Public

FastPress Project Alpha+ Report

Faculty

August 2025

Contents

1. Executive summary

The Alpha+ phase of the FastPress project successfully developed an automated, algorithmic approach to solving static gas network scenarios using the SIMONE software. Building on previous work, this phase advanced key components, notably the Base Scenario Solver (BSS) and the Limit Violation Checker (LVC), to a pre-production level. The BSS, which utilises a genetic algorithm, can now consistently and reliably solve a range of test scenarios at a performance level comparable to manual analyst times.

Additionally, two new features were developed to a proof-of-concept (PoC) stage: the Capability Limit Finder (CLF) and Resilience Testing (RT). The CLF automates entry and exit capability analysis, demonstrating its ability to find the maximum network capacity across all RIIO zones and various demand levels. The RT feature showcases the tool's ability to reconfigure solutions and solve scenarios even when specific assets are unavailable.

To facilitate user interaction and feedback, a Local Component Demonstrator (LCD) was deployed, providing a user interface that operates locally on an analyst's laptop. This has been instrumental in refining the tool's processes and addressing subtle inaccuracies. A significant engineering milestone was also achieved by successfully installing and running SIMONE on a Windows Azure Container instance, which de-risks a full cloud deployment in the future.

Looking ahead, the Beta phase will focus on transitioning these features to the current network model, improving efficiency, and formalising the tool for production use within the NESO cloud environment. This will include developing new features like a Transient Scenario Solver and an Infrastructure Timeline Optimisation tool. The work completed in Alpha+ provides a robust foundation for this future development, promising a more productive and efficient workflow for gas analysts that can ultimately be translated into a more efficiently managed gas network and reduced costs to customers.

2. Introduction

2.1. 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

[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

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.

2.2. Project Context

The gas analyst team at NESO undertakes various projects to understand the future performance of the gas network under different scenarios. This work informs reports such as the GNCNR. This report is comprised of numerous charts, generated from analyst work on entry and exit flows across RIIO zones. Each chart necessitates solving many gas scenarios to create the underlying data, a process that can take months for the entire report. From an analyst's perspective, this fundamentally involves flow adjustment and scenario solving, which was our primary focus. To demonstrate the feasibility of an algorithmic approach, we concentrated on static scenarios and did not address transient scenarios at this stage.

2.3. Problem areas to be addressed

While the current process successfully generates content for the GNCNR, it is hindered by several challenges. Enhancements to this process could significantly improve the speed and quality of the output. Additionally, these improvements would enable NESO to investigate a wider range of scenarios and enhance planning for emerging situations, such as the integration of hydrogen.

We have built tooling to address two key areas for analysts - solving scenarios and iterating flow. While each of these have specific pain points that are addressed later in this report, they both generally suffer from being highly manual and involved processes that take analysts significant time to set up and perform. This makes them prime candidates for algorithmic approaches that can use AI techniques to find results with limited analyst involvement, freeing them up to focus on more complex work.

2.4. Report Objectives

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

- **Outline Key Workflows:** Describe the "Solving Scenarios" and "Flow Analysis" workflows performed by gas analysts, highlighting current processes and challenges.

- **Explain Solutions and Features**: Detail the developed solutions and features supporting gas analysts in these workflows, including their status, performance, and future development areas.
- **User Interface:** Detail the user interface we have designed for the tool and the process of developing this with users. Additionally demonstrate the Local Component Demonstrator, a simplified demonstrator, intended for direct execution on a user's machine.
- **Engineering Progress:** Explain the engineering work completed to date and the progress towards a representative deployment to NESO's cloud.
- **Explore Future Development:** Outline the future development roadmap of the tool that would take place in the subsequent Beta phase.

Overall, this report provides a comprehensive overview of the tool developed to date, its features, how users will interact with it, and the benefits of its continued development.

3. Solving Scenarios

3.1. Current Process

Scenarios are evaluated using SIMONE, a hydraulic modeling tool. The process begins with the creation of scenarios using a dedicated Scenario Creation Tool (SCT), which is an Excel spreadsheet that generates SIMONE-compatible scenarios based on forecasted supplies and demands for the day of interest. Each scenario represents a given 'gas day' under a set of Future Energy Scenarios (FES) forecasting assumptions, where the year is split into gas days in decreasing order of demand level. This means that a Day 1 scenario represents the highest demand day for the year, and a Day 365 scenario represents the day with the lowest forecasted demand.

To find if the network can support a given forecasted gas day, static and transient scenarios are solved for that day, with the static scenario being addressed first. In this project, the scope was limited to investigating the process of solving static scenarios only.

Once a scenario is created, it is opened in SIMONE. The analyst then enters a cyclical process:

1. **Executing the scenario**: The analyst runs the simulation to see the initial results.
2. **Checking for violated limits**: They review the results for any pressure requirements or limits that have been breached.
3. **Adjusting asset settings**: The analyst modifies the asset settings within the model in an attempt to resolve the violated limits.

This cycle continues until one of two outcomes is reached: either a solution is found where all pressure requirements are met and no network limits have been violated, or the analyst determines that a solution is not possible. In the latter case, the analyst identifies the root cause of the constraint by inspecting the failing scenario in SIMONE.

After completing this analysis, the analyst creates an audit document detailing their findings. This document and the solved scenario are then handed over to a senior analyst. The senior analyst reviews the audit and validates the original analysis. They do this by independently reproducing the steps and attempting to improve upon the proposed solution.

3.2. Issues and problems faced

Solving scenarios presents several challenges, primarily because the process is time-consuming and manual. A significant limitation is the lack of a system for reusing settings from previously solved scenarios, which forces analysts to start from scratch each time. Additionally, checking if a scenario is solved is a slow and manual task because not all necessary settings are included within the SIMONE software itself, so analysts often must cross-reference additional documents and spreadsheets. On top of the challenges faced by analysts when actively solving scenarios, SIMONE licenses come at a significant expense and are under utilised when analysts are not actively engaged in work.

3.3. Solutions developed

Our solution aimed to solve these issues by designing a system that could:

1. Incorporate past scenarios as starting points for solving new scenarios
2. Automates the checking of whether solutions have passed
3. Automatically find and apply settings, iterating until finding a solution to a scenario
4. Can be run in the background without analyst oversight

The solutions we developed build on each other, and while some can be used individually as shown later, in combination they add up to a system that automates much of the static scenario solving process. Our solution broadly relies on trying to evolve a number of potential settings that could solve a solution using a genetic algorithm. This is then supported by tooling shown below.

Feature	Purpose	Alpha phase status	Alpha + phase status
Limit Violation Checker (LVC)	Automate the checking of whether a solution has passed	POC	Pre-production
Base Scenario Solver (BSS)	Automatically find settings to solve a scenario	POC	Pre-production
Resilience Testing (RT)	Limit the compressors available to a solution	N/A	POC
Start Point Finder (SPF)	Use historical data to suggest starting settings	Pre-Production	Pre-Production
Solution Scoring Tool (SST)	Extends LVC to give comparable scores	POC	Pre-production

Table 1: Solutions to scenario solving and their status at the end of the Alpha and Alpha + phases

3.3.1. Start Point Finder (SPF)

We describe the SPF in more detail in the previous phase report; there were no significant changes to this feature in this phase.

