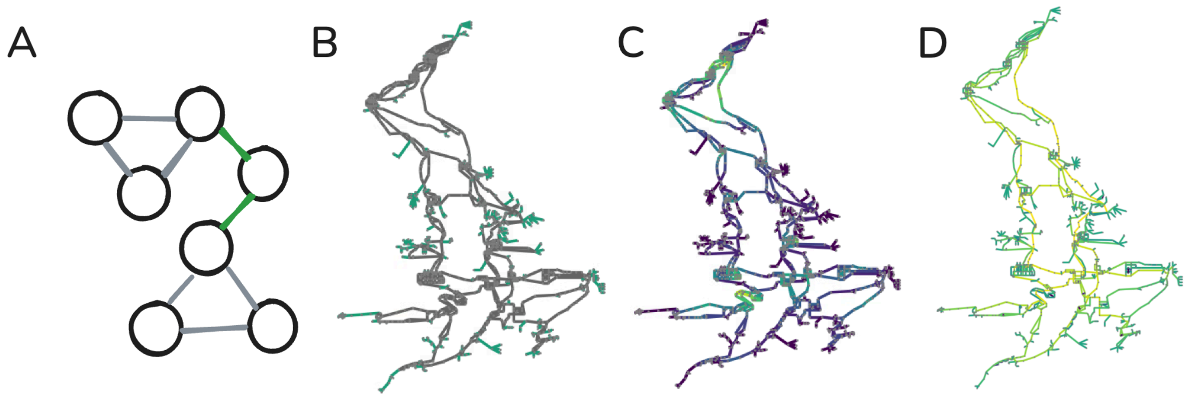


Figure 10 Bridge - Edge importance metrics on the gas network:
10(A) Schematics of bridge edges in a graph. Bridge edges (green) that if removed will break the network in two sub-networks. Non-bridge edges (grey) will leave the network connected in all points if removed.
10(B) Bridge pipes displayed in green pipes are mostly distributed at the periphery of the network.
10(C) Visualisation of the distance from bridge pipes, with edges coloured from blue (low distance) to yellow (high distance) to illustrate distance from bridge pipes.
10(D) Distribution of edge betweenness centrality within the gas network. Edges are coloured from blue (low centrality) to yellow (high centrality) to indicate variations in centrality value.

04.2.2 Segment identification

From previous user research we learnt that in gas network analysis, pipes are often segmented to simplify complex systems for modelling and management. Analysts seem to include or exclude pipes in a segment based on their past experience and personal expert knowledge about the gas network. This approach is thus prone to human errors and inter-individual variations. Furthermore, it does not allow us to leverage the wealth of information that the configuration of the network offers. To tackle this, we have identified two possible approaches - topological and functional - for defining segments. Both of these approaches have different strengths and challenges but give us complementary sets of information that will assist analysts to choose pipe segments as candidates to remove from the network. These segments will then be useful to either aggregate or re-compute the features from the previous section to help analysts make data driven choices of pipe segments to remove and homogenise the definition of pipe segments across analysts.

04.2.2.1) Topological approach for pipe segment identification

The following methods exclusively utilise topological strategies to partition the network into smaller segments. Topological information encompasses the structural details that define the interconnections between nodes via edges, *i.e.* pipes. This includes the arrangement and connectivity of nodes illustrating the connections and pathways among them, and key metrics like node degree (the number of pipes attached to a node). Using this information, we can effectively segment the network based on its overall structural configuration.

Pure Topological Method: This method defines a segment as a group of contiguous pipes between a bifurcation and another bifurcation or end node (as shown in Figure 11). This method is relatively simple and only requires a full traversal of the gas network graph. Our algorithm has been optimised using a Breadth First Search (BFS) in order to traverse the gas network efficiently. While being very simple conceptually this method lacks sophistication to properly segment highly interconnected areas. As demonstrated in Figure 11, this method typically results in numerous, overly small segments due to minor bifurcations.

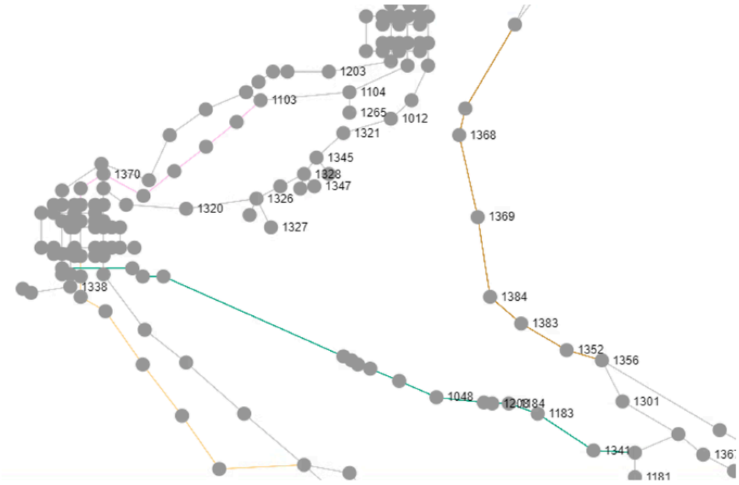


Figure 11 - Demonstrating a pure topological method for edge grouping. This is a subsection of the graph network showing representative segments (green, yellow, pink and brown) computed using the pure topological method. Segments start and end at bifurcations or end-nodes. Note the shortening of the pink segment because of a pipe at node 1103 links it with another subsection of the graph.

Spectral Clustering Method: To overcome the limitations of the previous method, we implemented spectral clustering, a widely-used unsupervised clustering technique for graph data. This approach uses network similarities to group nodes based on their connectivity and density, which are informative for understanding the structure of the graph. By focusing on nodes rather than edges, spectral clustering enables the assignment of similar nodes to the same label, with connected edges inheriting this label—resulting in coherent groupings, such as pipe segments in our network. See the details of the method in Appendix 09.4.

This method allows a greater cohesion of the clustering and provides more "natural" segments, especially in intricate subsections of the gas network (Figure 12). However this method would require further optimisation as it tends to over-segment long, continuous pipes, requiring further research into graph weighting or post-processing adjustments.

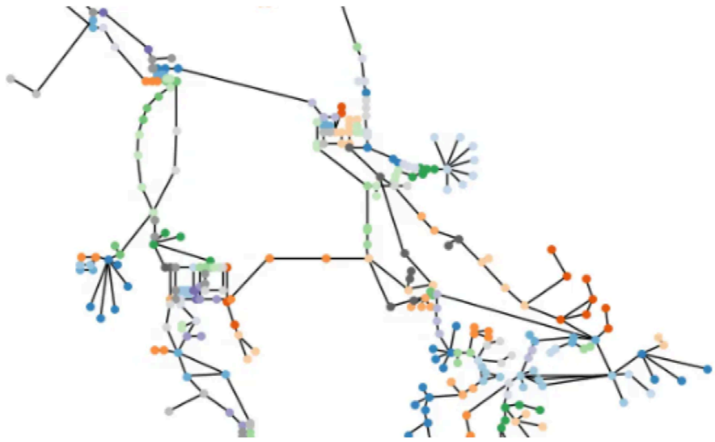


Figure 12 - Spectral clustering method for edge grouping: A subsection of the network showing the clustering from the spectral clustering method. A better segmentation of the network can be observed notably in densely interconnected areas.

04.2.2.2) Flow variance approach

How pipes are connected does not directly reflect how the gas flows into the network; the configuration of the network and different patterns of gas supply or demand subtly changes the way gas flows. Gas flow patterns can help identify which group of pipes operate together and which pipes are redundant or dependent on each other. Historical data from successful SIMONE runs can be leveraged to inform us on how the flow of gas covaries without running any new simulations.

To illustrate how co-variation of the gas flowing in pipes can be informative, let's take the example of five segments of pipes running in parallel in Figure 13. A topological method will identify five segments (coloured pipes in Figure 13(A). However, an approach based on variation of the gas flow, here the Principal Component Analysis (PCA) clustering on covariance matrix detailed in the next section, will uncover that four of those segments consistently flow together (green pipes in Figure 13(B)). An increase of gas influx in one of those pipes is mirrored in the other ones at the exception of the fifth pipe. Depending on the situation, an analyst could thus choose to remove one of the redundant pipes (green) or to understand why the fifth pipe is not in the group of pipes. If the flow of this pipe is low or rarely used, it will mean that most of the distribution load is performed by the green pipe group and the grey segment is thus a good candidate for removal. As depicted in this example, this approach can gain insights beyond the scope of topology and on a larger geographical scale or on pipes that are not directly connected.

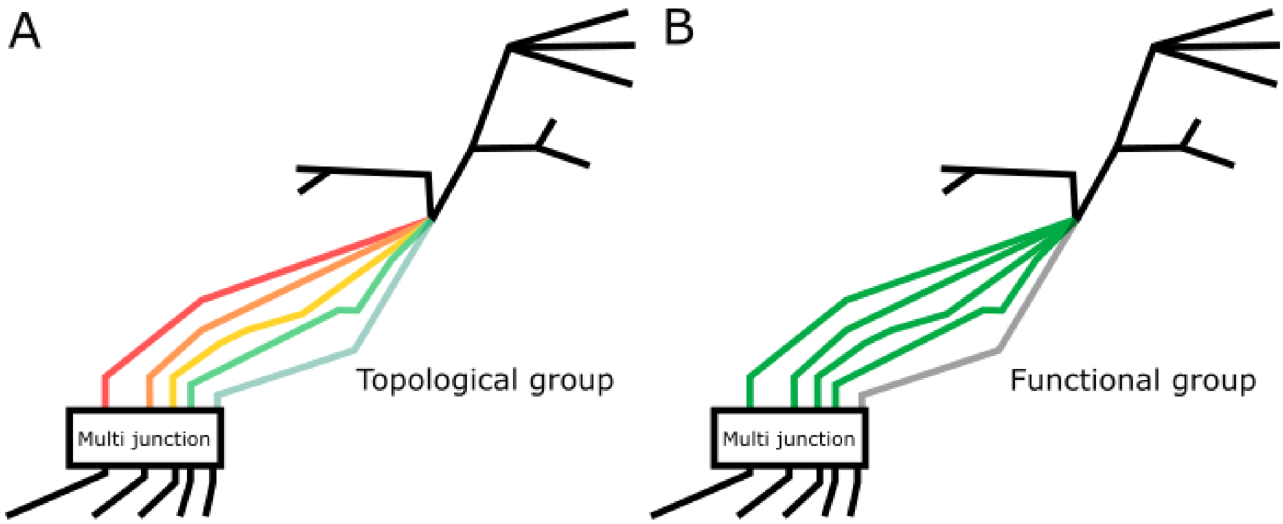


Figure 13 - Schematic of topology A vs functional B group identification.

An initial exploration of the data flow revealed that most pipe flows follow a normal or bimodal distribution (Figure 14(A)). Notably, the variance and mean of the pipe flow are significantly correlated (Figure 14(B)) $\rho = 0.78$, $p < 0.01$). Applying PCA to this dataset indicates that approximately 95% of the variance can be explained by around 47 components (Figure 15), suggesting that groups of pipes exhibit significant co-flow within the network, making them suitable for clustering. Based on these insights, we implemented two distinct procedures to generate functional groups of pipes.

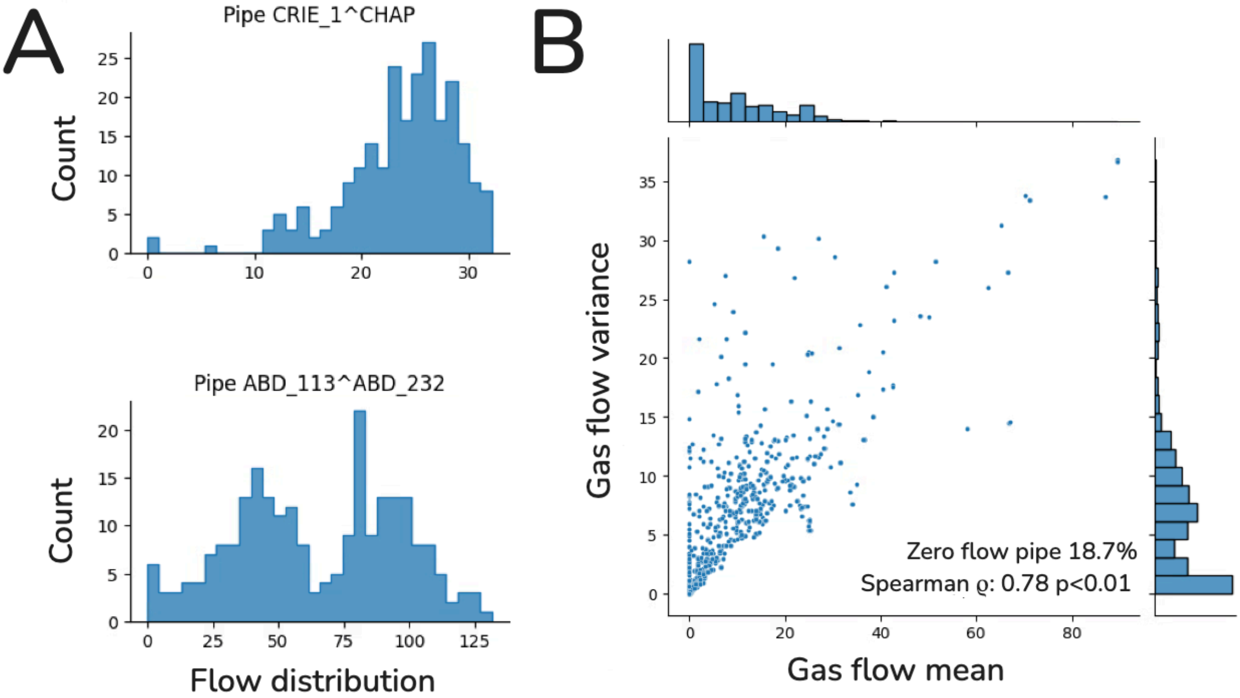


Figure 14 - Flow variance descriptive statistics: A) Two representative examples of distribution of flow pipes. The pipe CRIE_1^CHAP (top) shows a unimodal distribution while ABD_113^ABD_232 (bottom) displays a bimodal distribution. This suggests that flows in pipes have a stable regime. B) Mean flow and variance are log-normally distributed with a majority of pipes having a low flow while only a few exhibit a high flow.

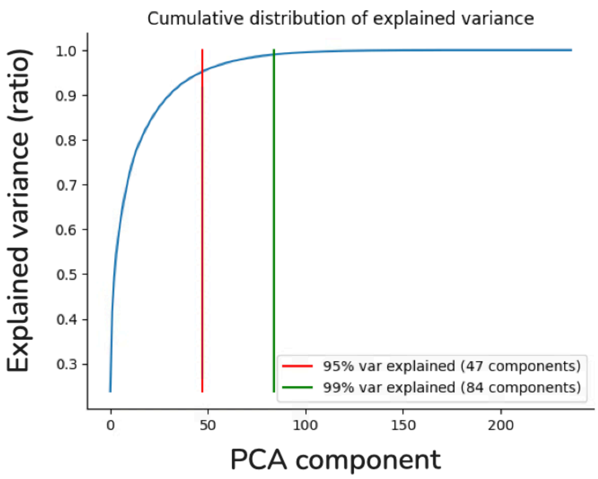


Figure 15 - Distribution of variance explained from a PCA features of the flow data in all pipes. The first 47 and 84 components explain 95% and 99% of the variance, respectively, indicating that the flow data from approximately 1300 pipes in the network can be reduced into functional groups.

Hierarchical Clustering: This method uses the covariance between all pairs of pipes in the network to group them. Covariance serves as a metric for assessing the similarity in the variation between two time series, in our case, the gas flows within a pair of pipes. A high covariance indicates that two pipes exhibit synchronised flow patterns, whereas a covariance close to zero suggests their flows are independent. By calculating the covariance for all possible pipe pairs, we can identify potential groups where pipes demonstrate correlated flow. To facilitate this, a hierarchical clustering tree is employed to systematically merge similar columns of the covariance matrix into similar clusters. This process involves iteratively fusing groups of pipes based on the maximum distance between their members, reflecting the strength of their flow relationship. Ultimately, the tree is "cut" at a designated threshold to form distinct clusters, effectively grouping pipes with similar flow patterns.

This method is simple and easily identifies interpretable groups of pipes co-flowing together. However, the challenge of this method is to determine a robust distance threshold to cut the clustering tree and thus determine the number and identity of pipes in the clusters. We thus developed a complementary method providing a quantitative way to define the number of clusters.

Principal Component Analysis clustering: Another approach to identify groups of pipes co-flowing is to directly leverage the information extracted during the dimensionality reduction. Performing PCA on the normalised flow time series will give us the distribution of eigenvalues but also feature vectors indicating the strength of participation of each pipe's activity in this feature. This method is inspired from a system neuroscience methodology[9] identifying assemblies of neurons that are co-active to collectively support the formation of memory. In our case, neurons and assemblies will be substituted by pipes and functional groups of pipes.

First, we normalised the flow time series to focus on flow variance for grouping (akin to hierarchical clustering). We then used PCA to select components explaining 95% of the data, ensuring that the number of groups is statistically significant and includes only components with substantial explanatory power. By analysing the feature vectors of each component, we determined which pipes belong to each group. Pipes with high weights in a component indicate strong involvement in that group, with PCA's orthogonal structure typically assigning each pipe to only one group. Additionally, we developed an alternative technique using Independent Component Analysis (ICA) to allow for more flexible grouping when pipe groups partially overlap. This method allowed to group co-active pipes while having a robust estimate of the number of functional pipe segments in the network.

04.3 Results and performance

In the previous section, we developed and detailed several topological metrics, which we utilised for an initial selection of candidate pipes for removal. Candidate pipes are defined as edges that are not bridge, have a low centrality and a short distance from either offset or supply nodes ($d < 2$). Using these strict criteria, our algorithm identified 169 out of 1365 network edges as potential candidates (Figure 16).

Further analysis using the dimensionality reduction technique t-SNE, applied to various metrics (length, roughness, distance to offtake or supply nodes, and centrality), revealed network clusters of groups of pipes with similar characteristics. This clustering is evident from the agglomerations of points in Figure 17 (see also Appendix 09.6 for detailed distributions of each variable in the low-dimensional space). The candidate pipes are overrepresented in some clusters of the low-dimensional space while being virtually absent from others (Figure 17). This shows that candidate pipes do not form a single, distinct cluster in the reduced dimensional space, highlighting the challenge of defining candidate pipes for removal using a single parameter. However, this represents an interesting direction for future research, for example exploring the development of alternative versions of the classifier, tailoring the thresholds depending on the use case of the analysts (*eg*: target node distant to supply, close to bridge…). The thresholds we used in the development of the proof of concept of this classifier are empirical or parametric. They are based on the observed distribution of the data during EDA and might not directly reflect the one that an analyst would use. Thus, a validation and improvement of the classifier should be informed by the knowledge of the characteristics of pipes removed in the past and user research sessions, as discussed in Section 07.2.

[9] Vítor Lopes-dos-Santos, Ribeiro, S. and Adriano (2013). Detecting cell assemblies in large neuronal populations. Journal of Neuroscience Methods, 220(2), pp.149–166. doi:https://doi.org/10.1016/j.jneumeth.2013.04.010.

Figure 16 - Candidate pipes for removal are highlighted within the graph network - where our initial iteration identifies "redundant segments" between bifurcation in the network.

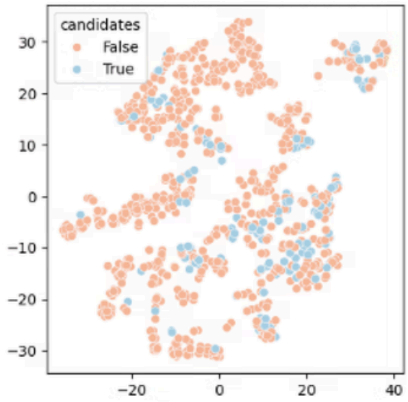


Figure 17 - t-SNE projections of the pipe metrics are clustered in a multidimensional space (see [Appendix](#) 9.6 for more details). Our selection process identifies a group of pipes that are primarily distributed from the bottom right of the reduced dimension space.

Historical data from successful SIMONE runs can reveal which segments of pipes operate together and which are redundant. This insight led us to develop methods for grouping pipe segments that align with the way analysts naturally identify them, addressing issues of over segmentation in segment identification of the topological method. Preliminary development of these methods suggested their potential to automatically select segments in a manner similar to analysts.

The first method, hierarchical clustering, identified 21 segments exhibiting similar coactivity profiles (Figure 18(A)). This approach offers promising insights by uncovering functional segments that extend beyond the one identified by the purely topological method. The identified segments form loops and bifurcations that are geographically close and likely to flow coherently, as well as segments that are geographically distant (Figure 18(B)). This highlights groups of pipes that, on average, operate at the same time and in the same way. In some instances, we observed that distant segments were likely caused by noise—characterised by small, sparse, and distant pipe segments—due to a limited sample size (data not shown, but demonstrated in the notebook).

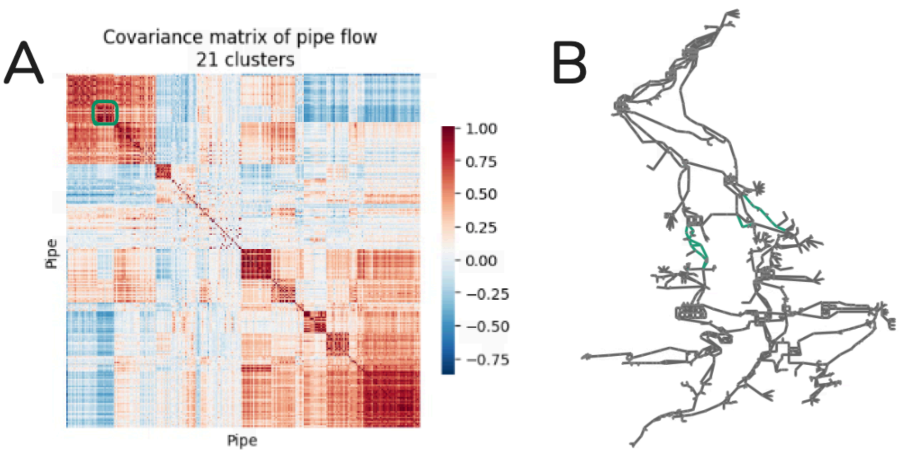


Figure 18 - Example of pipe segments computed from hierarchical clustering:
A) Covariance matrix of all the possible pairs of pipes in the network. This matrix shows that groups of pipes exhibit strong co-activity (red squares on the diagonal). Pipes belonging to segment 3 are circled in green

B) Distribution of the pipes in segment #3 (green) identified with the hierarchical clustering algorithm shows that the segments are coherent and sometimes distant geographically.