Briefly: to initialise our automated solving, we need a number of starting settings for a scenario. Random settings choices are likely to be invalid, since most combinations of settings are likely to be contradictory. The SPF tool leverages the fact that analysts have already solved a large number of diverse scenarios: it compares the new unsolved scenario to previous solved scenarios and returns the most similar ones with their settings - these settings then serve as the starting point for solving, as a form of informed guess. Comparing scenarios is done using their supply and demand flows - the values for these flows are treated as a vector of numbers for each scenario, and scored for how similar they are with cosine similarity, a technique used in Natural Language Processing.

3.3.2. Limit Violation Checker (LVC)

The Limit Violation Checker (LVC) is a core component used to define a solved scenario. A scenario is considered solved when it has a successful hydraulic solution in SIMONE and adheres to all operational limits.

The LVC is structured around a sequence of checks:

- The scenario must execute successfully in SIMONE.
- The actual linepack must be within 1 million standard cubic metres per day (mscm/d) of the target linepack.
- Supply flow must match off-take flow to two decimal places.
- No assets are breaching their limits.

If any of these checks are violated, the LVC flags the status of the scenario as having failed. A breakdown of specific failures is available for analysts to dig into the potential causes of the failure. For more detail on the outputs of the LVC, see the relevant section below.

The asset-level checks combine existing automated SIMONE checks with currently manual processes that otherwise involve cross-referencing the scenario results with reference assumptions documentation. The bulk of the LVC development was carried out during previous phases of the project; during this phase, the LVC interpretations were validated with analysts, and the pressure cover calculation was added, which had been out of scope previously.

For offtake points at Tatsfield and Choakford, the pressure limits are determined by reproducing the behaviour of the Pressure Cover Calculator, an Excel spreadsheet used by gas analysts. This involves calculating the correct combination of pressure covers based on whether the regional forecast CCGT demand is below the maximum CCGT demand and if any compressors are in use in that region.

3.3.3. Solution Scoring Tool (SST)

The Solution Scoring Tool (SST) is designed to evaluate algorithmic solutions beyond a simple pass/fail by assigning continuous scores based on the proximity of values to their limits. Scores for individual assets are combined into a single scenario score, with a failure threshold at 1. The final scores are balanced across linepack, supply and demand, and assets, and can also incorporate additional elements such as penalising the use of compressors such that a scenario using fewer compressors has a better score. In the most recent phase, the balancing of scores between the three primary error sources was improved by testing for the optimal balance that promoted the finding of a solution quickly, and adding scoring for QP mismatches.. For more information consult the Alpha report.

Asset	Property	MIN	MAX	ACTUAL	SCORE
CS_HAT~D	RPM	2880	4800	4421	0.45
CS_HAT~D	RPM	2880	4800	5000	1.52

Table 2. Relation of score to the actual value and the corresponding limits - a score below 1 means the actual value from the result has met the limits. The top row is from a solved scenario, and the second row modifies this to demonstrate a failing score.

3.3.4. Base Scenario Solver (BSS)

The Base Scenario Solver (BSS) is a complex tool designed to programmatically identify optimal settings for SIMONE scenarios. It operates by exploring a range of possible settings, applying them to a given scenario, and then executing the scenario and scoring the results. This iterative process continues until a satisfactory solution is found.

The methodology of the BSS incorporates several other specialised tools and encodes analyst knowledge. Key components include the Limit Violation Checker, the Solution Scoring Tool (SST), and the Start Point Finder (SPF). The BSS iteratively refines settings using a genetic algorithm, which forms the core of its problem-solving approach.

In its previous developmental phase, the BSS consistently improved results but had not yet achieved a complete scenario solution. The focus of the most recent phase was on:

1. Using a staged solving strategy
2. Initialising compressors at useful setpoints
3. Separate, dedicated strategies for linepack and supply/demand matching

These improvements, coupled with updated assumptions in the Limit Violation Checker, led to the significant milestone of the BSS successfully solving scenarios.

At its heart, the BSS employs a genetic algorithm. This involves improving a population of candidate solutions (settings) by evaluating their performance using the SST. Well-performing solutions are then combined and mutated until a solve is reached. The SPF and SST play crucial roles as enablers for generating the initial population and serving as the fitness function for the evolving population, respectively. The BSS utilises custom methods for crossover and mutation to maintain multi-junction integrity. It also includes specialised functionalities for balancing linepack, supply and demand, and QP for the population members, all facilitated by the inherent flexibility of the genetic algorithm.

One of the main features we introduced in this phase was staged solving: a reduction in the search space that led to a more efficient and reliable solution to scenarios. Previously, the algorithm would be simultaneously trying to find the right assets to use, while also modifying their setpoints. This meant that there were a large number of options available to change at any given step, making it difficult to efficiently find a solution. We leveraged the fact that, based on observations of solved scenarios, assets are mostly used with setpoints at a typical value relative to their maximum capacity - for example, compressors in an output pressure control mode are often operating near their maximum achievable output pressure level - and introduced a staged approach, where the algorithm would first only focus on switching assets on or off (using their default setpoints). Once a good candidate was identified for asset configuration, the settings of these assets were fine tuned. Often however this was not necessary, as simply modifying asset configuration was sufficient to achieve a solution.

3.3.5. Resilience Testing (RT)

In addition to assessing the NTS capability in its 'intact' form (i.e. all assets are operational), the resilience of the NTS is assessed by solving scenarios under the assumption that a subset of assets are unavailable. Currently, this means solving scenarios as usual but with restrictions on specific compressors. For GNCNR analysis, 'allowed' compressors are selected based on their historic or projected availability, specifically those expected to be available more than 99% of the time.

Resilience testing has been added as a new feature in the BSS to support this analysis. This feature allows for the definition of a set of compressors that the algorithm is prohibited from using when solving a scenario.

This is implemented by modifying the strategies used by the genetic algorithm:

- Only scenarios with the same compressor usage are used to generate the initial population with the SPF.
- If there aren't enough matching historic scenarios to meet the required population size, the remaining slots are filled by mutating the available matching scenarios.

- During population evolution, mutations are only allowed within the specified operational modes.

This method is flexible, as the algorithm only requires a list of the allowed compressors, making it compatible with various strategies for defining which compressors are relevant.

A limitation of this approach is the amount of data available. Because the method requires using historic scenarios with matching compressor usage, there's often less data available. This can lead to a lower-quality initial population that may be less similar to the source scenario and where its quality cannot be guaranteed (if generated through mutation rather than from a known solution), making it more challenging to find a viable solution under resilience testing. However data from analysts of solved resilience scenarios would mitigate this issue in a deployed solution.