Additionally, PCA clustering produced more numerous, defined and smaller segments compared to the hierarchical method, resulting in approximately 40 clusters. These segments were more numerous, smaller and could disentangle the activity of pipes that are close geographically but having different patterns of activity (Figure 19). However, this method also produced some distant pipe segments attributed to noise, likely due to unreliable flow variance estimations. In the next section, we propose mitigation techniques and outline subsequent steps to enhance our findings from this section.

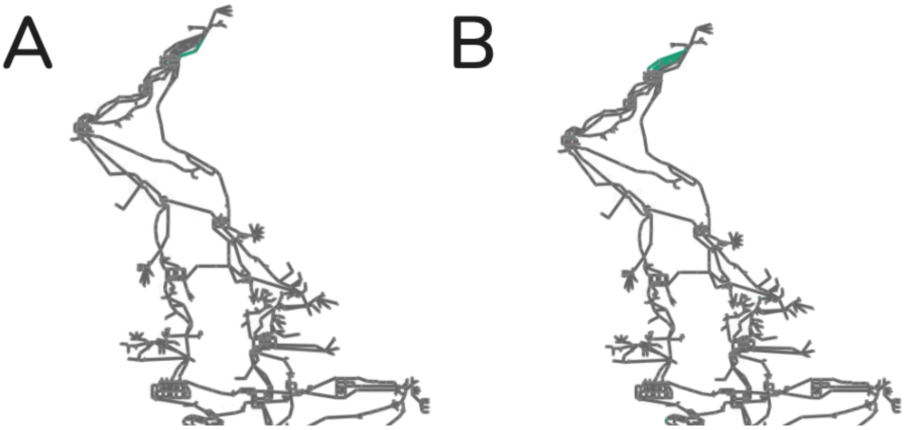


*Figure 19 - Example of pipe segments computed from PCA clustering: Distribution of pipe segments within two distinct clusters as identified by PCA clustering, highlighted in green. The method successfully identifies segments of pipes that are co-active (**A**) while distinguishing the activity of neighbouring pipes. Although the pipes in segments **A** and **B** are geographically close, they are disentangled by the method due to differences in their activity pattern.*

04.4 PoC development, learnings and feedback

Our exploratory data analysis revealed a rich dataset that was previously unexplored, enabling data-driven planning for pipe removal operations. Using metrics such as edge centrality, distance to offset nodes, and bridge identification, we narrowed down the candidate pipes from over a thousand to approximately a few hundred - an ~80% reduction in scope. The selection criteria can be easily modified to accommodate different objectives and contexts for pipe removal (detailed in Section 04.2.1).

However, some factors limited our approach. For some of the functional analyses, our results would have been more robust with a larger number of sample points, additional valid scenarios, and a broader sampling of potential network configurations. This expanded dataset would allow for a finer determination of the gas flow directions, enabling us to create a more precise directed graph. This, in turn, would enhance our ability to detect if multiple pipes operate in parallel or series, helping to identify pipe co-dependencies (series) and redundancies (parallel).

Having access to the identity and characteristics of the pipes previously removed from the network could also allow us to validate and refine our current methodology with information of what motivated the removal of a pipe from the network in the past.

In addition to pipe removal recommendations, the flow variance analysis method we developed could independently provide valuable insights into the network and the characteristics of gas flow through its pipes. Successful scenarios generated from the Basic Scenario Solver could contribute to a feedback loop, where validated network configurations enhance our understanding of flow dynamics of the network in its current configuration, thereby providing a coarse-scale approximation of network behaviour under various conditions.

04.5 Next steps

In this section we detailed an exploratory data analysis on the pipe removal recommendation system, revealing that a wealth of insights about the network and its flow dynamic can be derived solely from the gas network's topology and flow data.

To extend the current results, we propose three primary directions:

- **Data over extended or more precise timescale to validate our methods:** Having data on a longer or finer time scale would enhance the robustness of methods such as flow variance analysis and pipe segmentation.
- **Integrated User Interface Development:** The development of a User Interface (UI) that consolidates the metrics and grouping techniques we've developed. This UI will facilitate data manipulation and access, thereby streamlining the data-driven approach to pipe removal. User research has highlighted that analysts face challenges in simulating scenarios with removed pipe segments—such as adding valves or setting flow to zero. Furthermore, manually adding isolated pipe segment information (adding the gas offset of a group of pipe) is identified as a tedious and error-prone task. Integrating our existing code from the exploratory data analysis could automate these processes and reduce the risk of errors. We outline these enhancements in detail in the wireframe Section 05.5.5 and provide a comprehensive plan in Section 07.3.
- **Integration with Base Scenario Solver and Capability Limit Finder:** Another significant enhancement involves integrating the pipe removal tool with the Base Scenario Solver and Capability Limit Finder. This integration aims to assess whether a scenario can be resolved without a particular pipe segment, recommended by the methodologies described in this section. By doing so, we aim to streamline the pipe removal process for analysts while uncovering new insights about the resilience of the network that would allow a safer management of the gas network. Further elaboration of this initiative is presented in the Section 07.2 where we also discuss the development of system resiliency features.

05 Wireframes and Front-End

05.1 Background and context

Wireframes are skeletal layouts and interface design of software tools. Prior to investing into building the interactive components of software, wireframes are a great tool to quickly gather user feedback for usability testing. It provides assurance that the tool and interfaces to be built in the future will bring value to the users and meet their requirements. This has certainly been true throughout the Alpha phase of the project, with multiple rounds of user testing providing valuable insight into how real prospective end-users are expecting to use the tool in practice. Additionally, the development of a conceptual front-end gives a holistic context to the more technical elements of the tool, rather than just isolated processes.

05.2 Initial development

The three identified functional requirements concerning the tool's front-end are to allow an analyst to:

- Solve static base scenarios with the tool
- Submit more than one scenario in a single submission
- View a list of their own submissions, associated metadata and relevant results

With the knowledge that automated scenario solving would likely take some time for processing to complete, a system relying on an analyst to actively wait for results would not be deemed acceptable, especially when the solution is required to scale to support multiple concurrent users. A decoupled design addresses this issue, wherein processing jobs could be submitted to a backing server via an interactive input form. Following a new submission, an analyst may completely disconnect from the tool (e.g. shut down their laptop), and check in on job progress through a separate interface later. To enable this, the other half of the decoupled design consists of a single, searchable table containing information on all submitted jobs along with their processing state, and the ability to view details for a single job after selecting the row. A high-level diagram summarising this approach is given below:

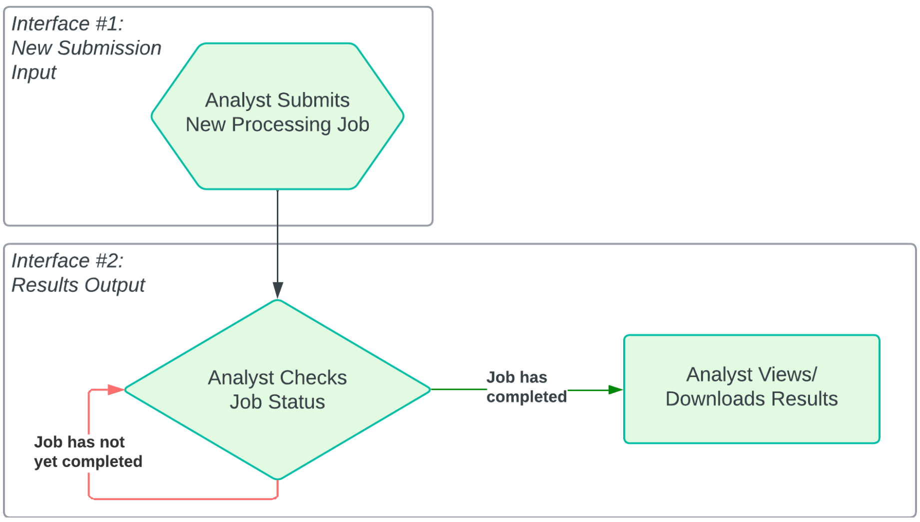


Figure 20 - Showing the workflow of required interfaces

Initially, pipe removal was considered to be a process which should be kept separate from the rest of the user experience. For this reason, the initial design included a separate page for users to submit a network for pipe removal recommendations, and no means to view the results of a previous run. This was later changed as a result of user testing.

A set of initial wireframes were designed to incorporate these ideas through four different pages:

- Input form (with popups showing input validation results)
- Table of past scenario runs
- Details of a single scenario
- Pipe removal recommendations

Screenshots of these designs are given below (see Appendix 09.4.1 for larger versions):

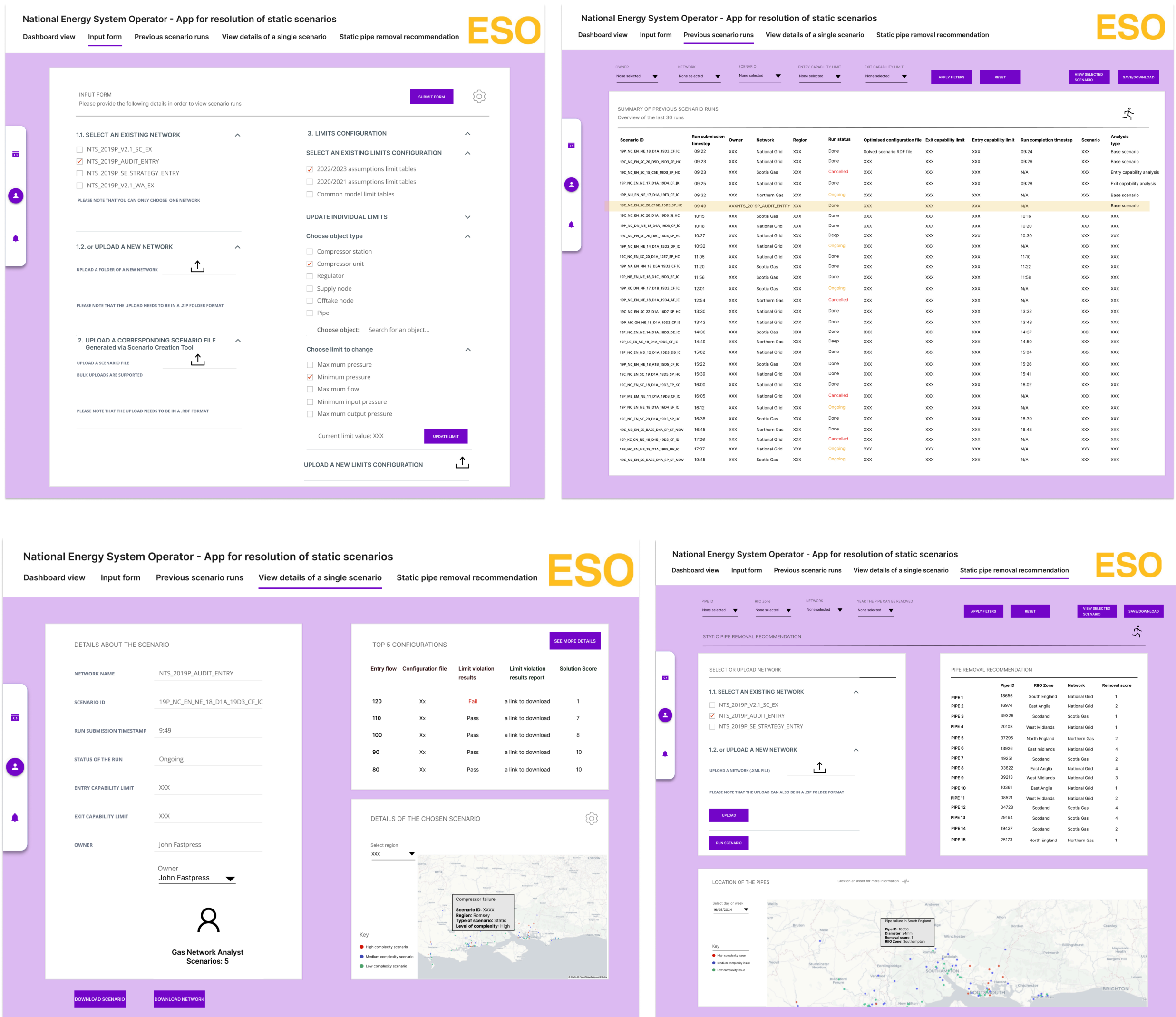


Figure 21 - The initial wireframes that were developed in the early stage of the Alpha phase

05.3 User testing

Over the course of the Alpha phase, two rounds of user testing were performed, both providing valuable insight into steering the continual improvement of the wireframe designs. A summary of the main improvements to the initial designs is given here.

05.3.1 Changes as a result of the first round of user testing

The first user testing session focused primarily on the contents of the input form, revealing multiple confusing elements of the initial design, including some misunderstanding regarding terminology and certain underlying technical components.

The general look and feel of the interface design significantly changed to match modern web design best practices, *e.g.* opting for “drag and drop” file upload zones rather than simple buttons, and searchable lists rather than large dropdowns. Additionally, the direction of user interaction was adjusted to flow from top to bottom, removing the need to scroll back and forth to fill out different sections.

One element of the initial design that was not clear was the ability for users to upload multiple scenario files during single submission, resulting in multiple individual submissions from filling out a single input form. This was resolved by switching to an interactive list of uploaded scenario files, with the ability to remove individual uploads if anything had been selected accidentally.

The function to give submissions a custom name was also added as a result of analyst feedback, as the purpose of a submission would not always be obvious from the contents of the network name and scenario filename alone.

A core component of the input form is the ability to configure the limits used within the underlying Limit Violation Checker process. In the initial design, this functionality was presented as a two-stage process, where an analyst would first search for an object before configuring one of its related limits. Considering the large quantity of limits possible to reconfigure, the design was changed to instead use a single, searchable key/value table of `[<object name>.<limit name>, <limit value>]` pairs. This design combines the ability to simply view limits with the ability to amend them, and follows an existing naming convention familiar to analysts.

Finally, a minor terminology change was implemented, renaming "Region" to "RIIO Zone" across the corresponding page designs[10]. Screenshots of the updated input form and runs table page wireframes are given below (see Appendix 09.4.2 for larger versions):

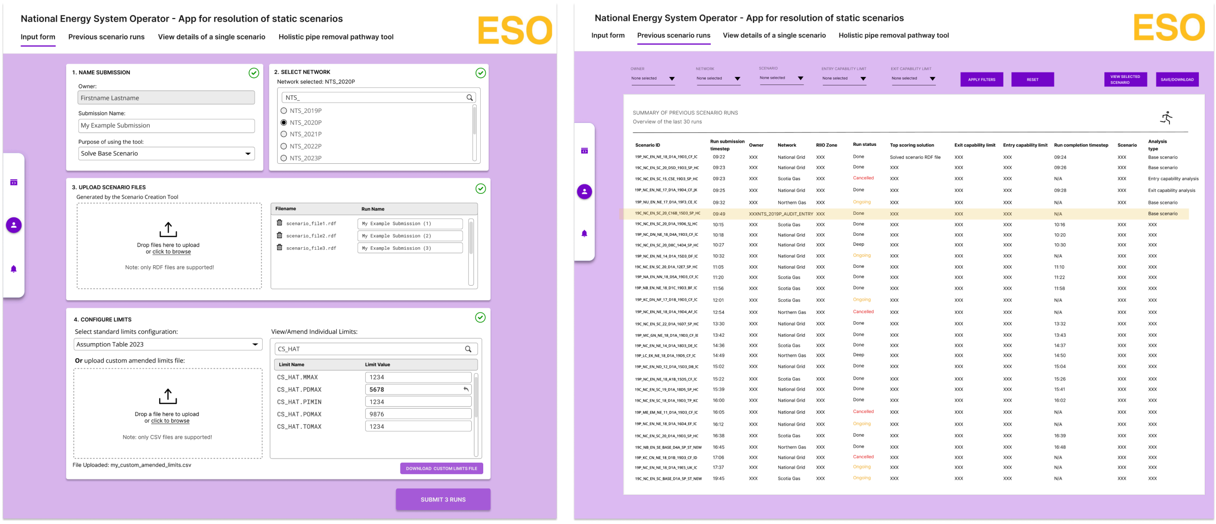


Figure 22 - The second iteration of the wireframes which were adjusted as per the details above.

05.3.2 Further changes as a result of the second round of user testing

After presenting the changes made during the first round of testing, the second user testing session focused primarily around discussion of the other wireframe pages, with only minor changes to the input form being implemented as a result of feedback given at this point.

One of the most notable visual updates to the wireframes implemented during this period was an implementation of the colours introduced in the public NESO rebrand, moving to a darker shade of purple throughout, and using the new NESO logo.

The use of the word "run" throughout the design was a source of some confusion, and so the decision was made to instead use "experiment" to refer to each individual tool submission (i.e. each row in the table page). This allowed for a clear distinction to be made between individual SIMONE "runs" and FastPress-specific "experiments" processed by the tool, which themselves will require multiple SIMONE runs.

---

[10] In the context of energy and utilities, especially in the UK, **RIIO zones** refer to areas under the regulatory framework known as **RIIO** (Revenue = Incentives + Innovation + Outputs). This framework was established by Ofgem - and has been adopted by NESO. It is essentially just an identifier of a specific network region.

The page containing the table of submitted experiments received a major visual update here, switching to a system which can display the full history of all submissions via pagination buttons, rather than just the ~30 results displayed in the previous design. Additionally, the filtering functionality was also built directly into the table, rather than being a separate form. Finally, the columns included in the table were refined, displaying only those which would be useful to uniquely identify the experiment, and processing status information, reserving any data related to experiment outputs for the individual results pages.

During the previous round of testing, the decision was made to remove the ability to upload network files, as it was assumed that changes made to input network data appeared to be infrequent. This decision was reverted during this round of testing, as it was clear from feedback that this functionality was indeed important.

As mentioned previously, the initial wireframe designs considered pipe removal recommendation analyses to be fully isolated from the rest of the tool. To avoid needless confusion here, the decision was made to instead treat the removal recommendations process as its own type of experiment. This means that the submission of pipe removal experiments would be possible from the main input form page, and information regarding the progress/output of such experiments would be accessible from the experiments table page.

A large overhaul of the per-experiment results page designs was also undertaken during this period, to address analyst feedback. Previously, these designs afforded too much space to metadata (e.g. the name of the user who submitted the experiment, the name of the input scenario file, etc.), as opposed to the actual analysis results. This was rectified by switching to a consistent layout for all results pages, where a tabular view of analysis output fills the majority of the screen space, alongside an accompanying interactive map for an alternative output display.

05.4 Final outcome

This section presents the final front-end wireframe designs developed over the course of the Alpha phase. Each design is accompanied by text describing the intended function of each component.

05.4.1 Input form

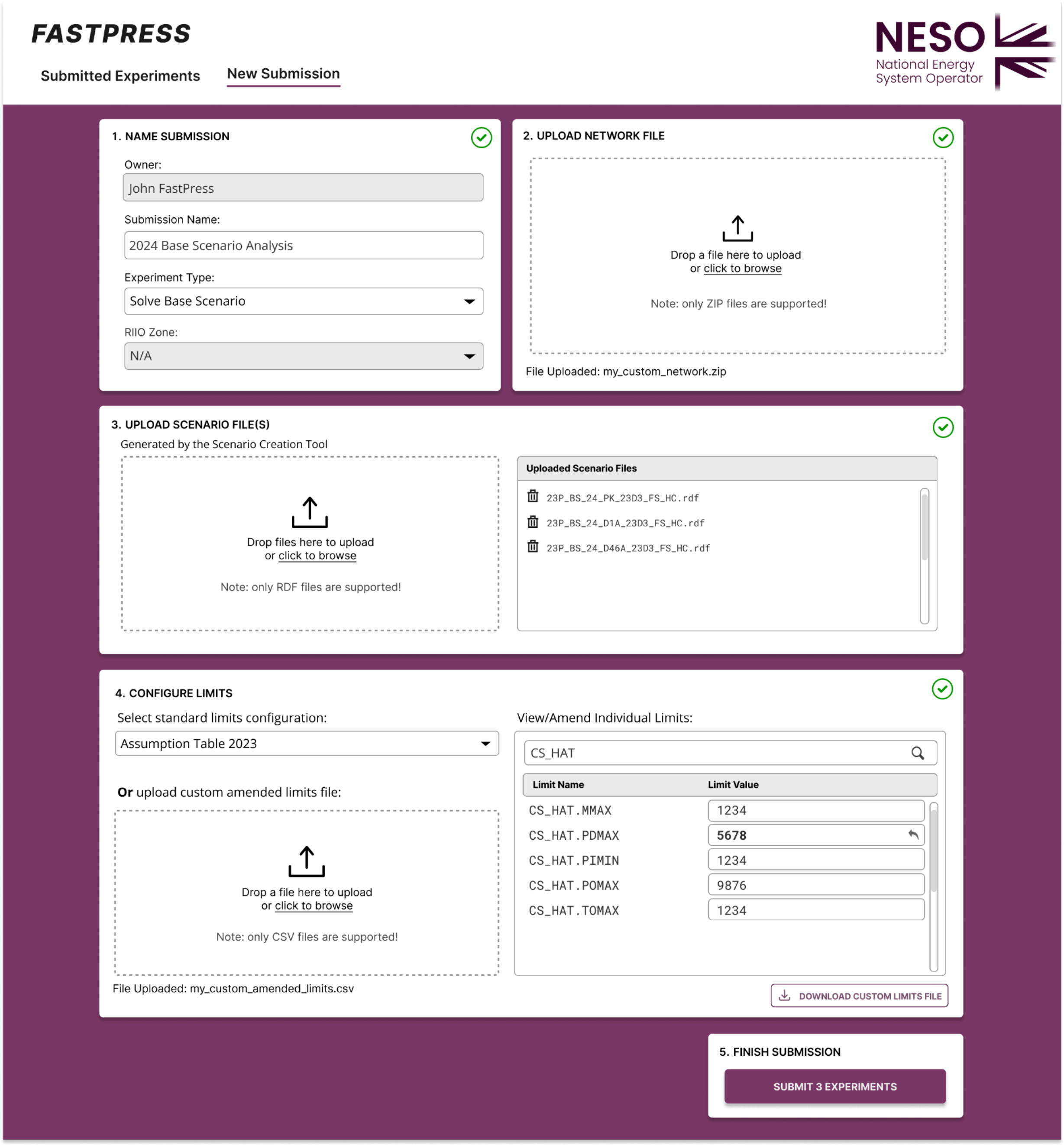


Figure 23 - The final wireframe for the input form

To configure the submission of new experiments, the input form process has been split into four distinct sections:

a. Name Submission

Here, the user can view and verify the user name associated with the submission, before giving the submission a descriptive name, allowing it to be easily identified at a later date. Next, a dropdown input component can be used to select the type of experiment that the user intends to run, with a choice of the the following options:

- *Solve Base Scenario*
- *Entry Capability Analysis*
- *Exit Capability Analysis*
- *Pipe Removal Analysis*

The final dropdown displayed here is only applicable to and editable when an entry/exit capability analysis is selected, and allows the user to select the desired RIIO zone for the analysis to operate within.

b. Upload Network File

This area contains a simple file drop-zone, for the user to upload a single zipped network file. Only zip files are permitted, preventing the accidental selection of incompatible file types.