3.4. Performance

We show the ability for the BSS to solve all test data that was given to us, across a range of demand days (Table 3). This means that our algorithmic approach can find settings such that they satisfy the codified limits in the Limit Violation Checker. These results have also been validated with gas analysts. While this will be discussed in the next part of the report, it is worth noting here that we can also solve scenarios with flows modified from their base values, as used in capability analysis, extending the solving capabilities of the tool beyond base scenarios.

Scenario	Demand Day	Solved
23P_BS_26_D300A_23D3_ST_HC	D300A	✅
23P_BS_32_D300A_23D3_FS_HC	D300A	✅
23P_BS_34_D300A_23D3_FS_LC	D300A	✅
23P_BS_26_D46A_23D3_ST_HC	D46A	✅
23P_BS_32_D46A_23D3_FS_HC	D46A	✅
23P_BS_34_D46A_23D3_FS_LC	D46A	✅
23P_BS_26_D1A_23D3_ST_HC	D1A	✅
23P_BS_32_D1A_23D3_FS_HC	D1A	✅
23P_BS_34_D1A_23D3_FS_LC	D1A	✅
23P_BS_26_PK_23D3_ST_HC	PK	✅
23P_BS_32_PK_23D3_FS_HC	PK	✅
23P_BS_34_PK_23D3_FS_LC	PK	✅

Table 3. Solution status for every test scenario provided

3.4.1. Solution consistency and speed

A key question for the usefulness of the FastPress tool is not just *whether* it can solve scenarios, but how quickly and consistently it does so. Given the stochastic nature of the algorithm, randomness is controlled via random seeds, which specify how random numbers used in the algorithm will be generated. Different random seeds can cause slight variations in the solution process, leading to possibly different settings for a solution, and sometimes, for a solution not to be found, even when there is one available.

Failing to find a solution will occur due to certain vulnerabilities in the genetic algorithm, i.e. populations losing diversity too quickly and lacking the necessary settings (genes) from which to build a solution. While given a large enough population and good hyperparameters the genetic algorithm should be able to solve any solvable scenario, this introduces the second consideration: time. Large populations and hyperparameters that are best for exploring them can take prohibitively long to run. While there was no agreed target time in which to solve a scenario, it is clear that taking more than an order of magnitude longer than analysts greatly limits the benefits of the tool, as it would be hard to integrate into existing workflows.

To identify how consistently and quickly we could solve scenarios, we performed hyperparameter optimisation of the genetic algorithm to select parameters that found solutions as quickly and consistently as possible. We have identified two types of approach to solving scenarios, which favour speed or consistency respectively.

The first scheme (Approach 1) has a 100% consistency for day 300 scenarios across five random seeds and can achieve this in a matter of minutes - see Figure 1. For more complex days however the consistency drops off while the speed to solve - or fail to solve - increases to hours. This approach is based on first finding the optimal compressor strategy, assuming default compressor settings, before fine tuning the settings for the best strategy. However, for high demand days which require more compressors, there is likely more flexibility required in the solving pattern.

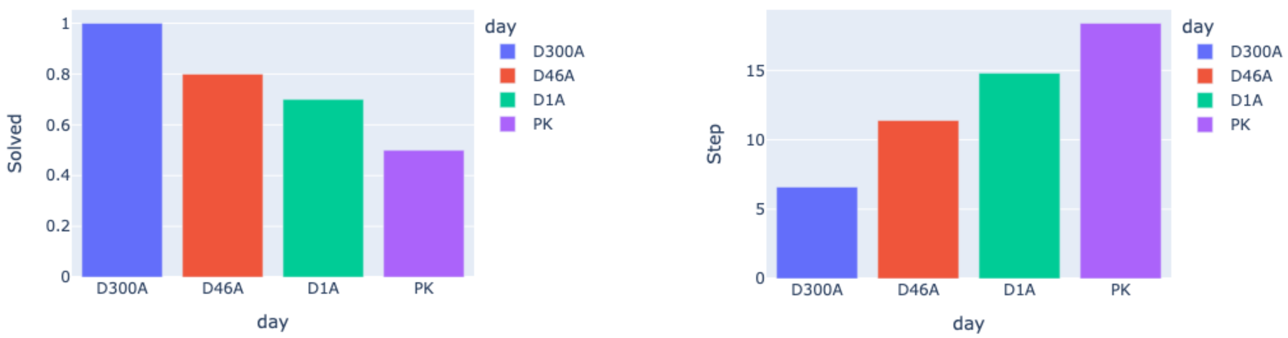

Figure 1. Solution consistency and speed for the initial approach to solving. Steps are units of how long it takes to solve a scenario, in this case 1 step is approximately 2.5 minutes with 6 licenses

Therefore we introduced a second approach, namely frequently alternating the selection of compressors with fine tuning their parameters. While this approach took longer for each attempt, it also had a 100% consistency in solving peak scenarios across five random seeds. This shows that ideal hyperparameters are dependent on the type of scenario that is being solved, and finding ideal hyperparameters will be an ongoing process for a deployed product.

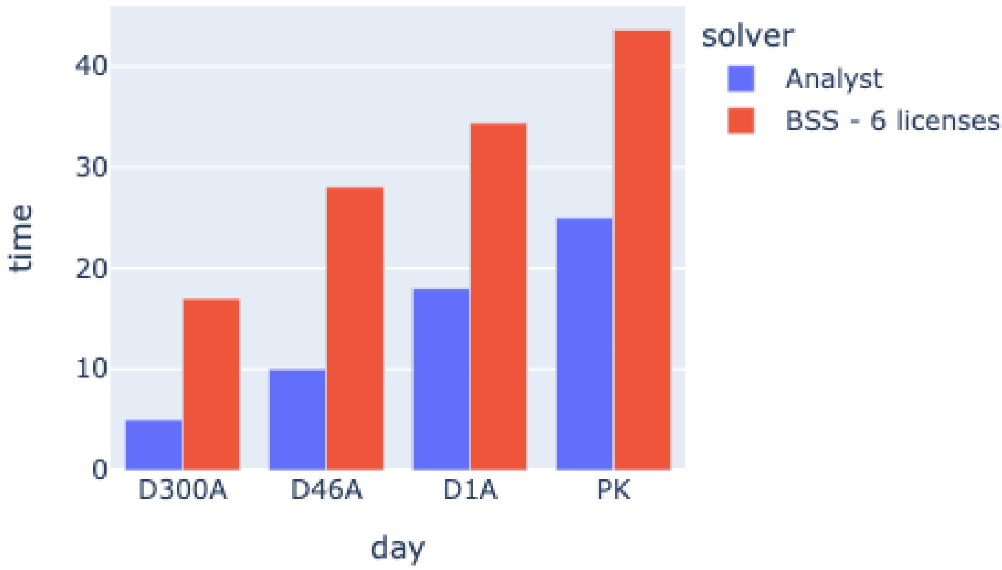

Figure 2. Analyst solve times compared to BSS solve times. The BSS with multiple licenses can solve base scenarios on the same order of magnitude as analysts

Compared to analyst solve times, we find our algorithm takes only slightly longer to solve scenarios (Figure 2). While it does use multiple licenses to achieve this, this still puts it at a level of performance that makes it possible to integrate into analyst workflows. Given 4 licenses per machine and accounting for retries, the BSS could solve scenarios across gas days at a rate of approximately 1 per hour per machine. Running 4 machines overnight for 14 hours brings us to a potential for 56 scenarios to be solved without interfering with analyst license usage during working hours. This could provide a significant boost to analysts and accelerate their workflows.

3.4.2. Scenario Solution Case study

We share the details of solving a peak day scenario to demonstrate the BSS function. We track the progress of the score of the best performing scenario for each generation[2], and can see improvement over generations until a solution is reached (score <1). Although the best scoring scenario plateaus at times, the population is still evolving during these periods.

[2] A generation is a single iteration of the BSS, where a group of proposed settings is evaluated and modified through crossover and mutation. For a complete definition of generations and more detail on the genetic algorithm, see the Technical Specifications deck and the Alpha stage report.

The algorithm used staged solving[3] to find a solution in 18 generations. Of the 18 generations, the first 15 only turned assets on and off, using default settings. This was not sufficient to find a solution, so the solver progressed to the next stage (16-18), where it fine tuned the settings of the best performing candidate from the categorical stage (see Figure 3).

We can also see the evolution of the linepack, and that of the number of compressors used in the best performing candidate. We see that linepack reaches a tolerable value in 5 steps[4], whereas the compressor values fluctuate between 10 and 12 compressors before settling on 10, for which refined settings were able to find a solution.

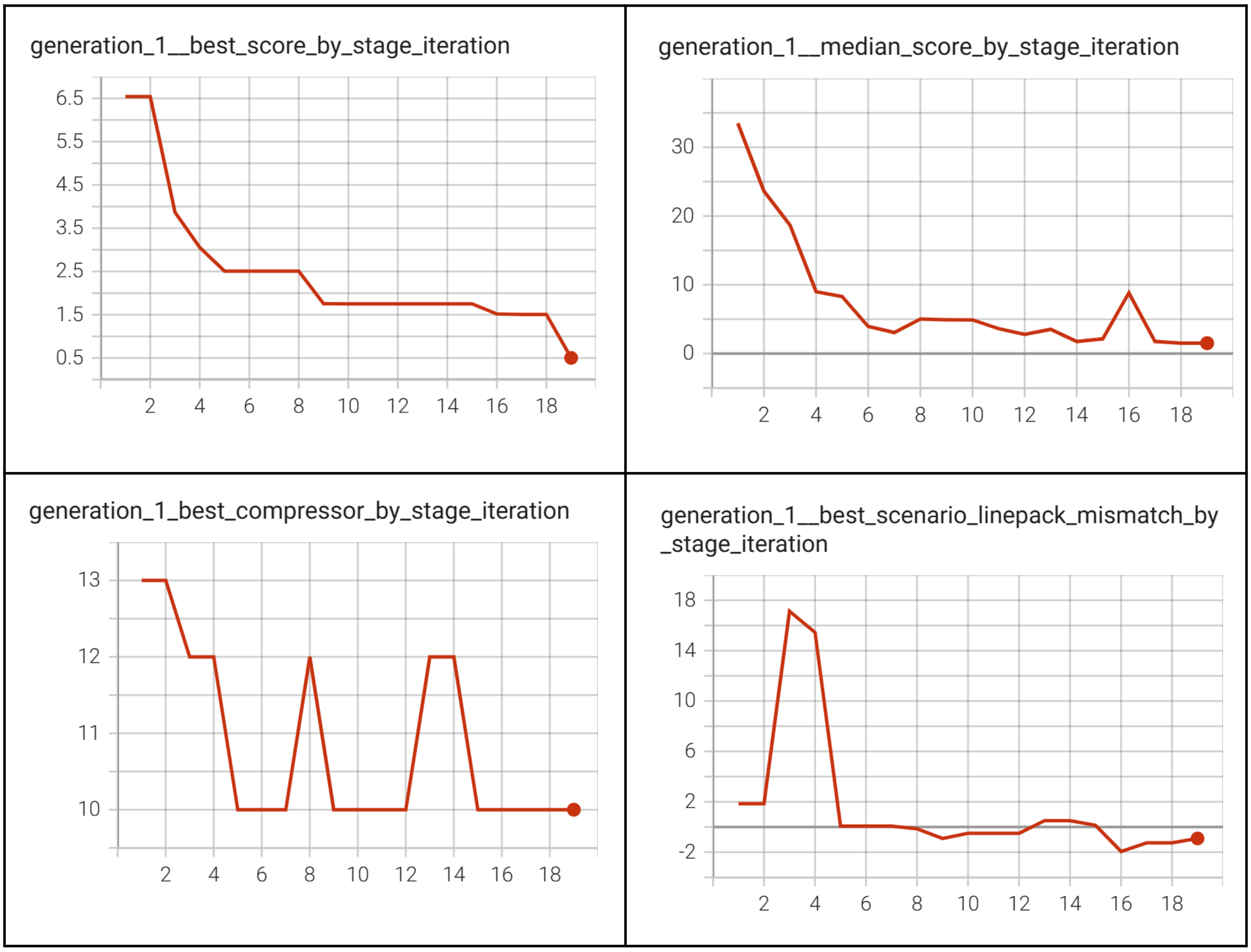

Figure 3. Clockwise: Improvement of the best score, Improvement of the population's median score, linepack mismatch of the best candidate, number of compressors in the best candidate.

These results were acquired with no penalty being applied to the number of compressors that were used by the BSS. Solving the same scenario with a compressor penalty yielded a solution in

[3] See BSS

[4] The allowed linepack mismatch is 1

a similar number of steps, but using only eight compressors. This demonstrates the flexibility of the algorithm to take into account multiple requirements, i.e. passing all LVC checks while also minimising the number of compressors used.

3.4.3. SIMONE licenses and solution speed

In the previous phase, we enabled the use of multiple SIMONE licenses to be used simultaneously[5] on the same machine to solve multiple members of a population in parallel. This can help accelerate scenario solving, at the cost of locking down several licenses. We show that using multiple licenses at once has a significant but diminishing effect on BSS runtimes (Figure 4). This can be used to inform the choice of how many licenses are used to solve a scenario.