As with all of the other file upload components throughout the wireframes, this would allow for a drag-and-drop style of file selection, alongside being clickable, revealing a standard "file upload" dialog box.

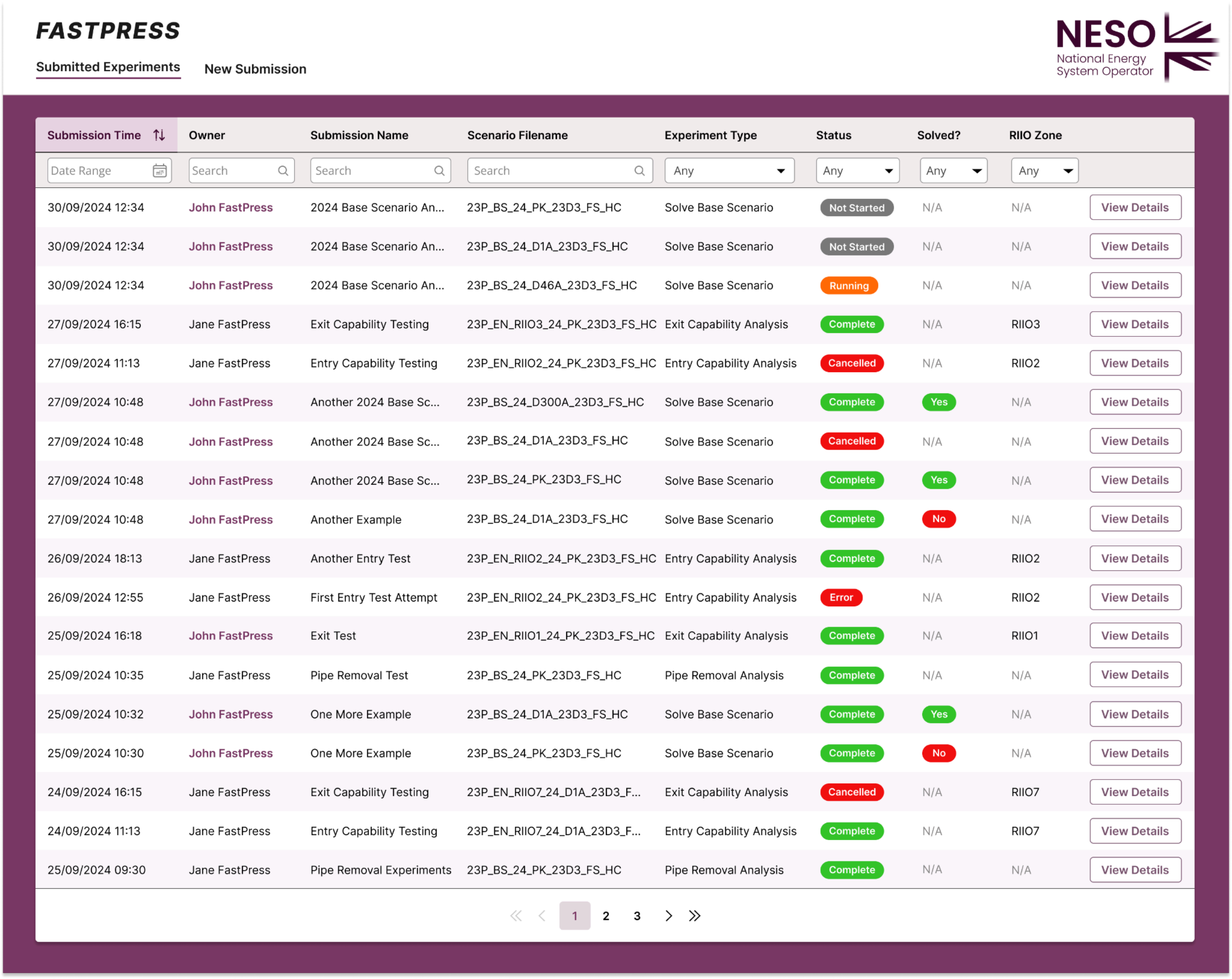
c. Upload Scenario File(s)

The next stage of the form allows the user to upload one or more scenario files. Each uploaded file is then viewable in the neighbouring list, where individual items can be removed if uploaded by accident. Being able to upload multiple files here gives the user the ability to easily submit multiple experiments where only the scenario differs, without having to configure the network or limits over and over again.

d. Configure Limits

The final stage concerns the configuration of limit values utilised by the Limit Violation Checker. First, the user may select whether to use a common preset configuration using the dropdown, or to instead upload a "custom limits file". After choosing either, all limits contained within the configuration can be searched and viewed using the neighbouring table. Amendments can be made to individual values for further customisation. These amendments may then be downloaded as a "custom limits file", for faster configuration of similar future submissions.

05.4.2 Table of submitted experiments



Submission Time	Owner	Submission Name	Scenario Filename	Experiment Type	Status	Solved?	RIIO Zone	
Date Range	Search	Search	Search	Any	Any	Any	Any	
30/09/2024 12:34	John FastPress	2024 Base Scenario An...	23P_BS_24_PK_23D3_FS_HC	Solve Base Scenario	Not Started	N/A	N/A	View Details
30/09/2024 12:34	John FastPress	2024 Base Scenario An...	23P_BS_24_D1A_23D3_FS_HC	Solve Base Scenario	Not Started	N/A	N/A	View Details
30/09/2024 12:34	John FastPress	2024 Base Scenario An...	23P_BS_24_D46A_23D3_FS_HC	Solve Base Scenario	Running	N/A	N/A	View Details
27/09/2024 16:15	Jane FastPress	Exit Capability Testing	23P_EN_RIIO3_24_PK_23D3_FS_HC	Exit Capability Analysis	Complete	N/A	RIIO3	View Details
27/09/2024 11:13	Jane FastPress	Entry Capability Testing	23P_EN_RIIO2_24_PK_23D3_FS_HC	Entry Capability Analysis	Cancelled	N/A	RIIO2	View Details
27/09/2024 10:48	John FastPress	Another 2024 Base Sc...	23P_BS_24_D300A_23D3_FS_HC	Solve Base Scenario	Complete	Yes	N/A	View Details
27/09/2024 10:48	John FastPress	Another 2024 Base Sc...	23P_BS_24_D1A_23D3_FS_HC	Solve Base Scenario	Cancelled	N/A	N/A	View Details
27/09/2024 10:48	John FastPress	Another 2024 Base Sc...	23P_BS_24_PK_23D3_FS_HC	Solve Base Scenario	Complete	Yes	N/A	View Details
27/09/2024 10:48	John FastPress	Another Example	23P_BS_24_D1A_23D3_FS_HC	Solve Base Scenario	Complete	No	N/A	View Details
26/09/2024 18:13	Jane FastPress	Another Entry Test	23P_EN_RIIO2_24_PK_23D3_FS_HC	Entry Capability Analysis	Complete	N/A	RIIO2	View Details
26/09/2024 12:55	Jane FastPress	First Entry Test Attempt	23P_EN_RIIO2_24_PK_23D3_FS_HC	Entry Capability Analysis	Error	N/A	RIIO2	View Details
25/09/2024 16:18	John FastPress	Exit Test	23P_EN_RIIO1_24_PK_23D3_FS_HC	Exit Capability Analysis	Complete	N/A	RIIO1	View Details
25/09/2024 10:35	Jane FastPress	Pipe Removal Test	23P_BS_24_PK_23D3_FS_HC	Pipe Removal Analysis	Complete	N/A	N/A	View Details
25/09/2024 10:32	John FastPress	One More Example	23P_BS_24_D1A_23D3_FS_HC	Solve Base Scenario	Complete	Yes	N/A	View Details
25/09/2024 10:30	John FastPress	One More Example	23P_BS_24_PK_23D3_FS_HC	Solve Base Scenario	Complete	No	N/A	View Details
24/09/2024 16:15	Jane FastPress	Exit Capability Testing	23P_EN_RIIO7_24_D1A_23D3_F...	Exit Capability Analysis	Cancelled	N/A	RIIO7	View Details
24/09/2024 11:13	Jane FastPress	Entry Capability Testing	23P_EN_RIIO7_24_D1A_23D3_F...	Entry Capability Analysis	Complete	N/A	RIIO7	View Details
25/09/2024 09:30	Jane FastPress	Pipe Removal Experiments	23P_BS_24_PK_23D3_FS_HC	Pipe Removal Analysis	Complete	N/A	N/A	View Details

« ‹ 1 2 3 › »

Figure 24 - The final wireframe for the table of submitted experiments

This interface contains a single, full-page table, containing one row for every submitted experiment. Each column of the table is filterable, reducing the displayed results to a subset matching specific search criteria. The “Status” column clearly displays the progress of each experiment, making it much easier to keep track of which jobs have completed, or if any problems have arisen during processing.

The intention here is for all past experiments from all members of the authenticated user’s team to be present, to enable straightforward collaboration within teams working on the same overall project, but filters can be applied to show only experiments submitted by the signed-in user themselves.

Each row contains a “View Details” button, which when clicked would open the individual experiment output page corresponding to the specific experiment.

05.4.3 Individual experiment output (solving the base scenario)

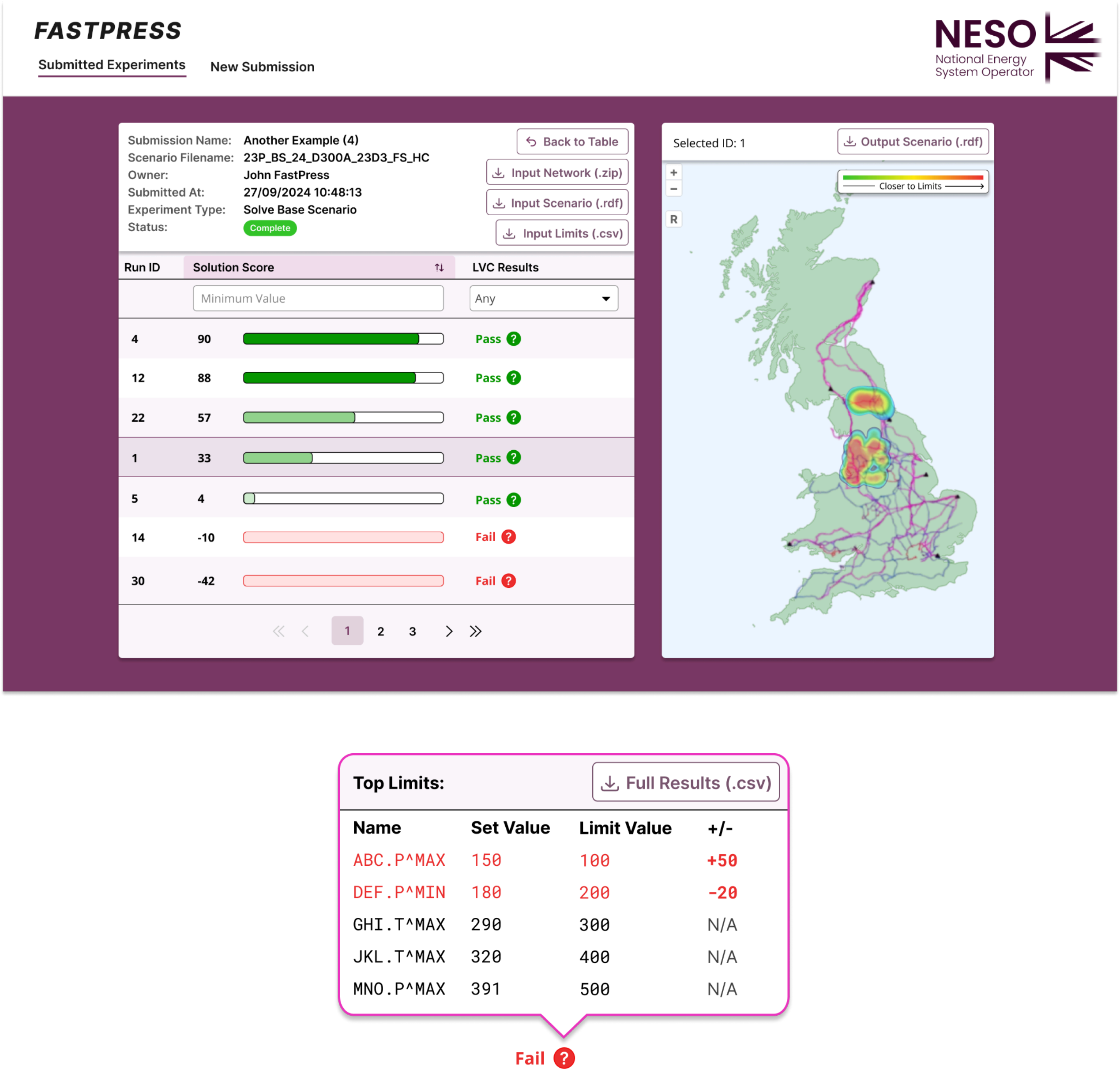


Figure 25 - The final wireframe for the individual experiment output and a zoom on the pop-up block that would appear if the user clicked on a certain Run ID within the results list.

Each experiment output wireframe contains a similar-looking metadata block at the top of the page, describing the input parameters used to create the experiment, alongside the experiment processing status, and buttons allowing for the originally supplied input data to be downloaded.

For Base Scenario Solver output, a single table summarises the results, with one row for each underlying solution (i.e. each distinct SIMONE execution). An associated score value is provided for each run, scaled such that larger numbers indicate better solutions, with everything greater than zero being a valid solution. Additionally the results of the Limit Violation Checker can be viewed by hovering over the corresponding pass/fail value. The revealed pop-up contains a quick view of assets which have exceeded their limits, and the amount by which they have exceeded them. This is shown in the second image given above.

On the right hand side of the wireframe is an interactive map view, displaying a geographical visualisation of the selected solution, where assets within the network are coloured depending on how close they are to their limits. When zoomed out far enough, coloured regions are combined into a high-level heatmap, allowing analysts to quickly identify important areas within the found solution.

05.4.4 Individual experiment output (entry/exit capability analysis)

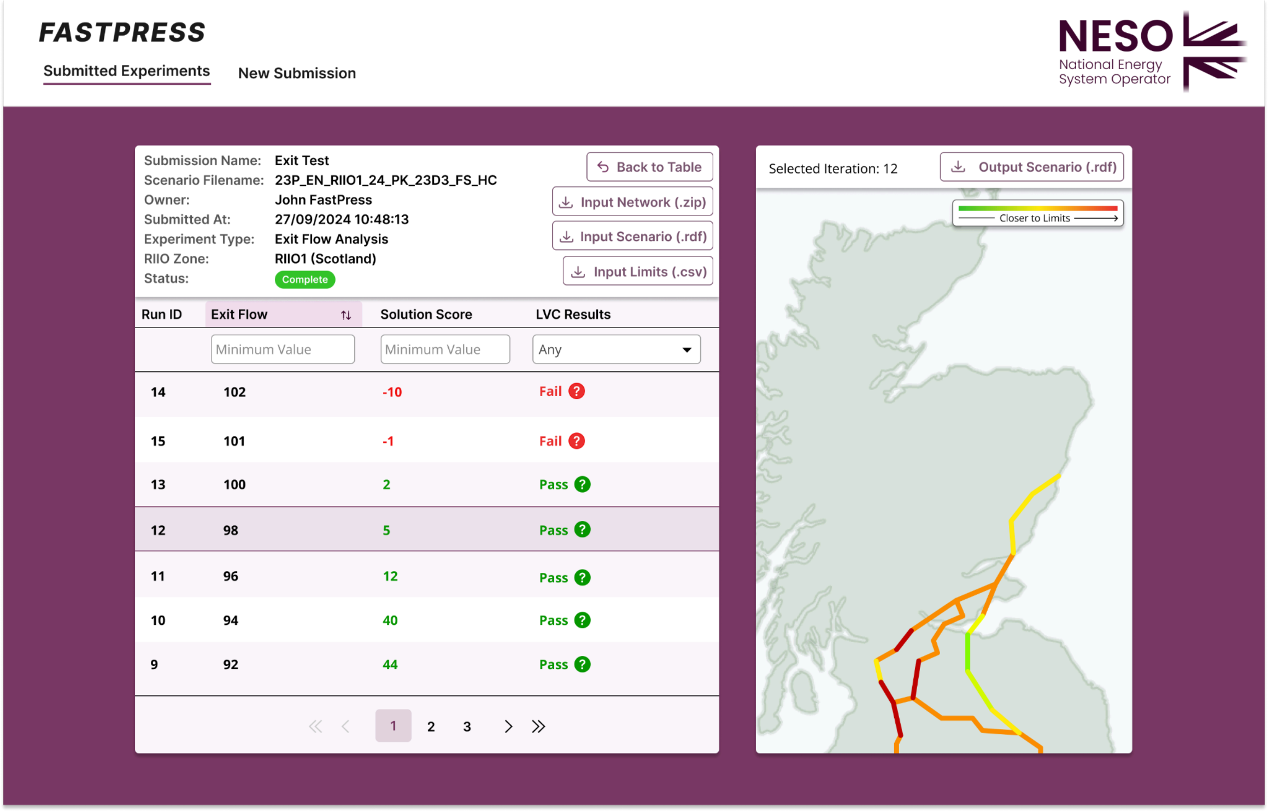


Figure 26 - The final wireframe for the individual experiment output when entry/exit capability analysis is incorporated

Whilst very similar to the results given for Base Scenario Solving, for entry/exit flow analyses the focus of the designed interface is to display the found flow threshold. As before, each row in the results table here corresponds to a distinct SIMONE execution. The boundary in which the solution's score flips from positive to negative clearly displays the identified threshold.

Once again, selecting an individual solution populates the interactive map view with coloured assets. In this view, a zoomed view is shown, corresponding to the configured RIIO zone in which the analysis has been performed.

05.4.5 Individual experiment output (pipe removal analysis)

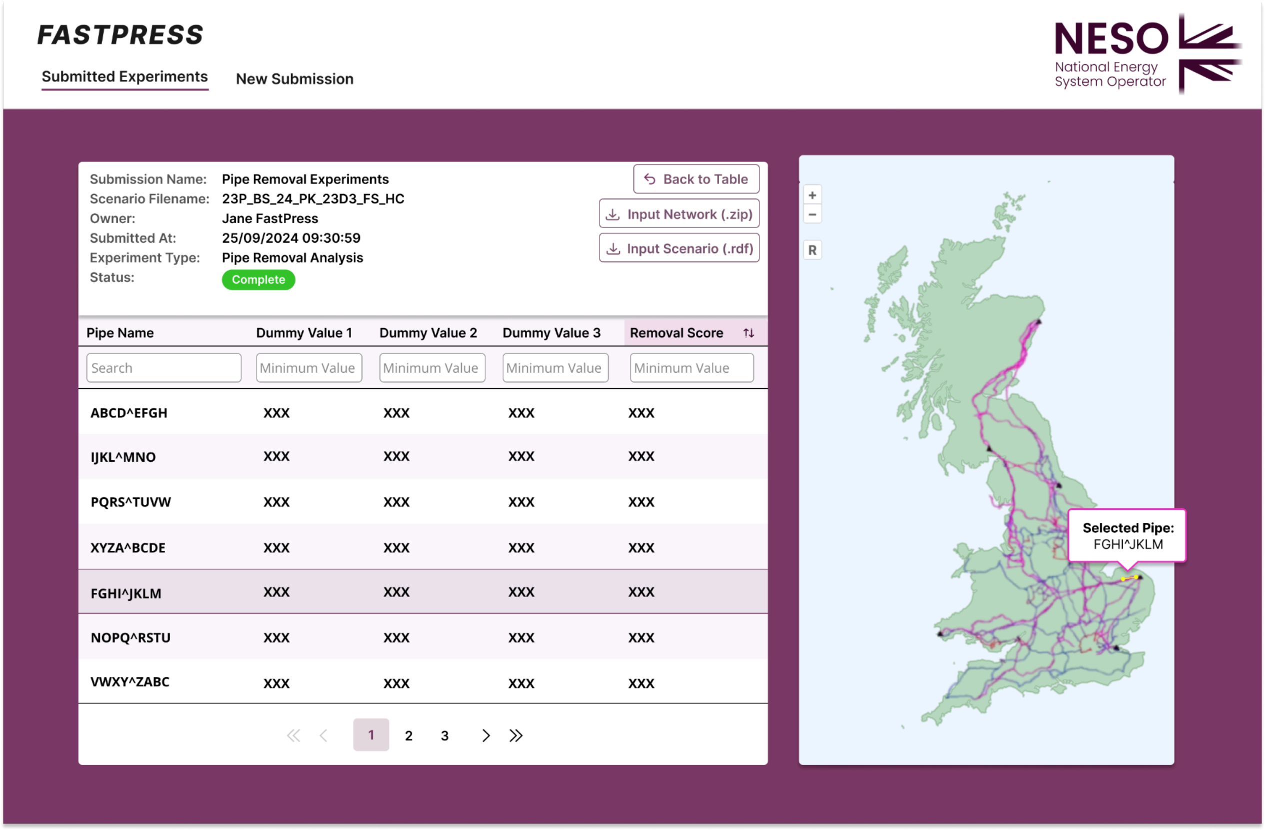


Figure 27 - The final wireframe for the individual experiment output of pipe removal analysis, with a pop up box highlighting on the map what the pipe name you have clicked on in the table corresponds to.

The output page for pipe removal recommendation analysis differs from the previously shown designs, in that each row in the data table equates to a single pipe within the network. The corresponding values within the row are intended to contain quantitative values produced by the underlying recommendation process. To reflect the experimental nature of the pipe removal process as it currently stands, some columns are labelled as “Dummy Values”, with the intention being to replace these with intermediate output from the pipe removal process following further development. An overall “Removal Score” is also provided, summarising the recommendation of removal for each individual pipe.

One final difference in this page is that the interactive map component simply displays the geographic location of a selected pipe within the overall network. Whilst not shown in this design, an additional intention is to also display any inter-pipe relationships discovered during the analysis here, graphically highlighting any pipes on the map view which are grouped together by the metric detailed in the pipe segmentation section.

05.5 Next steps

To progress from conceptual wireframe designs to an interactive, production-ready front-end, a variety of key implementation decisions must be made. Firstly, prioritising the ease of accessibility of such an interface is essential, and has been addressed by targeting a purely web-based deployment, where users access the tool through the web browser already installed on their machine.

The state of modern web development is constantly evolving, with React[11] being the leading JavaScript/TypeScript framework for efficient user interface development. Leveraging React is a sensible choice due to its powerful component-based architecture, which allows for efficient scaling and easy long-term maintenance. Whilst more simple UI solutions exist, such higher-level alternatives often allow no way to create the bespoke interface components outlined in the wireframes, like interactive maps or efficient tables containing a very large number of rows. By also incorporating advanced third-party form management

[11] https://react.dev/

libraries (like React Hook Form[12]), we can streamline the development of the complex input form, including aspects like input validation, another key component of the wireframe design.

Additionally, adopting TypeScript and adhering to current best practices in web tooling will not only enhance code quality but also further facilitate long-term maintainability.

[12] https://www.react-hook-form.com/

06 Tool Structuring

06.1 Background

The development of the FastPress features was not only impacted by user research, timelines and the different choices of technical approach, but also the working environment that we were using for development. This section highlights the limitations and workarounds imposed by the current setup, and indicates what could be changed in future workstreams to enable better, faster and simpler development.

The majority of the work was done on Windows Virtual Desktop Infrastructure (VDI) machines using logins provided by NESO. The VDIs are Windows environments that have SIMONE installed on them. While they are sufficient for running SIMONE, these are not machines dedicated to AI tool development, i.e. they do not have large volumes of compute or memory. While this did not pose an issue for the majority of development, it is not an optimal environment for large scale runs, as there are many Windows processes that take up a large amount of compute, and the VDI hardware can become a limiting factor. This may become an issue when developing features that require higher volumes of runs. The VDIs are also subject to regular shutdowns, which interrupt long running processes, making it harder to develop solutions.

06.2 Multithreading / Parallelisation

The topic we have deemed "parallelisation" concerns the simultaneous execution of SIMONE simulations, running two or more distinct scenarios at the same time. We wanted to explore parallelisation as implementing it can drastically reduce the time required to process large datasets or complex calculations, for example those being processed as part of the BSS. Whilst this could theoretically be achieved by utilising multiple completely unique compute instances, each processing one scenario at a time, this approach would not scale well in terms of cost and increased maintenance burden. Additionally, the ability to execute SIMONE simultaneously on one machine neatly aligns with the Base Scenario Solver process implemented during this phase of the project. Within a single generation of the underlying Genetic Algorithm, each member of the population is evaluated independently from one another, and so will instantly benefit from parallel execution. If possible, this approach would significantly reduce the overall processing time of a single Base Scenario Solver experiment.

Multithreading is a programming technique that allows multiple threads (smaller, independently executable parts of a program) to run concurrently within a single process. Each thread can handle a different task, enabling parts of a program to operate in parallel, which can lead to more efficient use of resources and faster execution, especially on multicore processors.

Some early testing has been performed during the Alpha phase to assess the potential of using multithreading to achieve single-machine parallelisation of SIMONE. We encountered some reasonably large issues when executing this which made it particularly time-consuming to resolve, namely:

- A single API client connection to SIMONE cannot execute multiple scenarios at once, meaning multiple clients are necessary
- Each API client must authenticate with the SIMONE licence server independently, consuming its own licence
- The authentication process takes a considerable amount of time relative to the time required for SIMONE to execute a single scenario

To resolve these issues, we developed an approach which first instantiates a shared pool of re-usable SIMONE API client connections, where licence authentication occurs only once on thread initialisation. An efficient, thread-safe mechanism provides locked, per-thread access to an available client instance when one becomes available. As an example, consider a processing job relying on four threads, attempting to execute 16 SIMONE scenarios in total. Each of the four threads repeatedly acquires an available API connection, executes the next scenario, then releases control of the API connection back to the shared pool. Ultimately, each thread will likely execute four scenarios each, with no per-thread downtime. This will consume four SIMONE licences for the duration of the parallel processing job.

The following graphs show rough timing results for the repeated execution of scenarios using SIMONE, as the number of "workers" (threads) used increases. Note that results here have been collected from tests performed on Azure VDI instances, which may not provide a representative view of the multithreaded performance of a deployed compute instance. However, despite this, the results indicate the clear benefit of multithreading to reduce overall processing time.

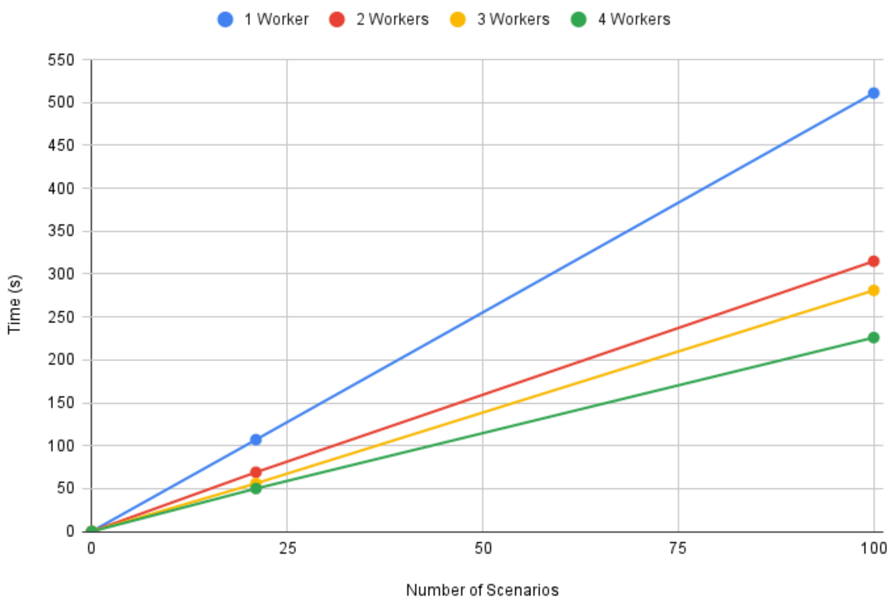


Figure 28 - The impact of thread (worker) count on the overall SIMONE execution time of a set of scenarios

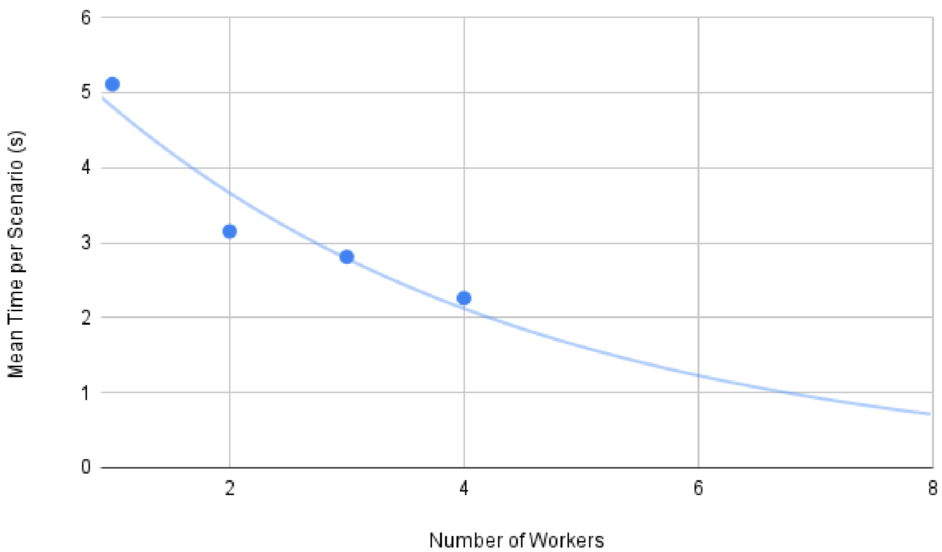


Figure 29 - The impact of increasing the number of threads (workers) on mean SIMONE scenario execution time

06.3 Technical set up

In addition to some standard workflow and access issues, the following technical limitations associated with using SIMONE were discovered and the corresponding workarounds were required:

- **The SIMONE version does not include a Python API.** This required developing a Python wrapper around the C++ API library. This links in with the next point on 32-bit implementation, as the wrapper design was also influenced by that. This would be resolved by moving to a newer version of SIMONE as version 6.37 and above includes a Python API (currently using version 6.35).

- **The SIMONE installation is 32-bit.** This requires that any interaction with the SIMONE API comes from a 32-bit tool. The initial strategy used for this was to work entirely in 32-bit Python, but this was not a sustainable solution due to many Python libraries removing support for 32-bit systems due to the majority of users working only with 64-bit systems. The current workaround is using the Python package MSL-LoadLib[13], which wraps the 32-bit C++ API library and uses Inter Process Communication (IPC) to allow interactions from 64-bit Python. A 64-bit version of SIMONE is available and is covered by the current licence; switching to 64-bit would likely support a more efficient and reliable FastPress tool.

- **The SIMONE licence only covers Windows installations, not Linux.** This affects deployment more significantly than development environments, but it meant that development was carried out on Virtual Desktop Infrastructure (VDIs) rather than running in AzureML due to the limited support for Windows environments in AzureML. There were additional minor limitations throughout the development of the codebase from working on a Windows environment (e.g. fewer tools available to track code memory usage, code timeout functionality).

- **Interacting with SIMONE either requires the tool to run in the same environment as a SIMONE installation or a SIMONE licence that supports a remote API.** Given that running a server to host a remote API requires significant maintenance overhead and that the current SIMONE licence does not cover remote API use, the tool must run in the same environment as a SIMONE installation. For Alpha phase, this meant that development had to be carried out either in VDIs or on laptops rather than in separate cloud environments. It also motivated the investigation of automated installation of SIMONE during the Alpha phase, which will be required to correctly set up the environments for a future cloud deployment of the FastPress tool.

[13] https://msl-loadlib.readthedocs.io/en/stable/index.html

07 Possible future phases of work

07.1 Background

We have identified two primary follow-on phases that could progress the project further, herein referred to as Alpha+ and Beta. For clarity, these are not mutually exclusive options and could progress in parallel if desired.

The Alpha+ path emphasises continued exploration and innovation, building on the work conducted in Alpha and exploring additional features that were identified in the roadmap at the end of Discovery (as above in Figure 1, Section 02.2). This path focuses significantly on expanding the pipe removal workstream, aiming to identify novel methodologies to exploit available data and add in additional functionality which could enhance the effectiveness of the pipe removal recommender, as well as exploring new opportunities/techniques for asset decommissioning.

The Beta path is centred on solidifying and refining existing features that have already been developed at the PoC stage. Our team would seek to develop these preliminary features into a robust, fully embedded and user-friendly tool accompanied by a front-end that enables seamless gas analyst interaction. The aim would be to hone the reliability and performance of the existing PoC features, in order to increase the likelihood of their adoption by analysts, and to conduct a substantial amount of engineering work to embed the tool into NESO's cloud environment and systems.

Each of these potential phases is explored in further detail in the remainder of this section. Both have their own individual benefits, dependencies and risks, which are articulated in subsequent sections.

07.2 Alpha+

The Alpha+ phase of work continues to explore new ways for identifying asset removal options, and utilising data that will be produced by NESO in future that doesn't necessarily exist at present. The primary focus is on further development of the pipe removal recommender tool, identifying different ways to explore the possible space of pipes to remove and providing functionality to explore the resulting network for its resilience and capability limits. The long-term vision is that this will then enable analysts to holistically look at removal options over time to make complex recommendations for successive actions that meet a specific goal on the network, such as meeting long term reduction in demand and adapting the network to new energy sources like hydrogen

This functionality is supported by a number of contributing features which are detailed below.

07.2.1 Roadmap for development (including explanation of unexplored features etc), goals and deliverables

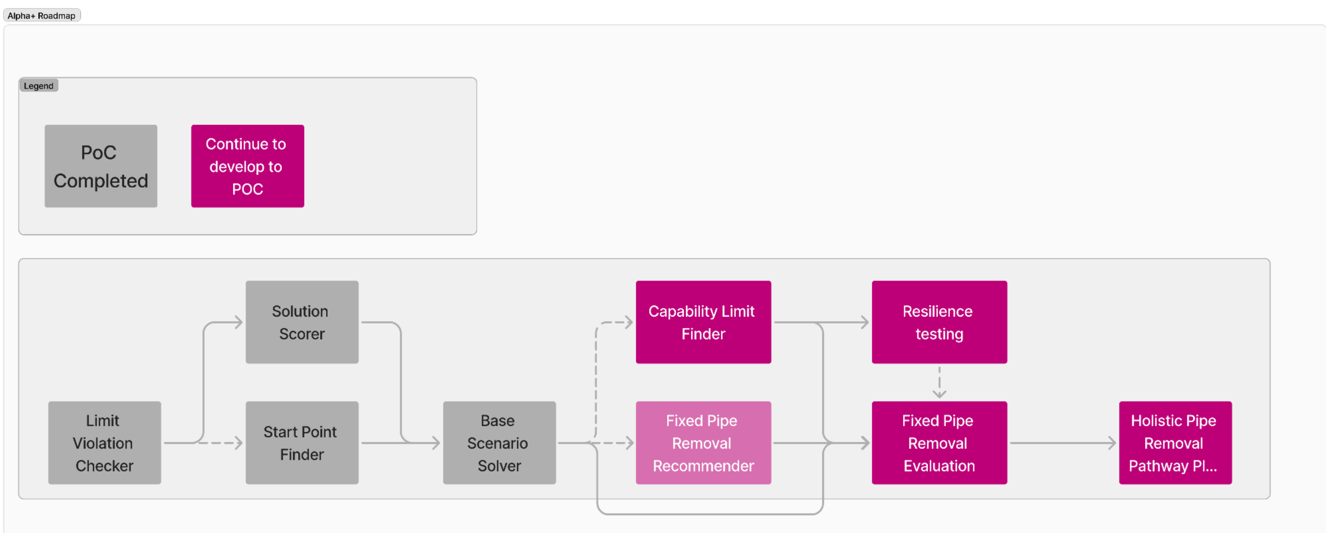


Figure 30 - Roadmap of features for the proposed Alpha+ phase

The Alpha+ package of work builds on the PoC capabilities developed in this project and utilises them to inform the overall goal of this phase; building a smart Holistic Pipe Removal Pathway Planning capability. This capability will result from a combination of features/tools, including the Pipe Removal Recommender tool that we discuss in Section 04 above, as well as new features which are explained below. The aim is to move analysts from manually evaluating the impact of removing single pipes to having them be able to generate long term sequential pipe removal strategies to meet strategic goals. This would add significant new capabilities to analysts, while automating some of their more common tasks. The ability to intelligently recommend pipes for removal in a fashion that can take multiple removals and other complex requirements into consideration is strongly aligned with NESO's long term ambitions.

The package of work consists of five interlinked features. Similarly to the previous phase, features further along the roadmap require earlier features to be completed to work. This is typically so that a process performed by analysts can be codified and adapted into an algorithm for faster processing and input.

While the first feature, the Fixed Pipe Removal Recommender (FPRR) was started as part of the current phase of work, this was with a focus on finding innovative solutions to the question of pipe recommendation. Adapting this work into a format that is suitable for downstream features would still require some work to evaluate and validate the multiple pipe scoring and grouping techniques in further user research sessions. Furthermore, we would need to adapt the output of the various developed algorithms to be subsequently fed to a SIMONE simulation.

To protect the scope of the first project phase, the Capability Limit Finder (CLF) was not included. This work would codify and optimise the logic that analysts use to progressively change input/offtake flows to find the maximum limit that a scenario can take in a certain region. As it is required in subsequent features and is a big part of the analysts' workflow, it is appropriate to tackle it at the start of the Alpha+ package. Coupled with the BSS, this would allow for a full automation of the capability limit analysis workflow. We believe that a combination of gradient based and rules based algorithms would be suitable to search the one-dimensional space of network capability limits. There is also likely further analysis that could be done on how capability limits across the different regions interact with each other under different conditions, which could help give new insight into the gas network's operation.

Another feature that is required to support pipe removal is the current procedure of Resilience Testing (RT), where solutions are passed through a battery of pre-defined alterations to check how well they would perform against shocks. This feature would enable a level of validation of pipe removal recommendations that is in line with current practice and enable analysts to trust the algorithmically generated solutions. Systematically evaluating scenarios against a set of test cases would help build confidence in results and could include analysis on how different scenarios are resilient in differing ways, which could suggest possible mitigations.

All the above features come together as part of the Fixed Pipe Removal Evaluation tool, which orchestrates the testing of pipe removal as recommended by the Fixed Pipeline Removal Recommender, completed by the Base Scenario Solver, stress tested by the Capability Limit Finder and validated with Resilience Testing. The vision is that it would be able to explore the possible space of pipes to remove and evaluate the resulting network for its resilience and capability limits. This would then allow analysts to make intelligent recommendations for pipe removal. While algorithmically straightforward and more of an orchestrating capability, by bringing together the previous features in this fashion we can add a significant boost in ability to analysts.

Finally, extending this capability to include the removal of subsequent pipes and optimising for various constraints is the Holistic Pipe Removal Pathway Planner. This would fully enable analysts to make complex recommendations for successive actions that meet a specific goal on the network, such as meeting long term reduction in demand and adapting the network to new energy sources like hydrogen. This could involve extracting likely demand profiles from Future Energy Scenarios (FES), suggesting when pipes may become available for removal to meet them and iterating this across several years to meet a target.

Plan

The implementation plan is set out below in Figure 31. An enlarged version of the implementation plan can be found in Appendix 9.8 for enhanced visual clarity.

The plan outlines the various areas of activity and their timeframe, although it does not specify the relative effort required for each. For instance, while Wireframes will run concurrently with Local Deployment and Tooling, it is expected to require less than half the time. Some key considerations are that, similar to the previous Alpha phase, we need to undertake more in-depth user research on the proposed features. Despite prior discussions with analysts, more extensive research is crucial for building these features effectively. At the beginning of the project, user research will not be a bottleneck since we will be focusing on further developing existing features to prepare them for expansion. For example, BSS, currently in a Proof of Concept (PoC) phase, requires additional work to be built upon. We've already established a solid foundation to accelerate its runtime, which will support the viability of future features. Furthermore, LVC has some features that were initially omitted as we were able to demonstrate its capability without them, but would need to be included moving forward. As we layer more features on top of each other, local deployment of tooling becomes essential to ensure smooth interaction between features. Establishing a deployment pattern will facilitate efficient integration and make transitioning to subsequent phases easier. Lastly, while Resilience Testing is not on the critical path, it will be included if time and resources allow.

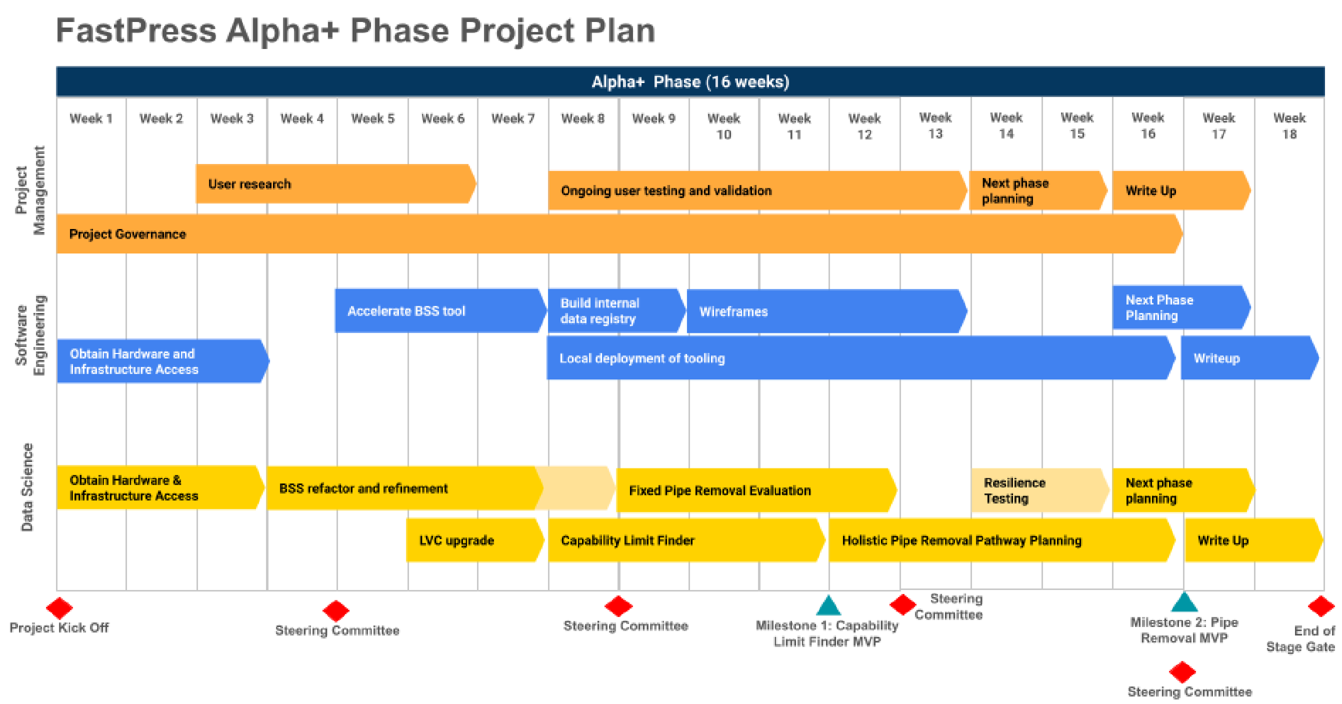


Figure 31 - Implementation plan for the proposed Alpha+ phase showing three main packages of work; project management, software engineering and data science.