If the aim is to solve the highest number of scenarios over a long time period, it is still best to use a single license per machine. However if there are constraints on the number of machines that can be used, or on the time available to solve scenarios (i.e. overnight), the use of multiple licenses on a single machine is favourable. It is worth noting that we also find significant speed differences with the hardware that is available for solving (i.e. developer vs non-developer laptops), and this is another avenue for accelerating solutions. In theory, larger speedups are a result of IO-bound SIMONE operations

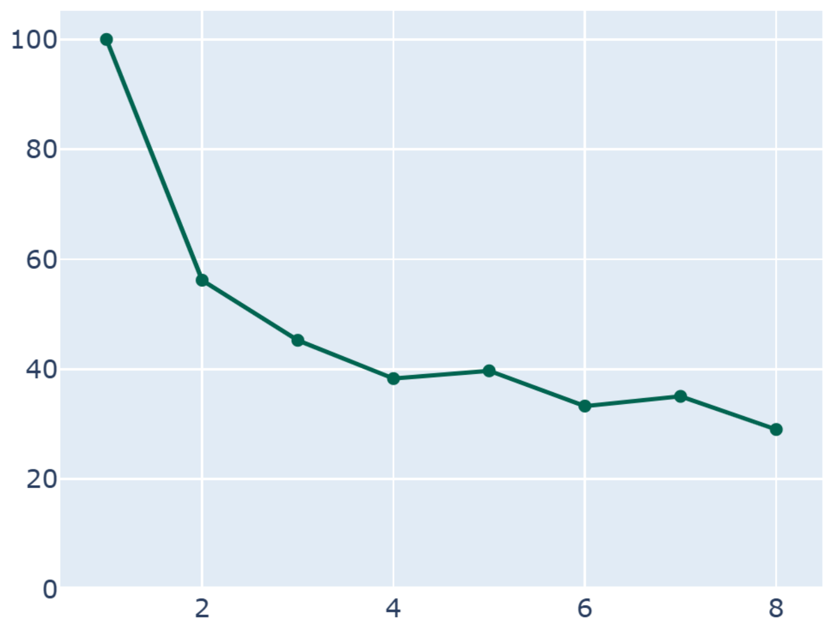

Figure 4. Normalised solution speed vs number of SIMONE licenses used (100% is the time it takes one license to iterate through a fixed number of generations)

[5] The acceleration of the solving through the use of multiple SIMONE licenses is enabled by the nature of the genetic algorithm with its many independent executions. It operates via a thread-safe pool of active SIMONE API connections.

3.4.4. Resilience Testing

To demonstrate the performance of our resilience testing capabilities, we took a scenario solved by the BSS, and specified that one of the compressors used in the solution was unavailable. The algorithm was then able to solve the scenario using an alternative configuration of compressors. This demonstrates the functionality of resilience testing, which will also allow for resilience testing at the flow iteration level, contributing to evaluating the capability of the network beyond its fully-intact state[6].

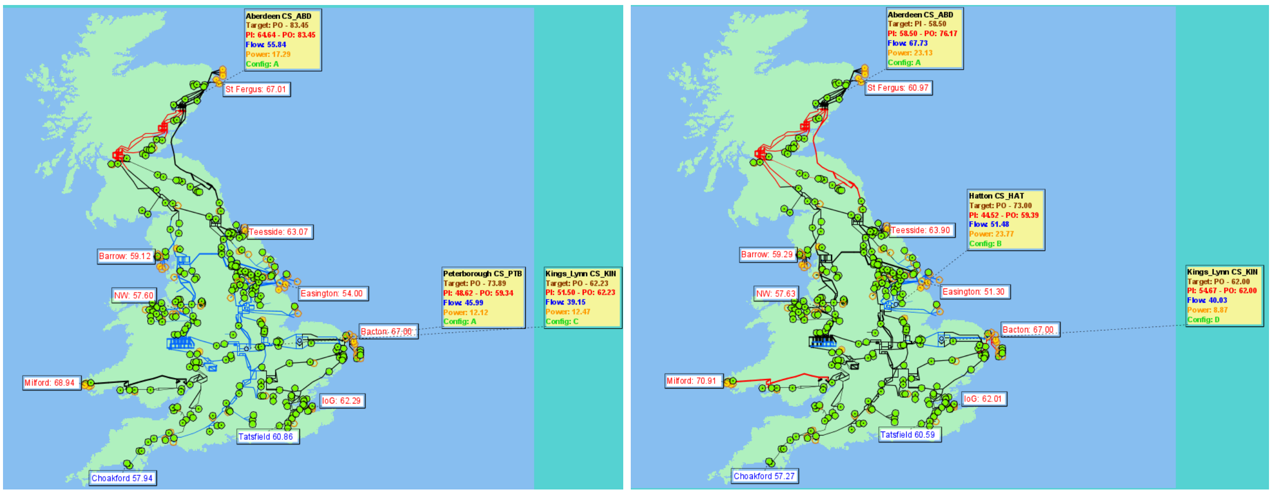

Figure 5. Screenshots of scenario solutions found by the base scenario, shown in the SIMONE UI, comparing an unrestricted solution that uses compressors in Aberdeen, Peterborough, and King's Lynn (left) and a resilience-testing solution where the Peterborough compressor station was unavailable, which uses compressor stations in Aberdeen, Hatton, and King's Lynn (right).

3.4.5. Time savings results

Given the fact that we can reliably solve most static scenarios, the time savings of the FastPress tool can be expressed as the fraction of time analysts spend solving these static scenarios. This is because the FastPress tool does not need to compete for resources with analysts, and allows a workflow where analysts focus on transient solutions while static scenarios can be solved en masse overnight and on weekends. Therefore, we propose the time-saving benefit of the tool to be the total analyst time spent on static scenarios. Unfortunately we do not have a figure for the fraction of time spent by analysts on static scenarios as part of their workload. The data we have from analysts on the time spent on individual scenarios has helped us validate that our tool can solve scenarios at a similar speed to analysts.

Another benefit of the tool is that base scenarios can be solved en masse before a project kicks off, getting analysts to the most important steps of flow analysis sooner. As part of ongoing project workflows, analysts could send static scenario jobs to run overnight or during the day while they work on other tasks, such as solving transient scenarios, auditing scenario solves,

[6] This corresponds to the non-intact capability lines in GNCNR flame charts

producing reports and ultimately work on solving more demand days to improve the quality of results.

3.5. Areas for further development

Several areas have been identified for future development to enhance the solution's performance and utility.

Hyperparameter Optimisation: To improve both solution quality and speed, further hyperparameter optimisation is required. This should be customised for different use cases, such as high- versus low-demand scenarios, base versus flow analysis scenarios, and intact versus resilience analysis. In a deployed tool this optimisation process would be an ongoing part of model maintenance instead of a one off analysis.

User Testing: Further user testing is needed to validate that the algorithmic solution meets domain-specific requirements. This will also help to optimise the solving process for strategies that are most useful to the users themselves, balancing priorities such as solution speed, quality, diversity, and support for user input.

Further development of Resilience Testing: The resilience testing feature was developed to a Proof of Concept (PoC) level in the current phase, demonstrating its feasibility. To become a truly useful tool, it requires further development in the following areas:

- Performance Testing: We need to test the feature's performance against realistic resilience testing requirements and to ensure good performance across a range of resilience testing experiments.
- Specific Hyperparameter Optimisation: As mentioned previously, specific hyperparameter optimisation for resilience testing is likely to improve its performance.
- Improving Low-Data Experiments: Performance exploration and improvement are needed for experiments where the Start Point Finder (SPF) data does not contain many, or any, scenarios that match the resilience requirements. Although the RT feature already supports the creation of artificial source data when data is limited, this generation process needs to be thoroughly tested and optimised.

Alignment with down stream tasks: While the BSS itself is a valuable tool, the capability it offers in automatically solving scenarios opens up the opportunity for further algorithmic approaches that build on top of it. Flow analysis, which is discussed in the next section, is a good example of how when these new algorithms are built, they require subtle modifications to the BSS to integrate with it effectively.

As the larger FastPress tool is expanded to new features which rely on using the BSS for automated solving, such as optimisation of pipe removal analysis, these will create the need for further refining the BSS in ways that maximise their effectiveness.

Transient solving: While the BSS has proven the capability for the automation of static scenario solving, the majority of analyst time is spent solving transient scenarios. Leveraging the experience of developing the BSS and applying it to creating a BSS equivalent to solve transient scenarios would tackle the last major hurdle in accelerating analyst workflows to enable them to work on more complex and impactful workflows.

3.6. Summary and Conclusion

Fundamentally, we have demonstrated that an algorithmic approach can independently solve static SIMONE scenarios consistently and on a time frame compatible with NESO workflows. We have refined this algorithm with analysts to align with their work flows and requirements, and leverage the existing SIMONE software for what it does best: modelling the gas network. Our algorithm provides a layer that removes several analyst painpoints, creating a product that will be the foundation of further automation efforts to reach a more productive and efficient organisation.

While there are further features and automations that can be developed, the automated solving of static scenarios is a significant step that tackles some of the most time-consuming work done by analysts. Features built on top of this one will likely be much simpler algorithmically, and will mostly rely on calling the BSS in new configurations to reach their goals, e.g. optimising pipe removal planning. Building the tool to this level is not an effort that will likely need to be replicated, besides extending it to transient solving; an extension with promising feasibility given the success of the static BSS.

The work done so far can already be integrated into analyst workflows (Figure 6), limiting the need for their direct involvement in static scenario solving and allowing for the automated batched running of static scenarios such as base scenarios for exit flow analysis. The tools we have built in this process are also individually useful, e.g. our Limit Violation Checker allows analysts to quickly check if their scenarios have been solved without the need to invoke the full BSS algorithm.

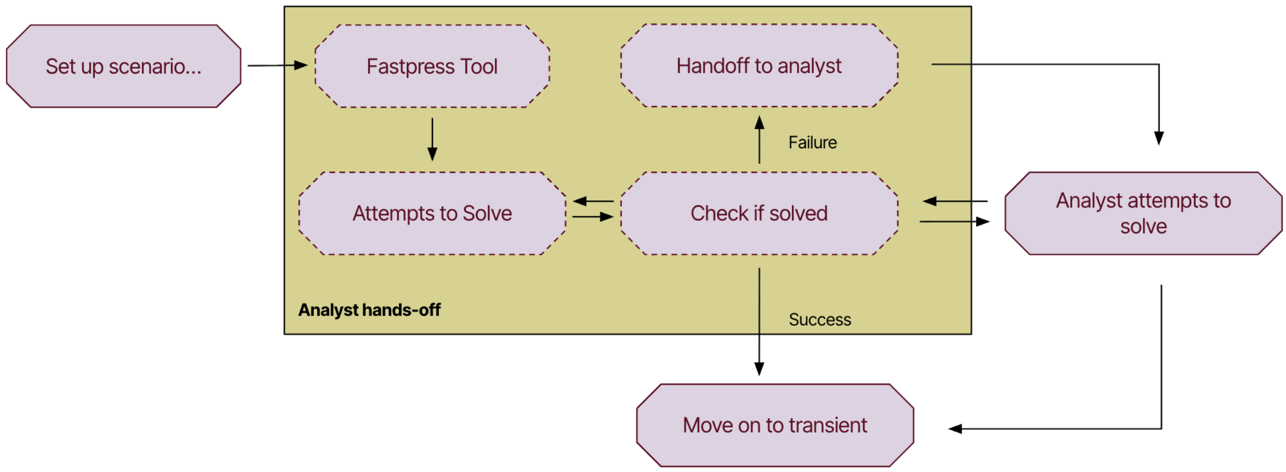

Figure 6. Possible new analyst workflow for static scenarios, with automated processes in the central green box interfacing with analyst steps

4. Entry and Exit Capability Analysis

4.1. Current Process

Analysis of entry and exit capability is conducted to assess the National Transmission System's (NTS) capacity, specifically the volume of supply and demand flow it can accommodate within each region. To get a full picture of the NTS capability for a given year, analysts evaluate these capacities at four different national demand levels: days 1, 46, 300, and peak, which cover a range from high to low demand. A single scenario represents one day, with fixed forecasting assumptions for supply and demand.

The process for assessing the capacity of a RIIO zone on a particular day begins with solving the base scenario, which uses the forecast supply and demand.[7]

If running exit capability analysis, the supply flow in the target zone is decreased to its minimum FES forecast flow (i.e. the D365 forecast flows). If needed, the scenario is then re-solved for this new flow distribution.[8]

Next, the flow in the target zone is increased: supply flow for entry analysis and demand flow for exit analysis. The size of this increase is based on the analyst's experience to find the capacity efficiently. To keep the total national supply and demand on the NTS fixed, the flow outside the target zone is balanced by removing a matching amount of flow. This adjustment process differs for supply and demand. For demand, the demand adjuster tool, an Excel spreadsheet, is used. Analysts copy demand flows from a SIMONE table into the spreadsheet, input the target RIIO zone and desired flow increase, and then copy the adjusted flows from the spreadsheet to a text file for upload to SIMONE. For supply, the process involves manually adjusting flows within SIMONE. Analysts add flow to supply terminals in the target zone and remove flow from terminals in other zones, referring to an Excel sheet called the Balancing Matrix to choose the 'least-interacting' balancing terminals. The flow adjustment follows a set of thresholds: for adding flow, terminals are first raised to their maximum FES forecast values, then to their maximum physical values, and finally arbitrarily higher if needed. For removing flow, terminals are lowered to their day 365

[7] If the base scenario cannot be solved (which may happen when running resilience analysis, where some compressors are considered unavailable and cannot be used to solve the scenario), the analysis continues as described, but with the flow in the target zone being *decreased* instead of increased, and flow is balanced elsewhere by adding instead of removing flow. In this case, the capability of the zone is determined to be the highest flow at which the scenario can be solved.

[8] Again, if the scenario cannot be solved under minimum supply assumptions, the analysis continues as described but with the flow in the target zone being decreased instead of increased.

minimum FES forecast values and then to zero, with the exception of Milford Haven, which has a physical minimum flow of 5 mscm/d.

After adjusting the flows, the analyst checks if the scenario still solves. If it does not, they search for a solution by modifying network asset settings. This iterative process of adjusting flows and re-solving continues until the scenario can no longer be solved. The final solved flow is then reported as the network's capacity.

This is a simplified description, and the process has significant nuance not included here. It's also important to note that gas analysts must solve both static and transient scenarios to confirm the true network capability, starting with following the above process for the static scenario and then repeating for the equivalent transient scenario.