07.2.2 Dependencies and Risks

Dependencies

Dependency	Description	Stakeholders
Before Project Kick Off		
Continued access to NESO's systems as per current set up	Each developer will require access to: • An active directory account • A VDI	• Lauren Cooper (Innovation Business Partner) • Michelle Stevens (Senior

	• A laptop	Manager)
Assessment of cost and time associated with obtaining additional SIMONE licences	Liaising and obtaining approval of licence usage (including associated budget required) for FastPress should be completed ahead of project kick off.	• Aneta Ivanova (SIMONE License Manager) • Lanre Alabede (IT Architect) • James Carlin (Lead Architect)
During Project		
Access to either future users / owner	The team will require user research sessions with someone who can inform and feedback on pipe removal capability	• Lauren Cooper (Innovation Business Partner) • Toby Thornton (ESO)

Risks

Risk Description	Likelihood	Impact	Mitigation
Pipe removal requires the API to work properly, and obtaining this requires a new licence.	**Medium**	**High**	We will work with NESO from project kick off to understand timelines and actions regarding licence obtainment. We have potential workarounds that we believe we can implement, but these are presently untested - we will test these early in the project to allow sufficient time for mitigation if they do not work.
The BSS may not solve well enough to enable additional features identified to work effectively	**Low**	**Medium**	We have tested the BSS capability during the Alpha phase of delivery and have a plan for further improvement that will be actioned in the early stages of Alpha+, as per the plan above.
The resilience testing package may require specific API settings for SIMONE which we are unaware of yet	**Low**	**Medium**	We will test this early in the project to allow sufficient time for mitigation/obtainment of appropriate licences if required.
There is unforeseen cost or time associated with obtainment of additional licences	**Low**	**High**	We will start conversations to quantify the process for obtaining additional licences prior to project commencement, so the cost and time impact of risks can be better understood.
There is a risk that the scope of the holistic pipe removal pathway planner is limited due to time restrictions	**Medium**	**Low**	As this is PoC, we have allowed enough time during this phase to understand the potential of the feature and make a judgement on whether it is technically possible to produce feasible results.

07.2.3 Benefits

Below we present some identified benefits of the features proposed for development in Alpha+. In order to understand the full benefit of exploring these, the assumption is that the exploratory work could lead to features which are deployed, which is where most of the primary benefit to NESO will be realised. That being said, there are benefits that exist as stand alone without potential further development, for example regarding identification of new insight and approaches to infrastructure planning, as these will be identified through the development stage. The target realisation date of these benefits assumes that an Alpha+ project would commence in Q1/Q2 of 2025 - these dates should be shifted accordingly depending on project commencement.

Benefit Title	Benefit Description	Proposed Metric	Target Realisation Date
Identification of new insights and approaches to infrastructure planning	Data exploration and analysis may reveal patterns and alternate approaches to asset assessment that have not been considered or applied previously. This could allow for timely changes to the infrastructure management process, helping NESO ensure the network is fit for the future.	>1 alternate approach to asset assessment identified through use of the tool	Q2 2025
Improved understanding of demand shift impact on pipe utilisation	These features will assist analysts in understanding when and how demand might shift and will provide them with a tool to visualise how this affects pipe utilisation, allowing them to incorporate this information into their planning.	Analysts report increased capability/scope of their assessments.	Q4 2025
Increased consistency of approach	The tool will have accompanying training material and documented process flows, meaning that new analysts will all be onboard to conduct asset assessment in a consistent manner.	\	Q3 2025
Enhanced auditability and compliance	Using the tool will allow for records of analysis conducted to be kept, enhancing the auditability of the work conducted by the analysts, which could then be supported by providing evidence for reports, analysis etc.	Run registry is capturing previously conducted analysis	Q3 2025
More effective project planning	Looking at the changes to pipe utilisation over time and applying a whole network approach will enable analysts to compare and develop optimal sequences for infrastructure upgrades and renewals.	Increased volume of comparisons conducted on asset removal strategies.	Q4 2025
Analyst time saved and capacity increased	Removing manual steps from analysts' workflows will mean they spend less time inputting data into simulations for pipe removal, and are therefore able to increase the number of scenarios/strategies they assess.	Increased volume of scenarios run/strategies assessed.	Q4 2025
New capability to understand the broad consequences of network changes	Through development of the capability limit finder, you enable the rest of the features to reflect capability limits in their workflows (e.g. if a pipe is selected for removal, how will that impact capability limits).	Increased number of factors considered when analysing pipe removal candidates	Q4 2025
Improved investment decisions	Forecasting demand, increasing the capability of analysts to run comparisons of strategy and aligning infrastructure changes accordingly could allow for improved investment decisions, ensuring capital is invested judiciously.	£GBP savings identified through implementation of most cost effective strategies.	Q2 2026 onwards

07.3 Beta

07.3.1 Deployment Plan/roadmap for development, description, goals and deliverables

Plan

The implementation plan is set out below in Figure 31. An enlarged version of the implementation plan can be found in [Appendix 9.9](#) for enhanced visual clarity.

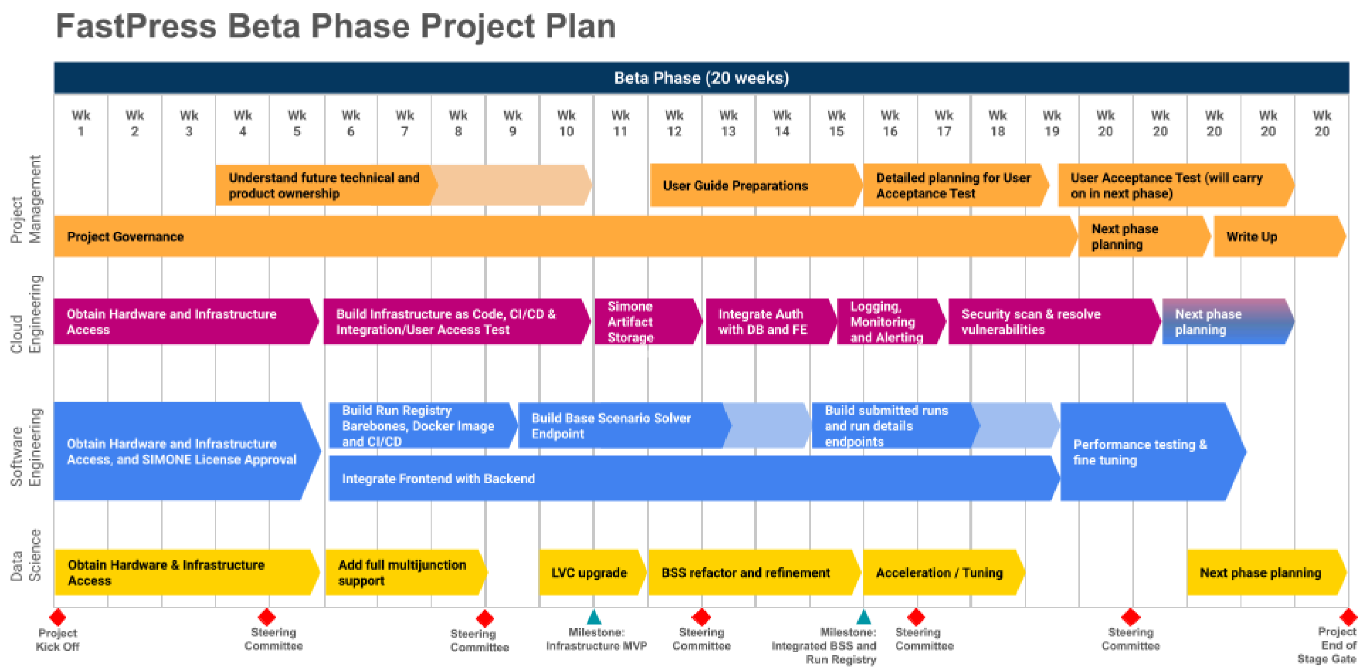


Figure 32 - Implementation plan for the proposed Beta phase showing four main packages of work; project management, cloud engineering, software engineering and data science.

Goals

The main goal of this phase is to deploy the FastPress tool on NESO's Cloud environment so that it is accessible by Gas Network Planning Analyst for user acceptance testing. Alongside this, we will also aim to:

- Complete and refine the Base Scenario Solver and Pipe Removal Recommender features
- Develop user guides for Gas Network Planning Analysts
- Plan for future phase of the project to ensure handover and maintainability

Deliverables

At the end of this phase, the following deliverables will be completed:

- FastPress Tool deployed on NESO's Cloud Environment, ready for User Acceptance Testing
- Improved algorithmic performance of the Base Scenario Solver with completion of multi-junction support and Limit Violation Checker

07.3.2 Dependencies and Risks

Considerations

NESO's Cloud Platform Development Timelines

In formulating the project plan, we have considered the development timelines of NESO's Cloud Platform.

NESO is currently undergoing the setup of its new Cloud Platform, scheduled to be ready for use in December 2024. This will enable web applications deployed in NESO's Cloud platform to be accessed by VDIs provisioned by NG/NESO. However, network integration to allow NESO laptops to be able to access web applications deployed in NESO's Cloud platform is not scheduled to complete until March 2025. This means that between December 2024 and March 2025, Gas Network Planning Analysts will not be able to access the FastPress tool. We will deploy on their laptop directly if this phase were to kick off during this period.

To reduce the risk of project delays, we suggest that testing user access to the FastPress tool from NESO laptops to **not** be in scope if network integration work is not complete. As a mitigation, NESO can provide Gas Network Planning Analysts access to VDIs to be able to access the FastPress tool.

Dependencies

Dependency	Description	Stakeholders
Before Project Kick Off		
Access to NESO's systems	Each developer will require access to: • An active directory account • A VDI • A laptop	• Lauren Cooper (Innovation Business Partner) • Michelle Stevens (Senior Manager)
Single point of contact from NESO cloud/ infrastructure team	There are many stakeholders that Faculty has to interface with in order to resolve all access issues as listed in this table. We require a single representative from the IT and Cloud team within NESO to be accountable for ensuring all permission and access issues are resolved.	• Michelle Stevens (Senior Manager) • Simon Wright (Senior Platform Manager - Cloud)
Developer access to a Windows Machine	In past phases of the project, permissions within VDI and laptop were very restrictive. We had issues installing softwares required to complete the project and had spent time coming up with various workarounds which negatively affected the quality of outputs. Obtaining developer access to a laptop will be beneficial in ensuring timely completion and assuring the quality of build of the project. Some examples of installation we require are: • Azure CLI • Terraform • PNPM • Docker Some of which require administrative access to the machines we use.	• Daren Harrison (Engagement Manager) • Michelle Stevens (Senior Manager)
Approval for SIMONE License	In the past, we have received approval to utilise SIMONE Licences for the development of the FastPress tool. As the existing licences are	• Aneta Ivanova (SIMONE License Manager)

usage in FastPress Tool	reserved for business-as-usual activities for the Gas Network Planning Analyst, the change in licence usage will require risk assessment and approval by the SIMONE Offline team. Liaising and obtaining approval of licence usage (including associated budget required) for FastPress should be completed ahead of project kick off.	• Lanre Alabede (IT Architect) • James Carlin (Lead Architect)
During Project (Week 1 - 3)		
Cloud subscription, network integrations and permissions	The required Azure set up and permissions have been listed in the following section **Fastpress Project: Alpha Report DRAFT V0.1** We anticipate the provisioning of resources and permissions may require multiple iterations of testing, and so have reserved 3 weeks in the initial phase of the project for teams in NESO and Faculty to work together.	• Simon Wright (Senior Platform Manager - Cloud) • Lyndon Ruff (AI CoE Manager) • Joshua Brooke (Machine Learning Engineer)
During Project (Week 4 onwards)		
Clarity on future Ownership of FastPress tool	To ensure longevity of the FastPress tool beyond the Beta phase, we require a Product Owner and a Technical Owner to be assigned. As part of our Governance workstream, we will assist NESO to identify the right fit within the business.	• Toby Thornton (Gas Network Planning Manager) • Haipeng Liu (Gas Network Planning Analyst)
During Project (Week 14 onwards)		
Access to users for User Acceptance Testing	To ensure the deployed tool meets the User and Non-Functional Requirements stipulated in this report, we require a select group of Gas Network Planning Analysts to allocate time to execute User Acceptance Testing for the tool from Week 14 of the project. Exact plans will be formulated during the course of Beta.	Gas Networking Planning Analysts

Risks

Risk Description	Likelihood	Impact	Mitigation
Not getting access to NESO's systems (hardware, software and cloud) with adequate permissions to complete product build	**High**	**High**	There are many stakeholders that Faculty has to interface with in order to resolve all access issues as listed in the Dependencies section above. We require a **single representative from the IT and Cloud team within NESO to be accountable for ensuring all permission and access issues are resolved**.
Not obtaining adequate access to SIMONE licences for deployed FastPress tool due to limited number of licences reserved for Gas Analysts	**Medium**	**High**	Approval for SIMONE licence usage for FastPress including finances required to procure new licences should be completed prior to the start of project to avoid any delays or a potential "No-Go".
Poor engagement during user acceptance testing	**Medium**	**Medium**	Ensure to make detailed plans and communicate expectations to those involved during the User Acceptance Testing phase of the project.

Unclear ownership for handover	**Medium**	**High**	Faculty to allocate time in project plan to assist NESO in identifying a Technical and Product Owner for the FastPress tool during the Beta Phase.

07.3.3 Benefits

Below we present the identified benefits of the proposed Beta phase, which will productionise the work already conducted on FastPress in the Alpha phase. The target realisation date of these benefits assumes that an Beta project would commence in Q1/Q2 of 2025 - these dates should be shifted accordingly depending on project commencement.

Benefit Title	Benefit Description	Proposed Metric	Target Realisation Date
First deployment into NESOs new cloud platform	This will be the first deployment of a tool in NESO's cloud platform, which means that the team will be uncovering and resolving issues, lessons from which can be provided to following projects and deployments. Additionally the tool will demonstrate the possibility of deployment and can be used to advocate for additional cloud deployments.	Deployment in cloud platform achieved	Q3 2025
Demonstrates the implementation of an innovation project	Taking an innovation project from Discovery and developing it into a deployed tool that analysts can actually interact with will demonstrate to NESO stakeholders the value of investing in innovation projects.	Improved business endorsement and engagements metrics for innovation projects	Q2 2025
Increased consistency of approach to gas network analysis	The elements of the analysts workflow that rely on subjective assessment will be reduced as this tool will be embedded into business-as-usual processes, improving consistency across analysts.	Difference between analysts in number of scenarios that need to be rerun post audit should reduce	Q2 2025
Enhanced auditability and compliance	Use of the tool will allow for records of analysis conducted to be kept, streamlining the process of auditing solutions for senior gas analysts.	Run registry is capturing previously conducted analysis	Q2 2025
Improved training for new analysts	The tool will have accompanying training material and documented process flows, meaning that new analysts will all be onboarded to conduct asset assessment in a consistent manner.	Production of documented training materials and process flow.	Q3 2025
Analyst time saved and capacity increased	Removing manual steps from analysts' workflows (for example by allowing them to view the Limit Violation Checker table output rather than individual compressor envelope windows) will mean they spend less time checking and adjusting simulations and are therefore able to increase the number of scenarios/strategies they assess.	Increased volume of scenarios run per analyst.	Q2 2025

08 Conclusion

In this report we have presented the findings of the Alpha phase of the FastPress project. The project has successfully demonstrated a PoC for an automated tool that enables gas analysts to solve base static scenarios, and through iterative development we have explored various functionalities that enhance that tool's ability to analyse scenarios with greater efficiency and reliability. The PoC has demonstrated the potential of this tool in allowing gas network planning analysts to address network analysis challenges with improved consistency and efficiency. The features provide a robust foundation for further development, showcasing the tool's potential to automate complex gas scenario analyses, reduce manual effort and improved decision-making support.

In addition, we have presented our work exploring the possibility of data science supported asset management through our PoC development of the 'pipe removal' tool. As explained above, this work has been very exploratory as asset management is a relatively new capability for NESO and there was not a high volume of documented processes for conducting asset assessment. Whilst the findings of the 'pipe removal' workstream have been definitively encouraging in terms of the application of data science techniques to this topic, it is clear that with just one single stand alone feature, this workstream is significantly less mature. There are clear requirements for more analysis and exploratory work to be conducted and fed into additional 'features' before this can be developed fully into a tool that can enhance NESO's asset planning capability.

Looking ahead we have identified two primary paths for further advancement, as covered in Section 07:

1. Beta: Productionising the existing features that feed into the Base Scenario Solver tool
2. Alpha+: Engaging in more exploratory work to incorporate additional, advanced functionalities within the pipe removal workstream

Productionisation will involve refining the current features to meet operational standards, ensuring performance, stability, and integration with existing systems. It will seek to identify a future owner(s) for the tool and will conclude with a phase of user acceptance testing which will continue into the live deployment phase. Alternatively, pursuing exploratory research into additional features for the asset decommissioning/pipe removal workstream could expand the tool's capability areas, and allow for data-based innovation in an area that is not currently well-defined. These options could be delivered separately as stand alone projects or in parallel. As per the benefits outlined in Section 07.2.3 and Section 07.3.3, both of these options provide the opportunity for NESO to realise a clear and measurable return on investment.

09 Appendices

09.1 User Requirements

Description	Priority
Gas Networking Planning Analyst is able to solve static base scenarios with the tool	High
Gas Networking Planning Analyst is able to get a recommendation of pipes to be removed from a network based on the scenario submitted on the tool	Medium
Gas Networking Planning Analyst is able to submit more than one scenarios in a single submission for solving static base scenarios	Medium
Gas Networking Planning Analyst is able to solve exit capability analysis for specified region with the tool	Low
Gas Networking Planning Analyst is able to solve entry capability analysis for specified region with the tool	Low
Gas Networking Planning Analyst is able to modify pressure limits that define the boundaries of a solved solution	Medium
Gas Networking Planning Analyst is a able to modify the Network that is being used by the tool	Low
Gas Networking Planning Analyst is able to validate their inputs to the system, before inputs are submitted to the tool	Low
Gas Networking Planning Analyst is able to view a list of their own submissions, associated metadata and relevant results	High
Gas Networking Planning Analyst is able to view a list of submissions made by other gas networking planning analysts in the same team, the associated metadata and relevant results	Medium
Gas Networking Planning Analyst is able to verify and validate SIMONE outputs from runs made by the tool	High
Gas Networking Planning Analyst is able to inspect a range pass/failed SIMONE runs trialled by the tool	High

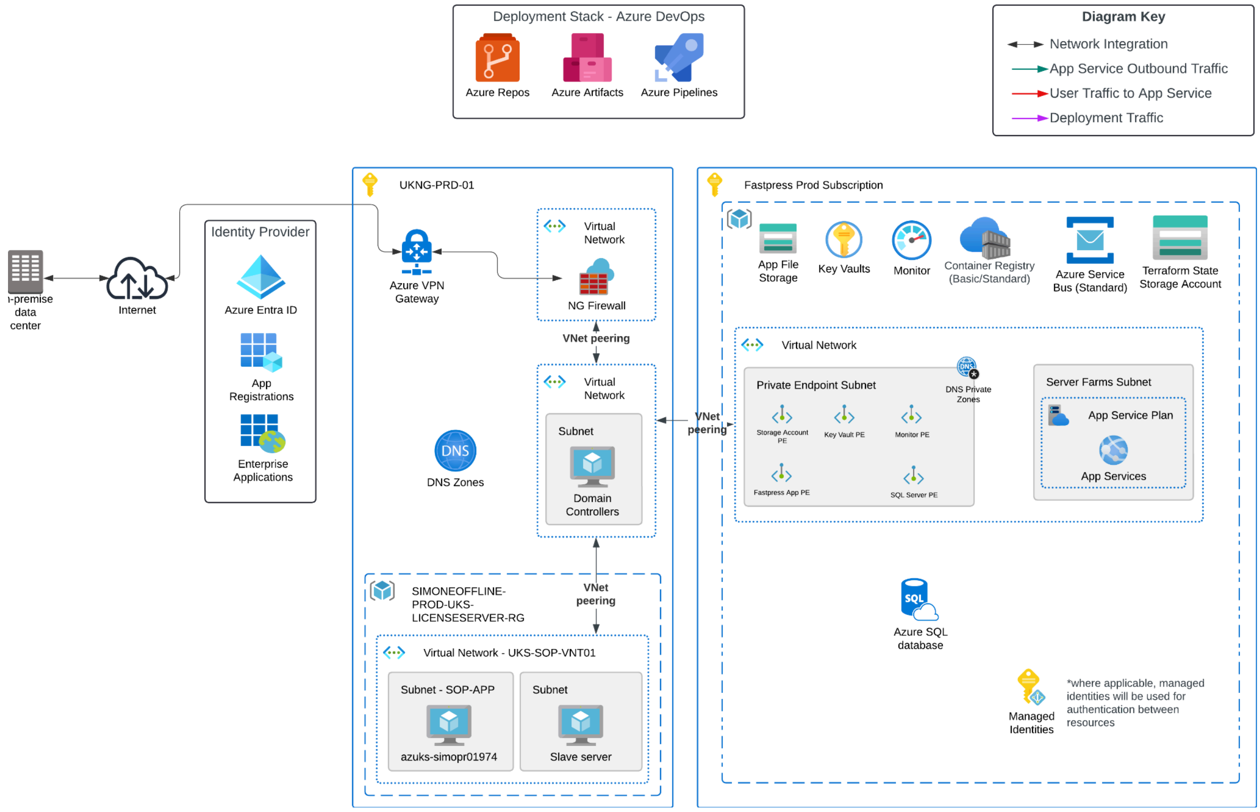
09.2 Non-functional Requirements

Description	Priority
The tool is only accessible within NESO's Internal Network using NESO's VDI and/or NESO laptop	High

Description	Priority
The tool is only accessible by a selected group of gas analysts within NESO (Gas Networking Planning Analyst)	High
Network traffic of all sensitive information handled by the tool should remain in NESO's Network	High
The server hosting the tool must have access to the Internet	High
The tool is able to support 10 concurrent users during business hours	High
The tool is able to solve a static base scenario in under 1 hour (based on active run time, time in queue is excluded).	Medium
The tool is able solve an exit capability limit in under 6 hours (based on active run time, time in queue is excluded).	Medium
The tool is able solve an exit capability limit in under 5 hours (based on active run time, time in queue is excluded).	Medium
Support Team will need to receive alerts when users' submissions are stuck in idle	Low
Support Team will need to receive alerts when application is down	Low

09.3 Cloud Deployment

09.3.1 Overall Architecture



09.3.2 Cloud Deployment Initial Set Up and Access Requirements

The following section details the set up required by Faculty on NESO's Cloud Environment

Subscriptions/Resource Groups

3 subscriptions/resource groups are required in total, Dev, QA and Production environments.