This entire process must then be repeated for every RIIO zone in that scenario in order to complete the analysis of that gas day, and then again for every RIIO zone across another 3 gas days. The result then describes the capability of the intact network over one year; to evaluate the network under a resilience assumption, all RIIO zones across 4 gas days are then re-analysed under the required asset availability assumptions.

4.2. Issues and problems faced

The process of adding and balancing flows is a significant limitation as it is manual and time-consuming. Since this task must be repeated numerous times throughout the analysis, it accounts for a considerable amount of the total effort. It is also an error-prone process that is difficult to validate, risking unreliable and inconsistent results. Another issue is that the choice of flow jumps is challenging and relies heavily on analyst experience. If the chosen jumps are too small, analysts have to repeatedly distribute flow for each tiny increment, adding extra work. Conversely, if the jumps are too large, they lose time trying to solve high-flow scenarios that are beyond the network's actual capability.

4.3. Solutions developed

Several key components were identified as necessary to create a feature that supports capability flow analysis:

- **Automated Flow Adjustment:** This component takes a desired flow change and target zone as input, and automatically distributes that flow across the network, obeying the standard processes and requirements. Separate versions must be built for demand and supply flow adjustment, as their distribution processes differ.
- **Flow Recommender**: This tool generates a recommended amount of flow to add to a specific zone in the network.
- **Integration with the BSS**: This component adds the functionality to automatically re-solve scenarios if the flow adjustment causes a scenario to fail.

4.3.1. Automated Flow adjustment

A set of solutions were developed to automate the flow adjustment process, addressing the previous manual and time-consuming issues.

Demand adjustment

The existing demand adjuster tool, previously an Excel sheet, has been built directly into the codebase. This tool is used for exit capability analysis and follows a defined set of steps to adjust offtake flows. It first adds flow to the offtake nodes in the target RIIO zone, with the flow being distributed proportionally to the amount already present at each node. It then removes flow from other RIIO zones: the tool identifies the appropriate balancing zones for the target RIIO zone and calculates the amount of flow to remove from each zone, such that flow is removed proportionally to the zone flow as a fraction of the total flow across all balancing zones. Within each of these zones, flow is removed from individual offtake nodes proportionally to their existing flow.

Supply adjustment

The supply adjustment process was also automated. There is no supply adjustment tool available that is similar to the existing demand adjuster tool, so this component automates a process that is currently otherwise fully manual. This tool supports both entry flow analysis (adding supply flow to the target zone and balancing by removing flow from other zones) and exit flow analysis (removing supply flow from the target zone to meet minimum supply assumptions and balancing by adding flow to other zones).

The process for adding flow to supply terminals in the target zone is threshold-based. First, all relevant terminals are raised to their maximum FES forecast value, and then to their maximum physical values. All terminals must reach their FES forecast before any are raised above it. Flow is removed from supply terminals in other zones based on the balancing matrix and a set of minimum flow thresholds. The removal sequence for terminals is determined by the balancing matrix, which prioritizes terminals with the "least interaction," meaning those further from the target terminal are preferred. First, flows at the balancing terminals are reduced to their day 365 minimum FES forecast values, and then to zero (except for Milford Haven, which has a minimum flow of 5 mscm/d). All terminals must reach their D365 forecast before any are reduced below it.

When running entry flow analysis, the supply adjuster tool simply raises zone supply flows as required. For exit analysis, it first adjusts supply flow to minimum assumptions before using the demand adjuster to raise zone exit flows. These tools can also be exposed to users for manual use, allowing them to input a flow step and have the tool handle the redistribution.

Recommending flow

In addition to the automated adjustment tools, a flow recommender component was added. This tool automatically iterates through flow step increases until a scenario fails, such that it finds the flow capacity for a given set of settings. By default, it uses a step size of 1 mscm/d, but this is customisable. At each step, it runs the Limit Violation Checker (LVC). If a failure is detected, the recommender stops and returns both the highest flow achieved in a solved scenario and the flow that caused the failure, along with the corresponding failed scenario.

4.3.2. Integration with BSS

The developed flow adjuster components and BSS were integrated to create an automated Capability Limit Finder (CLF). This tool goes beyond simply finding the flow capacity for a single set of asset settings; it can also actively adjust scenario settings to increase that capacity, thereby providing a more comprehensive understanding of network capability.

To achieve this, a wrapper was built around the CLF and BSS, allowing the system to dynamically switch between the tools as needed to increase flow and adjust settings. We found that flow increases often only required changes to linepack and supply/demand balancing to reach a solution. To boost computation speed, a stripped-down version of the solver was created that is used when only these specific steps are required, rather than running a full optimisation across all settings.

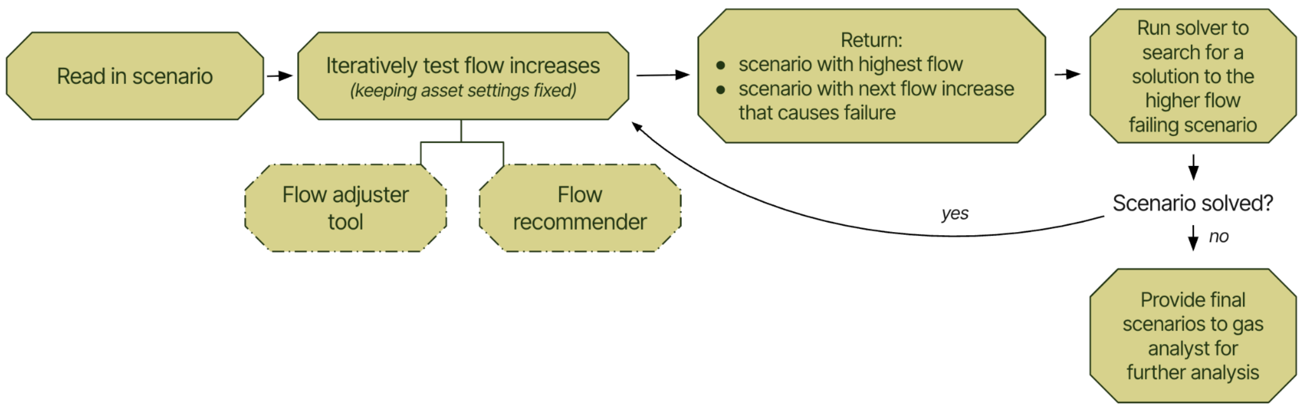

Figure 7. Flow diagram showing functionality of the automated CLF.

The resulting automated CLF works by iteratively increasing flow and adjusting settings until it can no longer find a solution for a higher-flow scenario. The tool then provides two outputs: the highest flow achieved with a corresponding solved scenario, and the flow that caused the failure along with the failed scenario itself. A gas analyst can then inspect this failing scenario to search for a solution. If a solution is found, it could be resubmitted to the automated CLF for further flow exploration.