Access Requirements

Developer Azure Entra ID User Account

All Faculty engineers and data scientists (emails and details provided separately) to be invited to NESO's Azure tenant/directory. We recommend that our engineers' MS Entra accounts are "member" accounts not "guest" accounts with permission to read the configuration of the app registration for debugging purposes.

Security Groups

We recommend creating three security groups within Entra ID (names are provisional):

- a group for faculty-engineers, and
- a group for faculty-database-administrators,
- a group for faculty-database-directory-reader (with Entra Role assignment enabled), this group should be assigned the "Directory Reader" Entra Role

Application Registration for SSO

This is required to provide authentication service to our web application.

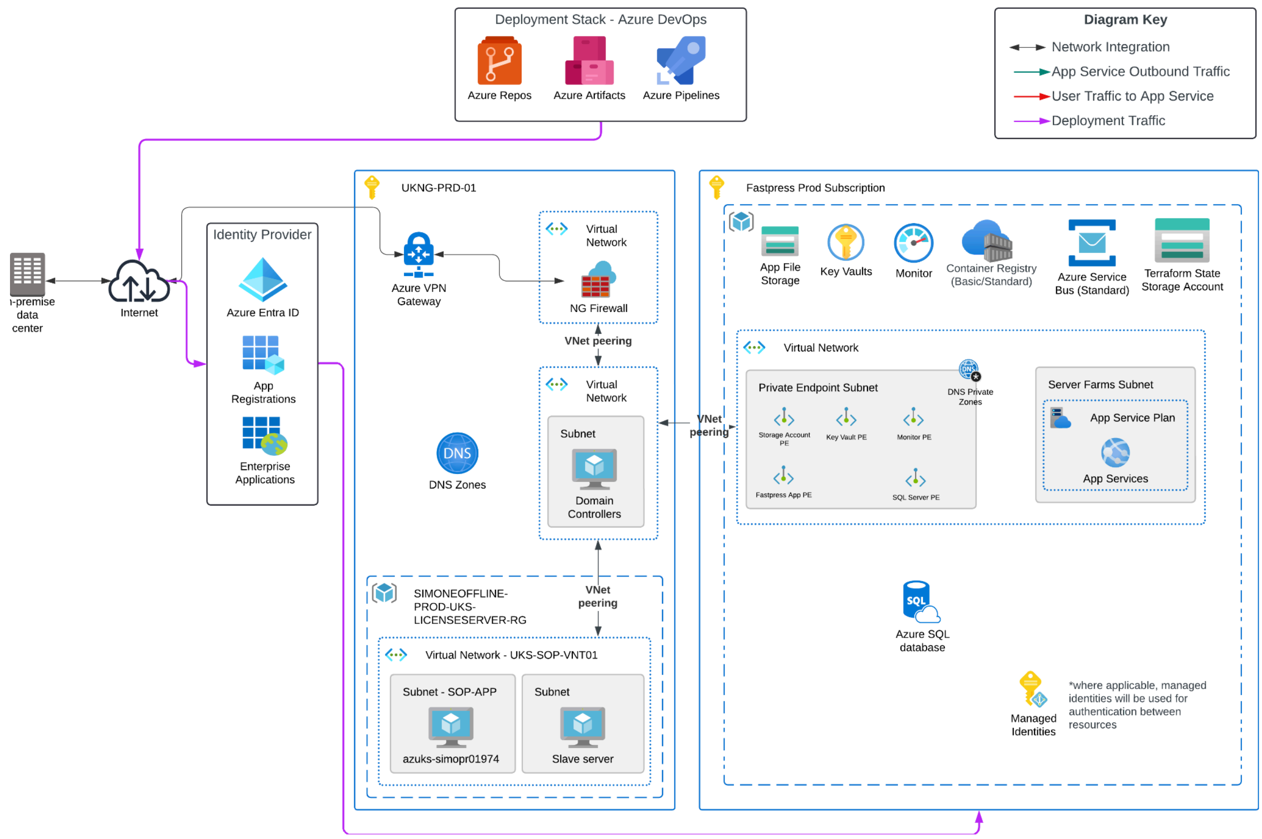
Settings required:

- ID and Access token generation enabled
- Grant access for Faculty in API Permission
- Redirect URI (dependent on Azure App Service default and custom URL)

We may rely on Enterprise Application and security groups for authorisation, but will provide more details after we have clarified requirements.

Service Principal required for Deployment via Azure Pipelines

The Service Principal will be used as an Identity by our Deployment Pipelines on Azure DevOps as laid out in the Deployment Architecture section. We will require client ID, object ID, client secret, secret ID and client secret expiry date.



Providing Directory Reader Entra Role to the SQL Server system-assigned Identity

To use MS Entra ID as an authentication provider for SQL Server, this role is required.

Security Group Membership

All Faculty Engineers and required Service Principals (see above) are to be added into both the faculty-engineers security group and the faculty-database-administrators security group. Faculty Engineers should be added as Owner to the faculty-database-directory-reader Group.

- The faculty-engineers group can be used to provide permissions to Azure resources.
- The faculty-database-administrators security group will be used to assign database administrator privileges.
- The faculty-database-administrators group will provide you with the flexibility to grant permissions to nominated individuals from your organisation by adding them to the group.

A SQL Server System-Assigned Managed Identity will be added as a member of faculty-database-directory-reader security group. This security group is to enable the SQL Server to read the Azure directory to be able to use information to infer logins for the database. As this requires

Faculty Engineers to first deploy the Database, it is recommended that Faculty Engineers are given ownership so they can add the Managed Identity when it is ready.

Networking

Subnets

We require 2 subnets per environment:

- A subnet for Azure App Service with minimum IP range is /26 (64 IP addresses), as recommended in Microsoft's documentation: [Enable integration with an Azure virtual network - Azure App Service | Microsoft Learn]
- A subnet for Private Endpoints with minimum of /27 (32 IP addresses). We require 1 IP address per private endpoint and 32 allows for slightly more than is currently needed, however we require more to allow flexibility in case of future changes. The Microsoft.KeyVault and Microsoft.Storage service endpoints must be selected.

Private Link Scope

An Azure Monitor Private Link Scope is required to enable private links between App Service and Azure Monitor.

Private Endpoints

A private endpoint will be set up for each of the following resources:

- SQL Server
- App Service
- Key Vault
- App Storage Account
- Application Insights (Microsoft.Insights/privateLinkScopes)*

*Note that when the Private Endpoint for Application Insights is created, ensure that Private DNS Integration is selected. This will automatically create several private DNS Zones required for the Azure Monitor suite of tools.

Private DNS Zones

This is used for providing name resolution for privately addressed VM instances and resources that are not accessible from the public internet. The following will be set up:

- privatelink.azurewebsites.net for Azure App Service

- privatelink.database.windows.net for SQL Server
- privatelink.blob.core.windows.net for Storage Account
- privatelink.vault.azure.net for Key Vault

alongside the Azure Monitor private DNS Zones automatically provisioned as mentioned in the above section.

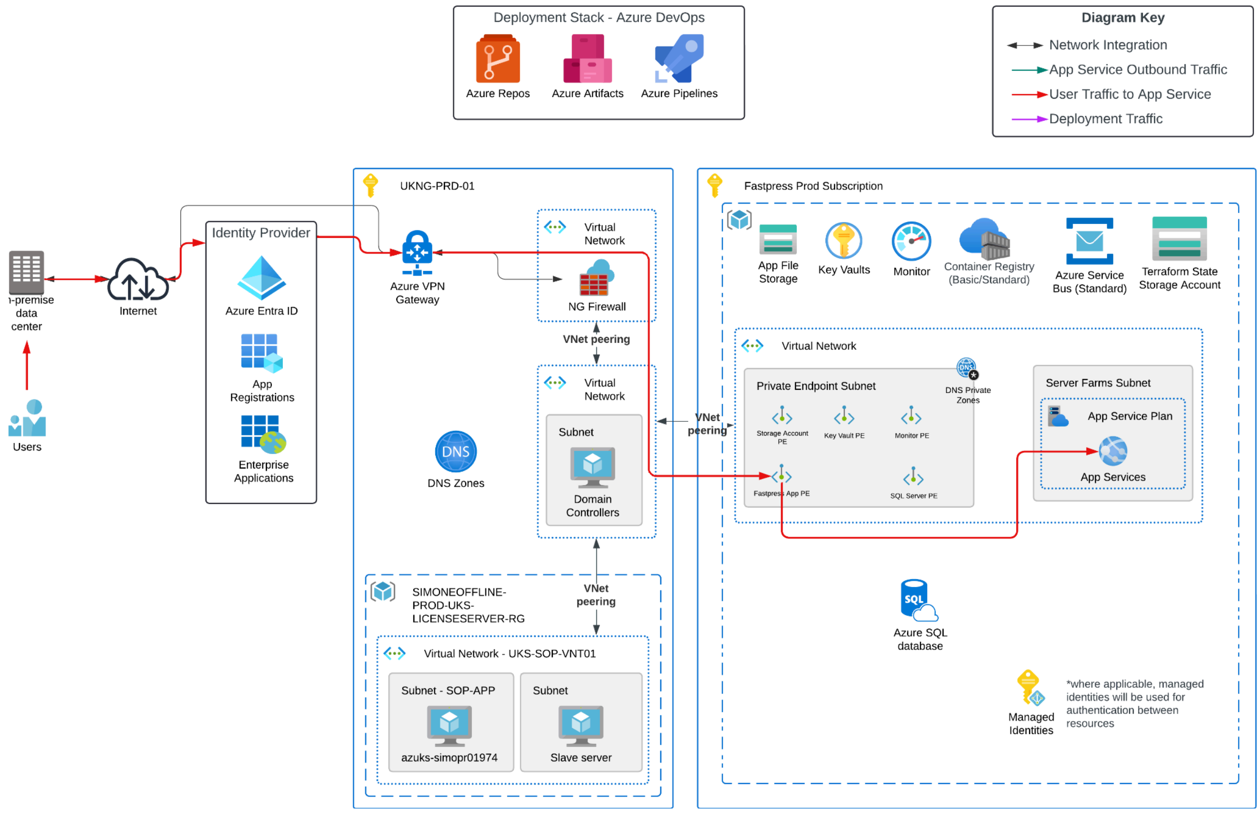
NB: These private DNS Zones need only to be deployed within the deployment VNets, and hence do not necessarily need to exist across any wider subscription that the VNet might live in. This ensures that the name resolution (of the private endpoints) works correctly.

Public Subdomain (TBD)

If a custom domain is required for the web application, we need a TXT Record and A Record added to NESO/NG's public DNS Server. Exact values for the records will only be known after the Azure App Service has been provisioned. Though this DNS record is public, the web application itself will only be accessible within NESO's network.

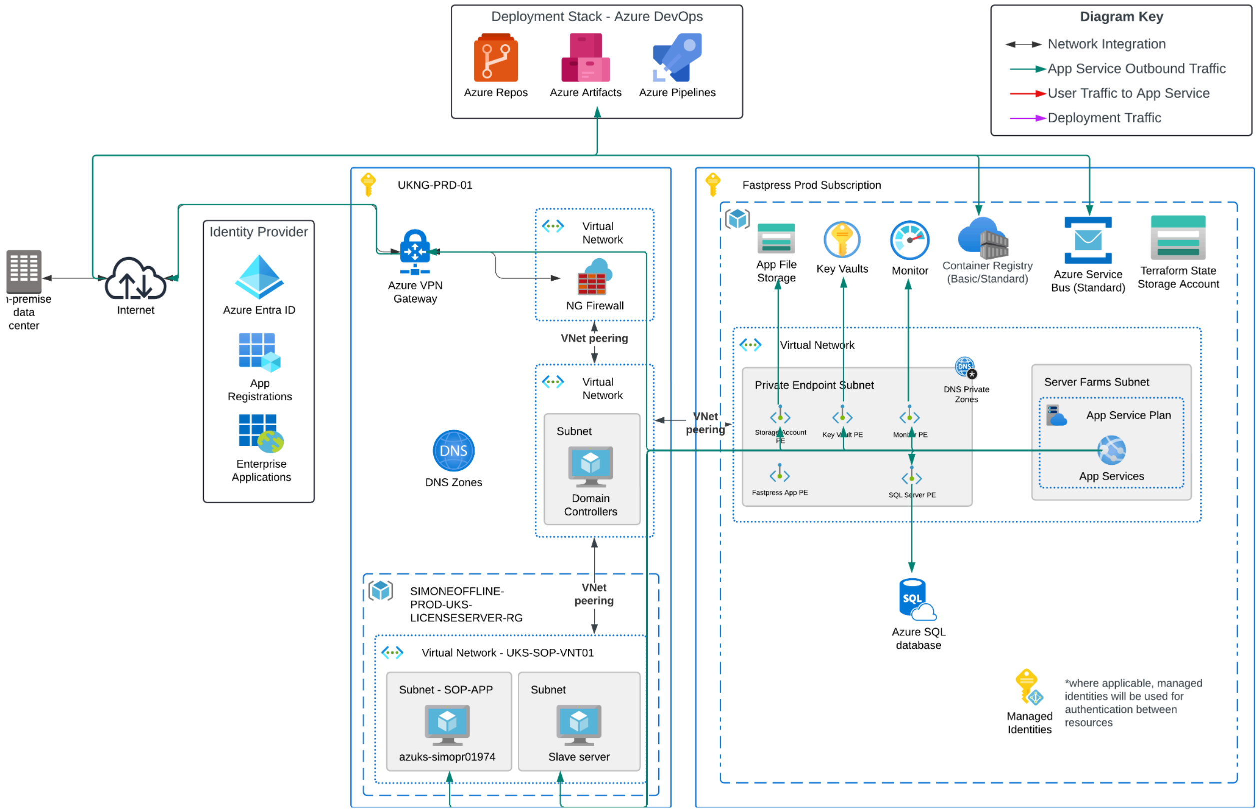
Firewall Whitelisting

- User laptops and VDIs should have access to the App Service Private Endpoint



- Internet Access from Azure App Service
- Egress from Azure App Service to all other resources - i.e. traffic from server farms subnet to private endpoint subnets

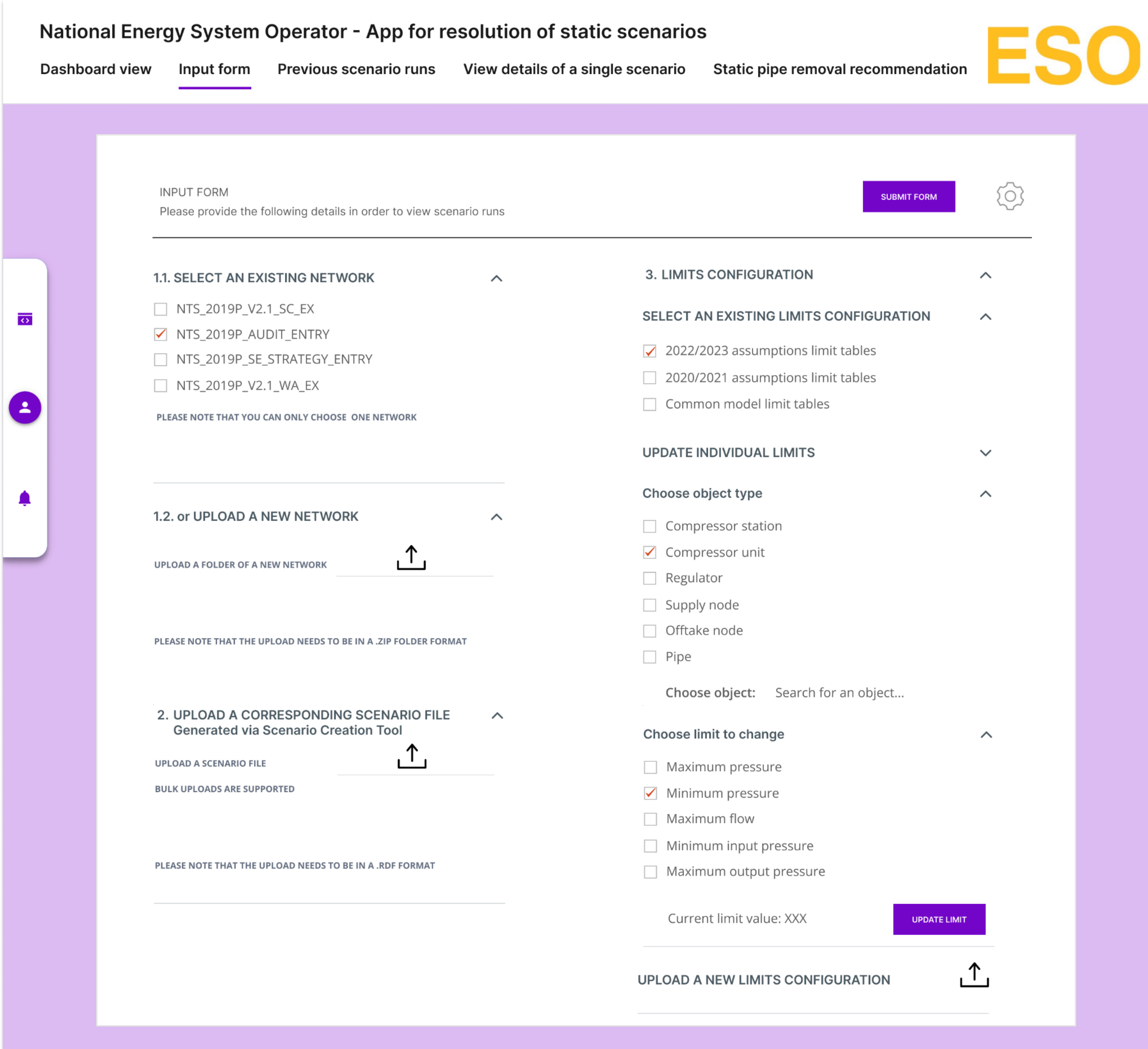
- Egress from Azure App Service to the SIMONE Offline License Servers - i.e. perhaps via VNET Peering



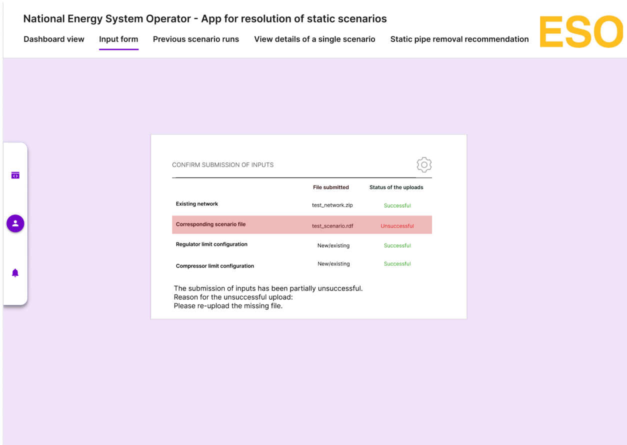
09.4 Early Wireframes

09.4.1 First iteration

Input form page



Post-submission popups



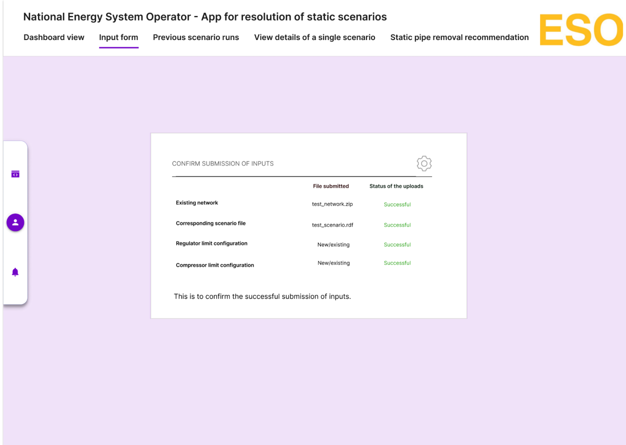
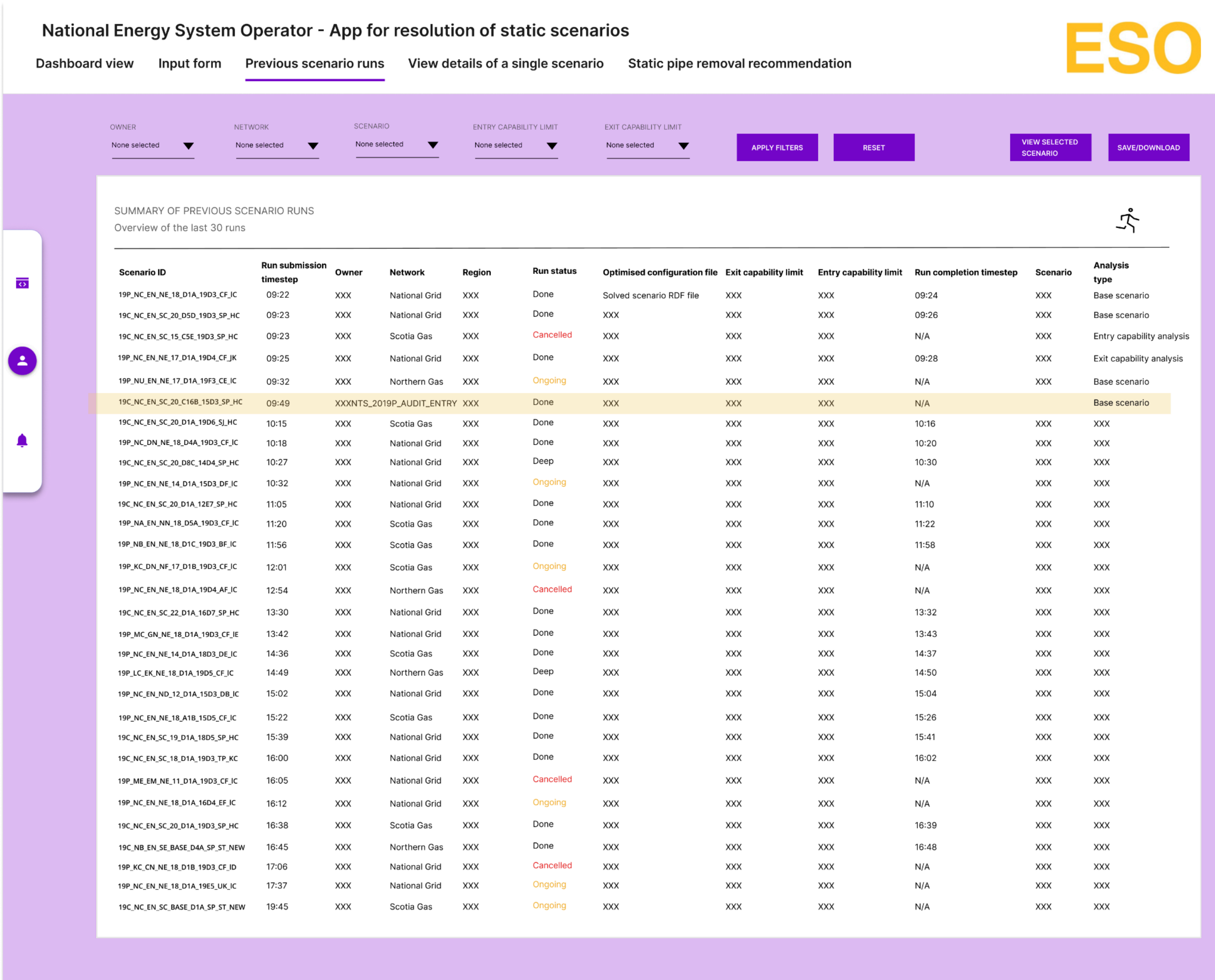
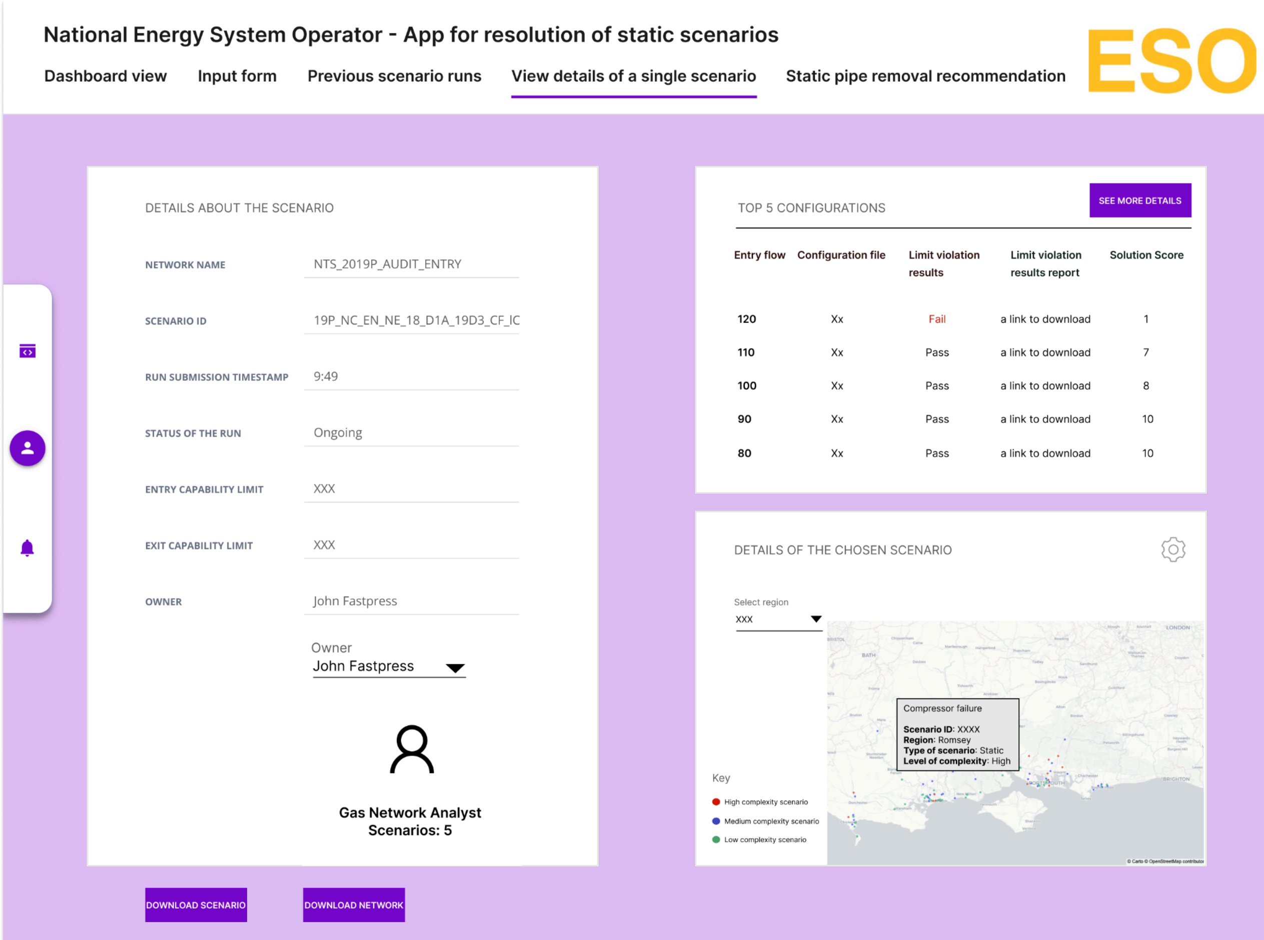


Table of past scenario runs page

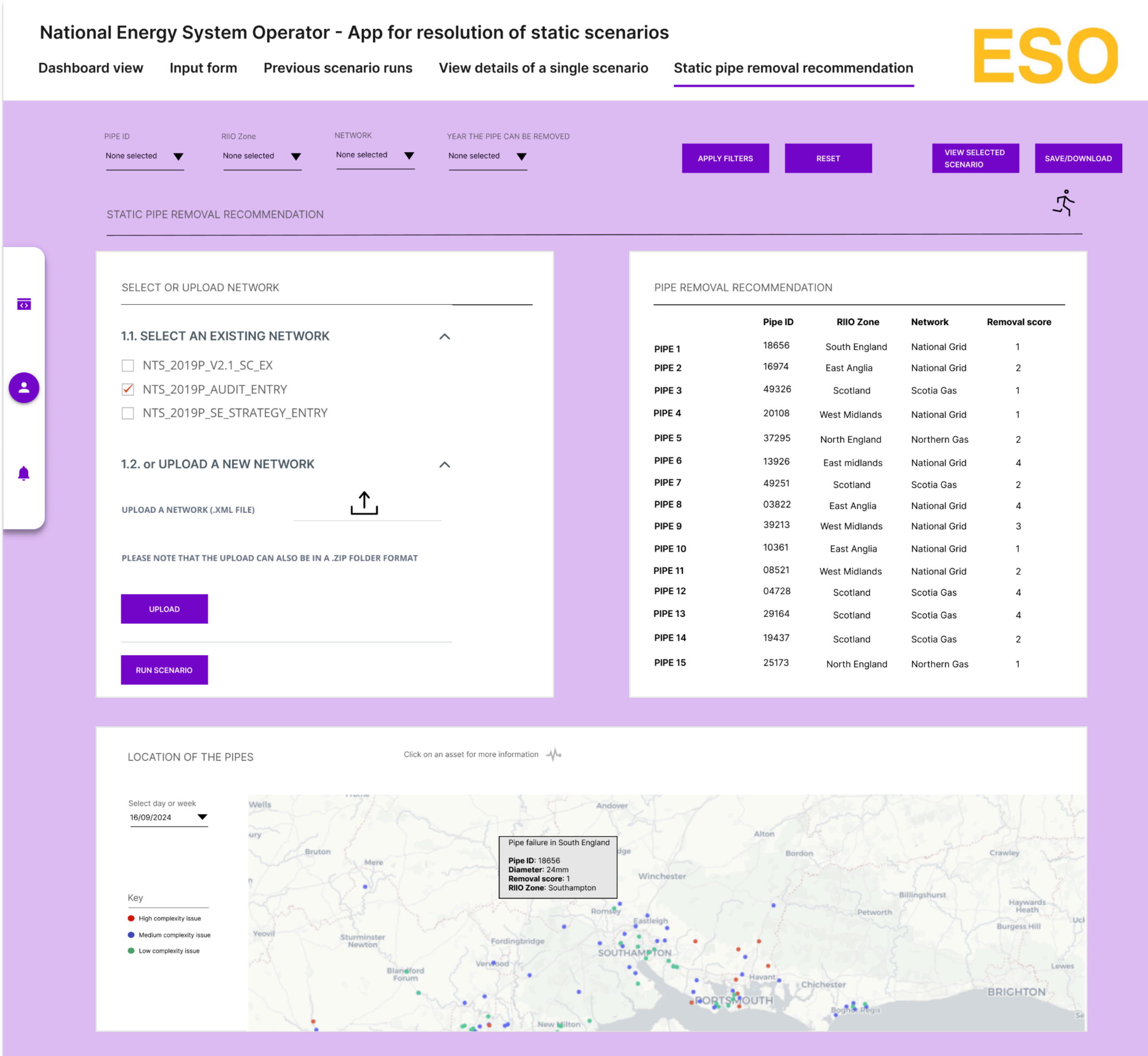


Scenario ID	Run submission timestep	Owner	Network	Region	Run status	Optimised configuration file	Exit capability limit	Entry capability limit	Run completion timestep	Scenario	Analysis type
19P_NC_EN_NE_18_D1A_19D3_CF_IC	09:22	XXX	National Grid	XXX	Done	Solved scenario RDF file	XXX	XXX	09:24	XXX	Base scenario
19C_NC_EN_SC_20_D5D_19D3_SP_HC	09:23	XXX	National Grid	XXX	Done	XXX	XXX	XXX	09:26	XXX	Base scenario
19C_NC_EN_SC_15_C5E_19D3_SP_HC	09:23	XXX	Scotia Gas	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	Entry capability analysis
19P_NC_EN_NE_17_D1A_19D4_CF_JK	09:25	XXX	National Grid	XXX	Done	XXX	XXX	XXX	09:28	XXX	Exit capability analysis
19P_NU_EN_NE_17_D1A_19F3_CE_IC	09:32	XXX	Northern Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	Base scenario
19C_NC_EN_SC_20_C16B_15D3_SP_HC	09:49	XXXNTS_2019P_AUDIT_ENTRY		XXX	Done	XXX	XXX	XXX	N/A		Base scenario
19C_NC_EN_SC_20_D1A_19D6_SJ_HC	10:15	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	10:16	XXX	XXX
19P_NC_DN_NE_18_D4A_19D3_CF_IC	10:18	XXX	National Grid	XXX	Done	XXX	XXX	XXX	10:20	XXX	XXX
19C_NC_EN_SC_20_D8C_14D4_SP_HC	10:27	XXX	National Grid	XXX	Deep	XXX	XXX	XXX	10:30	XXX	XXX
19P_NC_EN_NE_14_D1A_15D3_DF_IC	10:32	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_20_D1A_12E7_SP_HC	11:05	XXX	National Grid	XXX	Done	XXX	XXX	XXX	11:10	XXX	XXX
19P_NA_EN_NN_18_D5A_19D3_CF_IC	11:20	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	11:22	XXX	XXX
19P_NB_EN_NE_18_D1C_19D3_BF_IC	11:56	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	11:58	XXX	XXX
19P_KC_DN_NF_17_D1B_19D3_CF_IC	12:01	XXX	Scotia Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_19D4_AF_IC	12:54	XXX	Northern Gas	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_22_D1A_16D7_SP_HC	13:30	XXX	National Grid	XXX	Done	XXX	XXX	XXX	13:32	XXX	XXX
19P_MC_GN_NE_18_D1A_19D3_CF_IE	13:42	XXX	National Grid	XXX	Done	XXX	XXX	XXX	13:43	XXX	XXX
19P_NC_EN_NE_14_D1A_18D3_DE_IC	14:36	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	14:37	XXX	XXX
19P_LC_EK_NE_18_D1A_19D5_CF_IC	14:49	XXX	Northern Gas	XXX	Deep	XXX	XXX	XXX	14:50	XXX	XXX
19P_NC_EN_ND_12_D1A_15D3_DB_IC	15:02	XXX	National Grid	XXX	Done	XXX	XXX	XXX	15:04	XXX	XXX
19P_NC_EN_NE_18_A1B_15D5_CF_IC	15:22	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	15:26	XXX	XXX
19C_NC_EN_SC_19_D1A_18D5_SP_HC	15:39	XXX	National Grid	XXX	Done	XXX	XXX	XXX	15:41	XXX	XXX
19C_NC_EN_SC_18_D1A_19D3_TP_KC	16:00	XXX	National Grid	XXX	Done	XXX	XXX	XXX	16:02	XXX	XXX
19P_ME_EM_NE_11_D1A_19D3_CF_IC	16:05	XXX	National Grid	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_16D4_EF_IC	16:12	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_20_D1A_19D3_SP_HC	16:38	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	16:39	XXX	XXX
19C_NB_EN_SE_BASE_D4A_SP_ST_NEW	16:45	XXX	Northern Gas	XXX	Done	XXX	XXX	XXX	16:48	XXX	XXX
19P_KC_CN_NE_18_D1B_19D3_CF_ID	17:06	XXX	National Grid	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_19E5_UK_IC	17:37	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_BASE_D1A_SP_ST_NEW	19:45	XXX	Scotia Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX

Single scenario details page



Pipe removal recommendations page



09.4.2 Second iteration

Input form page

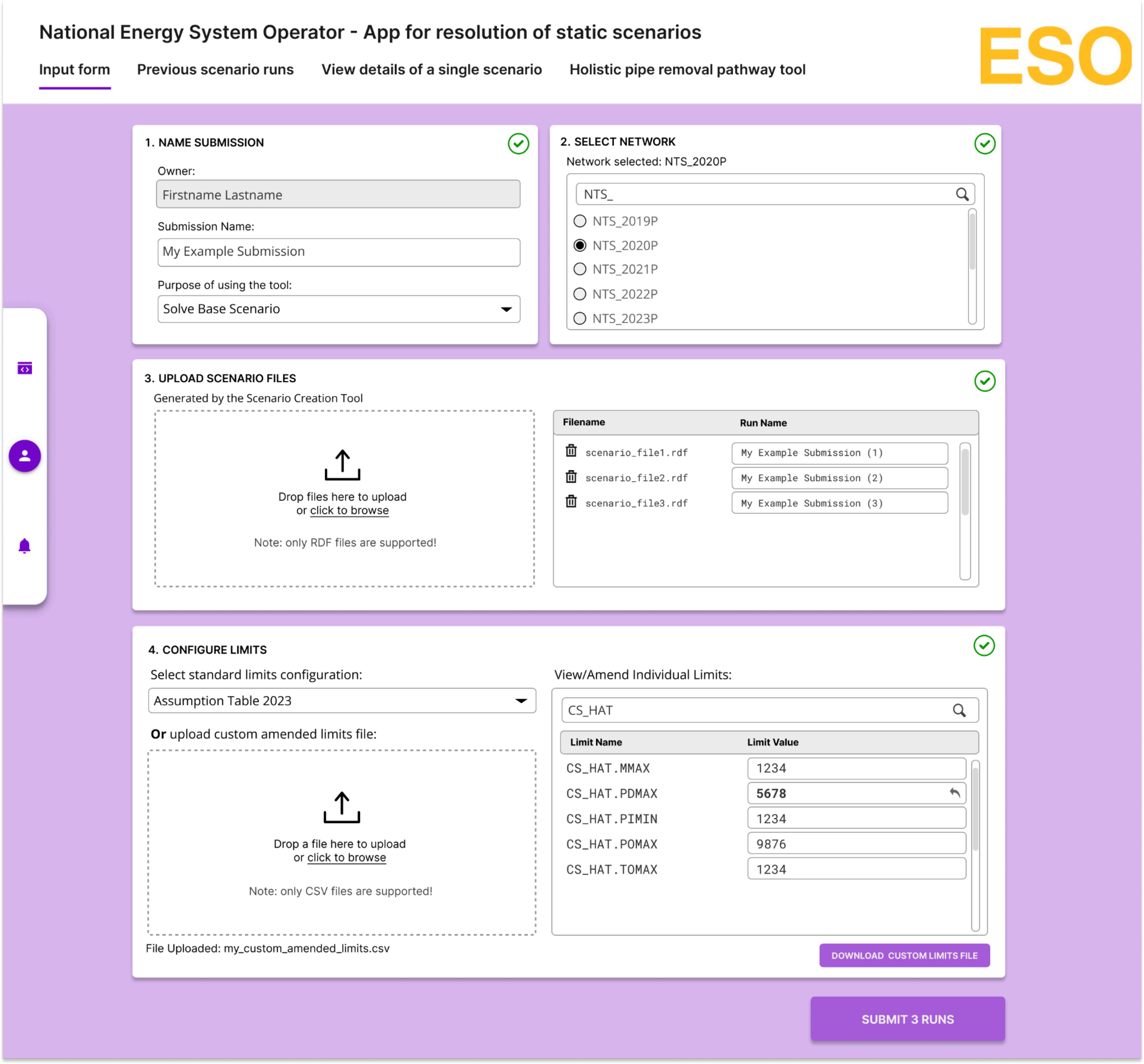


Table of past scenario runs page

National Energy System Operator - App for resolution of static scenarios

ESO

Input form | **Previous scenario runs** | View details of a single scenario | Holistic pipe removal pathway tool

OWNER: None selected
NETWORK: None selected
SCENARIO: None selected
ENTRY CAPABILITY LIMIT: None selected
EXIT CAPABILITY LIMIT: None selected

APPLY FILTERS | RESET | VIEW SELECTED SCENARIO | SAVE/DOWNLOAD

SUMMARY OF PREVIOUS SCENARIO RUNS

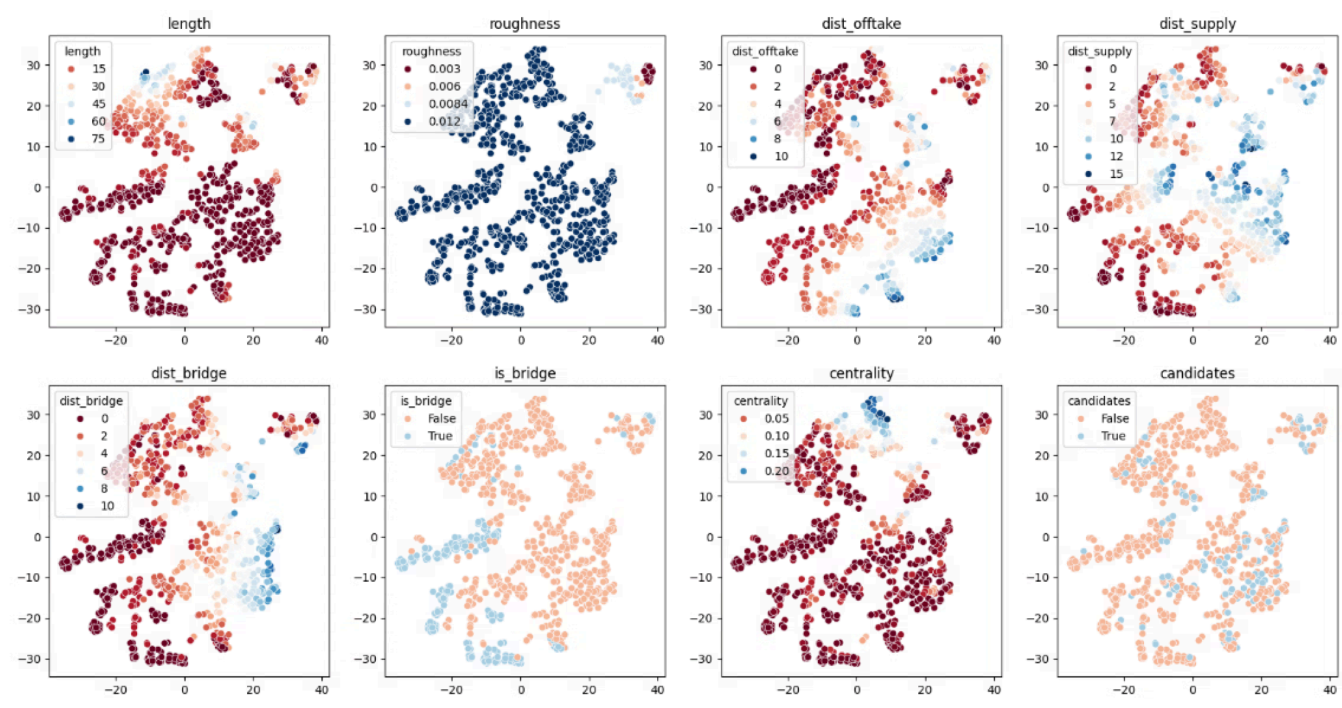
Overview of the last 30 runs

Scenario ID	Run submission timestep	Owner	Network	RIIO Zone	Run status	Top scoring solution	Exit capability limit	Entry capability limit	Run completion timestep	Scenario	Analysis type
19P_NC_EN_NE_18_D1A_19D3_CF_IC	09:22	XXX	National Grid	XXX	Done	Solved scenario RDF file	XXX	XXX	09:24	XXX	Base scenario
19C_NC_EN_SC_20_D5D_19D3_SP_HC	09:23	XXX	National Grid	XXX	Done	XXX	XXX	XXX	09:26	XXX	Base scenario
19C_NC_EN_SC_15_C5E_19D3_SP_HC	09:23	XXX	Scotia Gas	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	Entry capability analysis
19P_NC_EN_NE_17_D1A_19D4_CF_JK	09:25	XXX	National Grid	XXX	Done	XXX	XXX	XXX	09:28	XXX	Exit capability analysis
19P_NU_EN_NE_17_D1A_19F3_CE_IC	09:32	XXX	Northern Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	Base scenario
19C_NC_EN_SC_20_C16B_15D3_SP_HC	09:49	XXXNTS_2019P_AUDIT_ENTRY		XXX	Done	XXX	XXX	XXX	N/A		Base scenario
19C_NC_EN_SC_20_D1A_19D6_SJ_HC	10:15	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	10:16	XXX	XXX
19P_NC_DN_NE_18_D4A_19D3_CF_IC	10:18	XXX	National Grid	XXX	Done	XXX	XXX	XXX	10:20	XXX	XXX
19C_NC_EN_SC_20_D8C_14D4_SP_HC	10:27	XXX	National Grid	XXX	Deep	XXX	XXX	XXX	10:30	XXX	XXX
19P_NC_EN_NE_14_D1A_15D3_DF_IC	10:32	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_20_D1A_12E7_SP_HC	11:05	XXX	National Grid	XXX	Done	XXX	XXX	XXX	11:10	XXX	XXX
19P_NA_EN_NN_18_D5A_19D3_CF_IC	11:20	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	11:22	XXX	XXX
19P_NB_EN_NE_18_D1C_19D3_BF_IC	11:56	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	11:58	XXX	XXX
19P_KC_DN_NF_17_D1B_19D3_CF_IC	12:01	XXX	Scotia Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_19D4_AF_IC	12:54	XXX	Northern Gas	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_22_D1A_16D7_SP_HC	13:30	XXX	National Grid	XXX	Done	XXX	XXX	XXX	13:32	XXX	XXX
19P_MC_GN_NE_18_D1A_19D3_CF_IE	13:42	XXX	National Grid	XXX	Done	XXX	XXX	XXX	13:43	XXX	XXX
19P_NC_EN_NE_14_D1A_18D3_DE_IC	14:36	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	14:37	XXX	XXX
19P_LC_EK_NE_18_D1A_19D5_CF_IC	14:49	XXX	Northern Gas	XXX	Deep	XXX	XXX	XXX	14:50	XXX	XXX
19P_NC_EN_ND_12_D1A_15D3_DB_IC	15:02	XXX	National Grid	XXX	Done	XXX	XXX	XXX	15:04	XXX	XXX
19P_NC_EN_NE_18_A1B_15D5_CF_IC	15:22	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	15:26	XXX	XXX
19C_NC_EN_SC_19_D1A_18D5_SP_HC	15:39	XXX	National Grid	XXX	Done	XXX	XXX	XXX	15:41	XXX	XXX
19C_NC_EN_SC_18_D1A_19D3_TP_KC	16:00	XXX	National Grid	XXX	Done	XXX	XXX	XXX	16:02	XXX	XXX
19P_ME_EM_NE_11_D1A_19D3_CF_IC	16:05	XXX	National Grid	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_16D4_EF_IC	16:12	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_20_D1A_19D3_SP_HC	16:38	XXX	Scotia Gas	XXX	Done	XXX	XXX	XXX	16:39	XXX	XXX
19C_NB_EN_SE_BASE_D4A_SP_ST_NEW	16:45	XXX	Northern Gas	XXX	Done	XXX	XXX	XXX	16:48	XXX	XXX
19P_KC_CN_NE_18_D1B_19D3_CF_ID	17:06	XXX	National Grid	XXX	Cancelled	XXX	XXX	XXX	N/A	XXX	XXX
19P_NC_EN_NE_18_D1A_19E5_UK_IC	17:37	XXX	National Grid	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX
19C_NC_EN_SC_BASE_D1A_SP_ST_NEW	19:45	XXX	Scotia Gas	XXX	Ongoing	XXX	XXX	XXX	N/A	XXX	XXX

09.5 Spectral Clustering Method:

The first step involves constructing a *Laplacian matrix*, $L[i, j]$, where i and j represent pairs of nodes in the graph. This matrix encodes both the structure of the graph and the flow dynamics between nodes, where diagonal elements capture node degrees and off-diagonal elements indicate direct connections. Dimensionality reduction is then applied to the Laplacian matrix to create feature vectors, which describe the graph structure in a lower-dimensional space, making patterns more identifiable. This approach allows us to determine the optimal number of components, $k_{optimal}$, by detecting the largest drop in eigenvalues— a clear indicator of significant connectivity patterns[14]. With this optimal component count, we used K-means clustering on the feature vectors to group nodes with similar connectivity profiles. We also explored Agglomerative Clustering, but it did not outperform the simpler K-means approach in our experience.

09.6 Candidate for removal identification:



Sup_figureXX: TSNE projection of the variables extracted in the gas network graph.

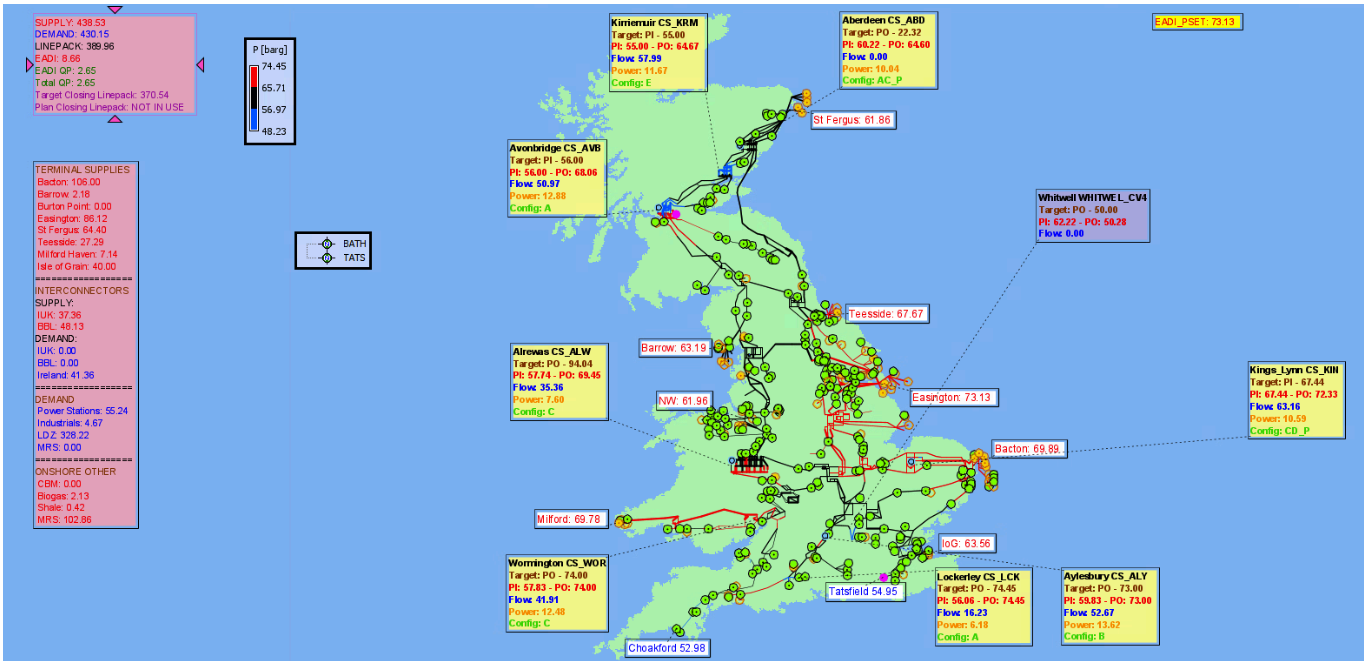
[14] Zelnik-Manor, L. and Perona, P. (2004). Self-Tuning Spectral Clustering. [online] Available at: https://proceedings.neurips.cc/paper_files/paper/2004/file/40173ea48d9567f1f393b20c855bb40b-Paper.pdf.

09.6 BSS hyperparameters

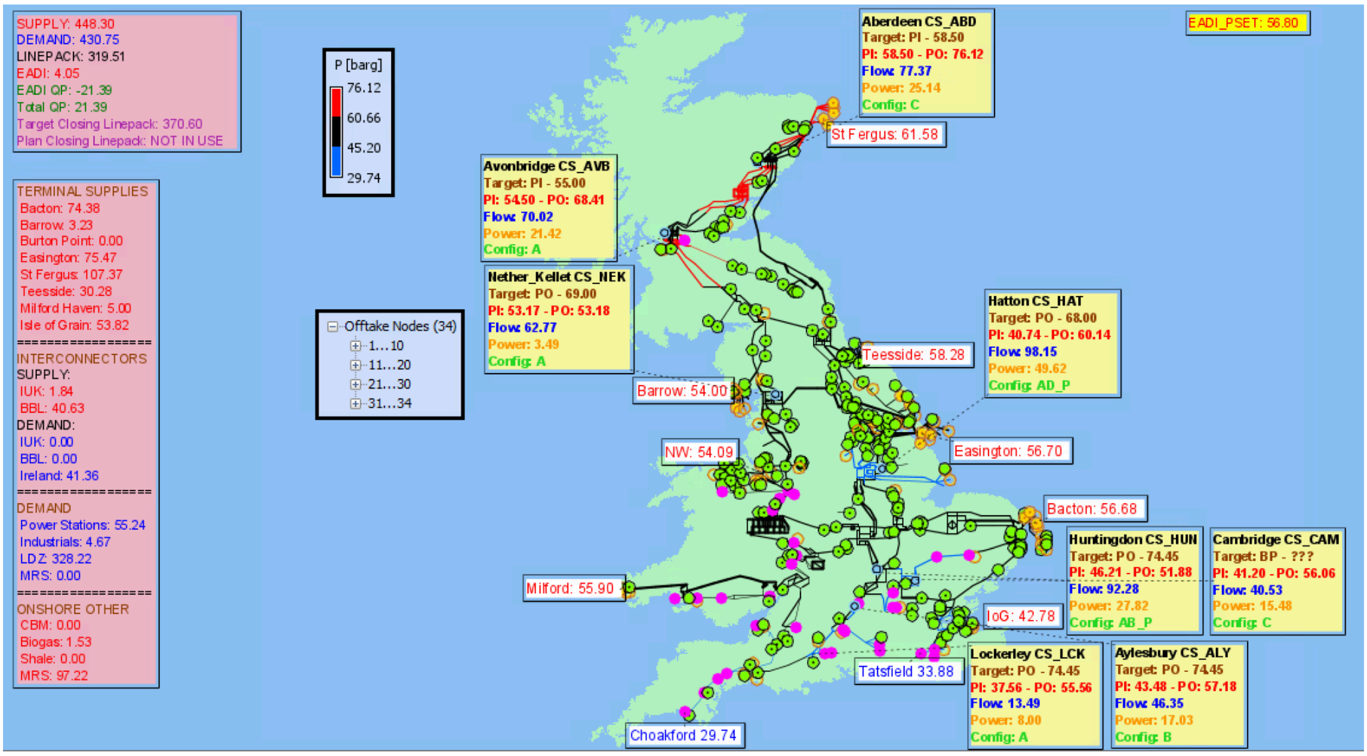
Name	Best Value	Description
initial_population_size	75	The number of solutions used for the initial population
generation_size	50	The number of children created in each generation
elder_pool_size	10	The number of solutions stored for use as elite immigrants
num_parents	25	The number of top-performing parents used to create the next generation children
elders_added_per_generation	2	The number of elite immigrants added to the population per generation
randoms_added_per_generation	2	The number of immigrants randomly selected from historic solution data to add to the population per generation
parents_kept_per_generation	2	The number of parents that are kept from one generation to the next
initial_population_method	“spf”	The method used to generate the initial population; the best-performing uses the SPF to generate the population
crossover_method	“multijunctions”	The crossover method that is used to combine parents to create children.
pset_mutation_probability	0.2	The probability of mutating the EADI PSET value
pset_mutation_scale_factor	0.1	Determines the scale of the mutation to the EADI PSET value
num_supply_node_mutations	2	The number of supply node flows to attempt to mutate
supply_mutation_probability	0.5	The probability that a supply node flow mutation attempt will succeed
supply_node_scale_factor	0.1	Determines the scale of the mutation to the supply node flow
num_continous_mutations	10	The number of continuous asset-property pairs (e.g. compressor setpoints) to attempt to mutate
continuous_mutation_probability	0.5	The probability that a continuous mutation attempt will succeed
continuous_mutation_scaling_type	“gaussian”	The type of distribution used to randomly generate a value by which a continuous property can be scaled during mutation
mutation_scaling_factor_lower	0.5	The lower bound for continuous mutation scaling factors when using a uniform distribution
mutation_scaling_factor_upper	1.5	The upper bound for continuous mutation scaling factors when using a uniform distribution
mutations_scaling_factor_gaussian	0.25	The standard deviation of the Gaussian distribution used to generate mutation scaling factors for continuous mutation.
num_categorical_mutations	5	The number of categorical asset-property pairs (e.g. compressor operation modes, valve modes) to attempt to mutate
categorical_mutation_probability	0.5	The probability that a categorical mutation attempt will succeed
num_multijunction_mutations	3	The number of multi-junctions to attempt to mutate
multijunction_mutation_probability	0.25	The probability that a multi-junction mutation attempt will succeed
mutation_weighting_threshold_lower	30	When *weight_by_score* is on, this is the threshold score below which the probability of mutation is decreased

mutation_weighting_threshold_upper	1e10	When *weight_by_score* is on, this is the threshold score above which the probability of mutation is increased
setpoint_range_lower	0	The lower bound of random setpoint values to select during a mutation where a compressor station or regulator has been turned on
setpoint_range_upper	100	The upper bound of random setpoint values to select during a mutation where a compressor station or regulator has been turned on
weight_by_score	False	Whether or not the probability of mutation and crossover should be weighted by how well the solution is currently performing
compressor_control_modes	["PO", "PI"]	The compressor control modes that the BSS is allowed to use
score_retries	3	How many times to attempt an evolution step for a child, where the condition for a successful attempt is that the child solution runs successfully in SIMONE

09.7 BSS results in SIMONE UI

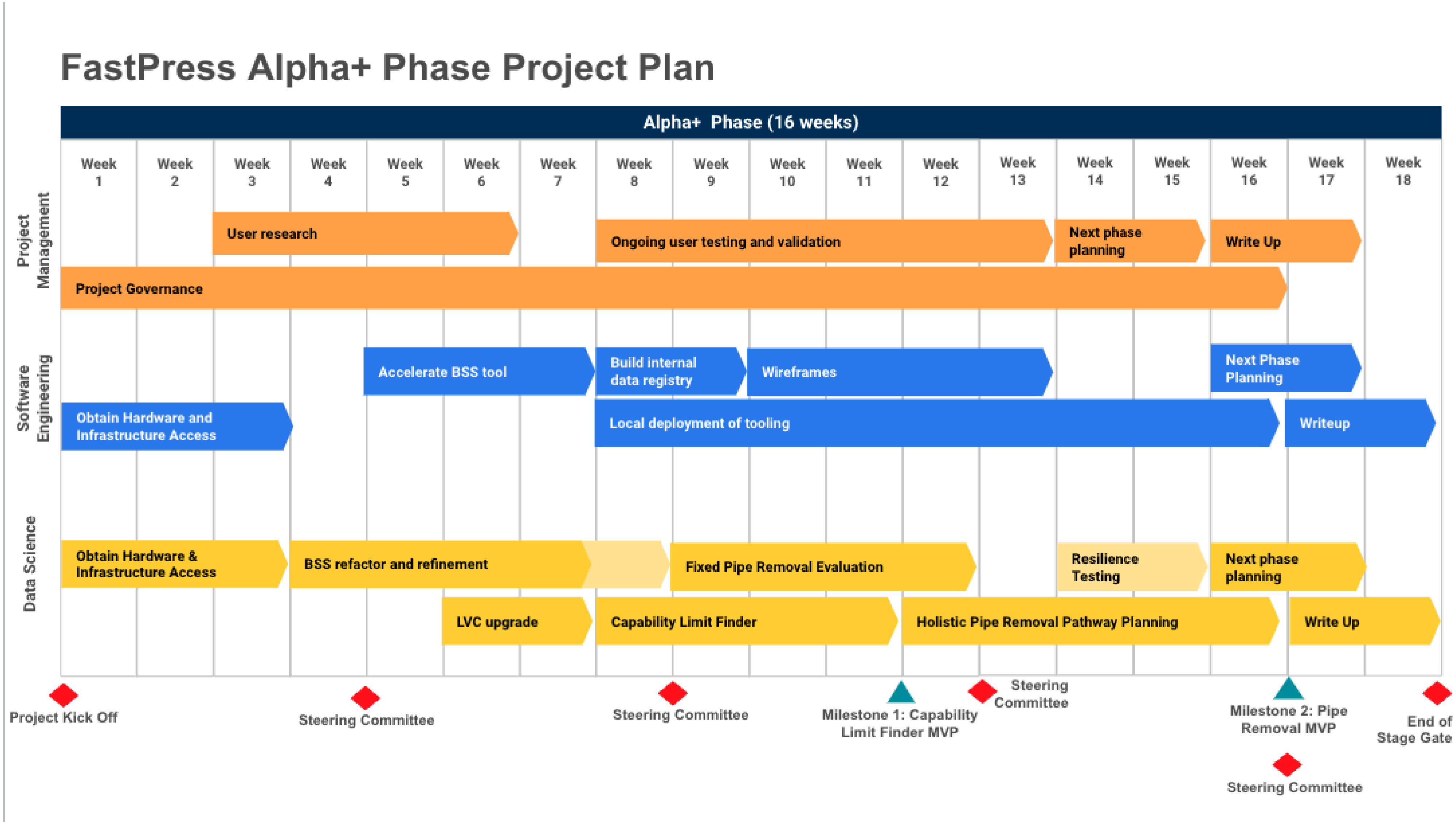


A SIMONE UI screenshot of a peak gas day scenario after running the scenario through the BSS.

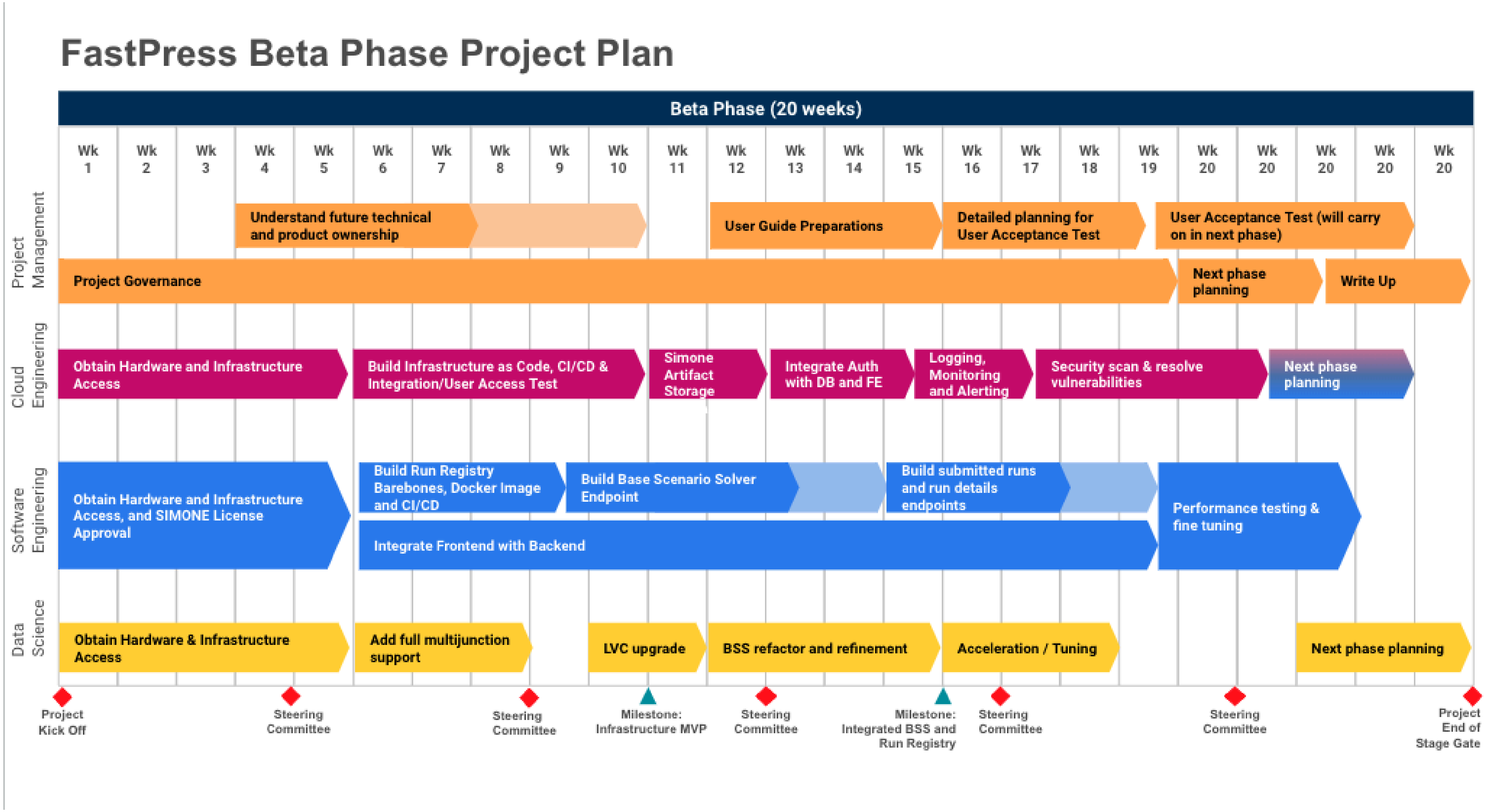


A SIMONE UI screenshot of a peak gas day scenario after running the scenario through the SPF.

09.8 Alpha+ plan



09.9 Beta plan



09.10 Glossary of Terms

BFS	Breadth First Search
BSS	Base Scenario Solver
CLF	Capability Limit Finder
CRV	Compressors, Regulators, Valves
DESNEZ	Department for Energy Security and Net Zero
EDA	Exploratory Data Analysis
FDRR	Fixed Pipe Removal Recommender
IPC	Inter Process Communication
LVC	Limit Violation Checker
MTE	Maximum Theoretical Error
NGT	National Gas Transmission
NLP	Natural Language Processing
PCA	Principal Component Analysis
PoC	Proof of Concept
RIIO	Revenue Incentives Innovation Outputs
SCT	Scenario Creation Tool
SoP	Standard Operating Procedure
SPF	Start Point Finder