4.4. Performance

The CLF has successfully demonstrated its ability to perform fully automated flow increases across all RIIO zones, various demand levels, and for both entry (Figure 9) and exit (Figure 10) capability analysis. The tool demonstrated its ability to independently handle several key steps that were previously manual:

1. Apply supply flow adjustments as needed for both entry flow increases and for applying minimum flow assumptions (Figure 8).
2. Solve minimum flow assumption scenarios before proceeding with exit flow analysis.
3. Apply demand flow adjustments.
4. Dynamically switch between increasing flow and solving scenarios, covering the full capability analysis process that is otherwise highly time-consuming.

A full range of results across all RIIO zones and demand days were achieved for entry capability analysis, but were less consistent for exit capability analysis due to the increased time and difficulty of solving scenarios under minimum supply assumptions, particularly at high demand levels. It is likely that this can be resolved in future by adjustments to the solving strategies and hyperparameters used by the BSS.

The flow increases achieved by the tool ranged from 3 to 60 mscm/d. While this performance is encouraging for a proof-of-concept tool, the flows it has achieved so far do not yet match the higher increases typically attained by experienced gas analysts, according to their feedback. However, there are clear next steps available for improving the tool's performance, from updating its functionality to better match analyst workflows to optimising the algorithm itself, which are discussed in the next section.

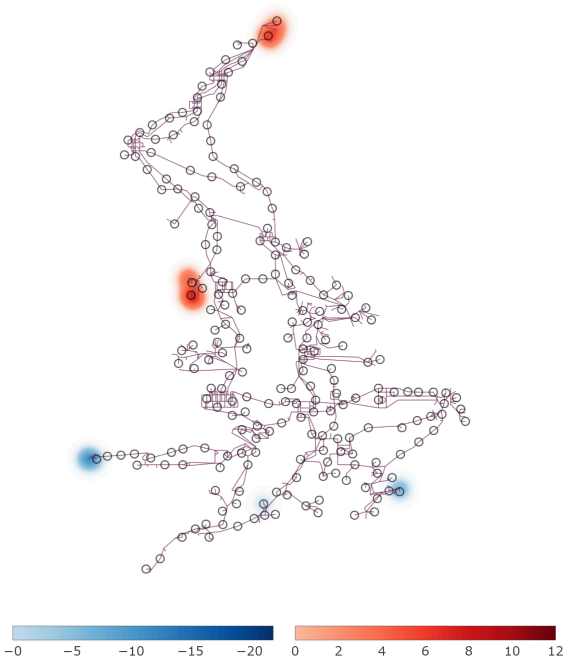

Figure 8. Map of where supply has been added (red) and removed (blue) by the CLF when applying a supply flow increase to zone 1 (Scotland) in an example scenario, demonstrating the automated flow adjustment component of the CLF.

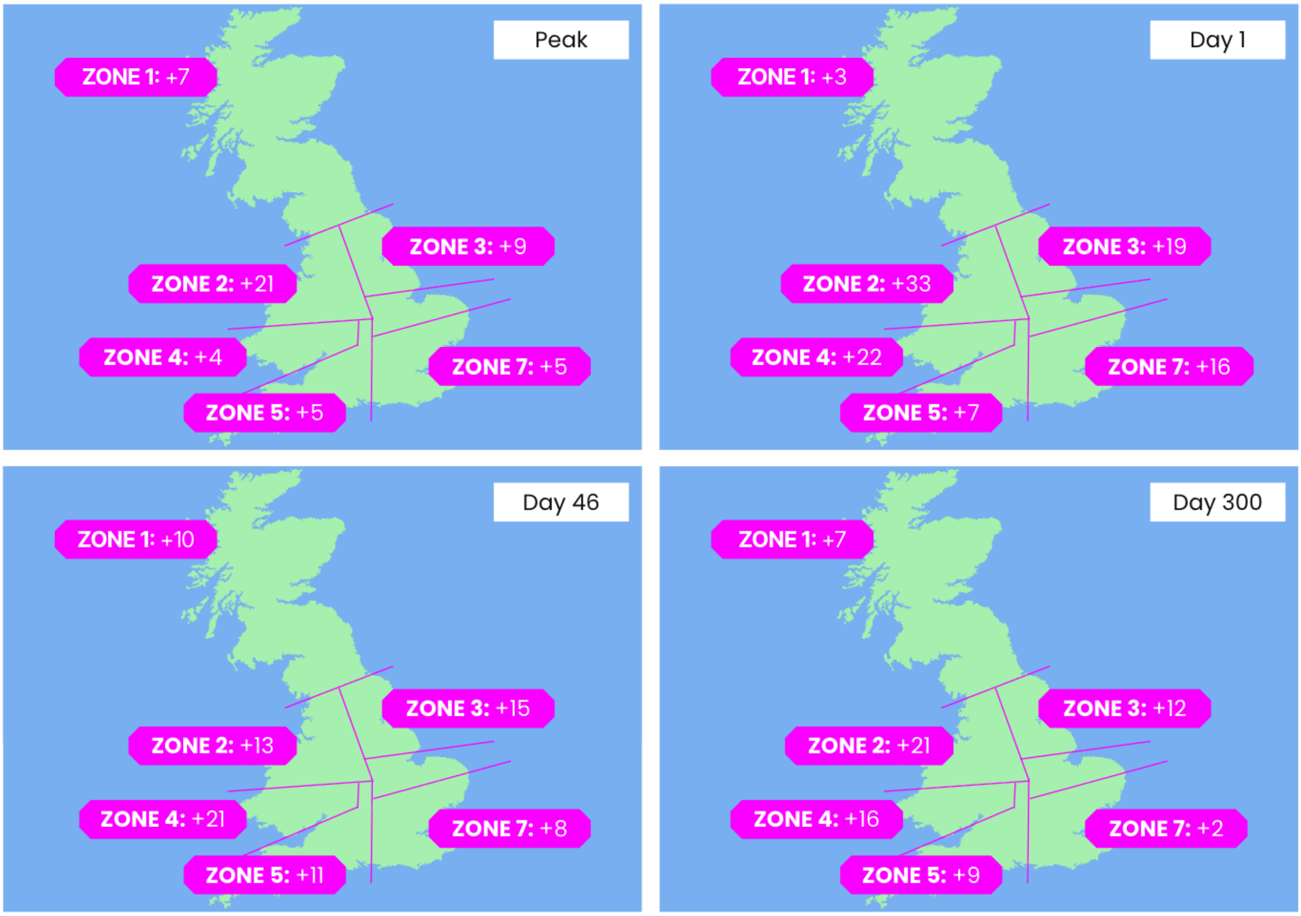

Figure 9. Representative examples of entry flow increases (relative to baseline flows) achieved by the automated CLF, showing performance across demand levels (indicated by the gas day used in the top right corner of each image) and across RIIO zones. Note that zone 6 is not included as it does not contain any entry terminals, so entry capability analysis is not applicable. Analysis was carried out for 2026 scenarios where supply and demand have been forecasted according to the FES assumptions of high continental and system transformation.

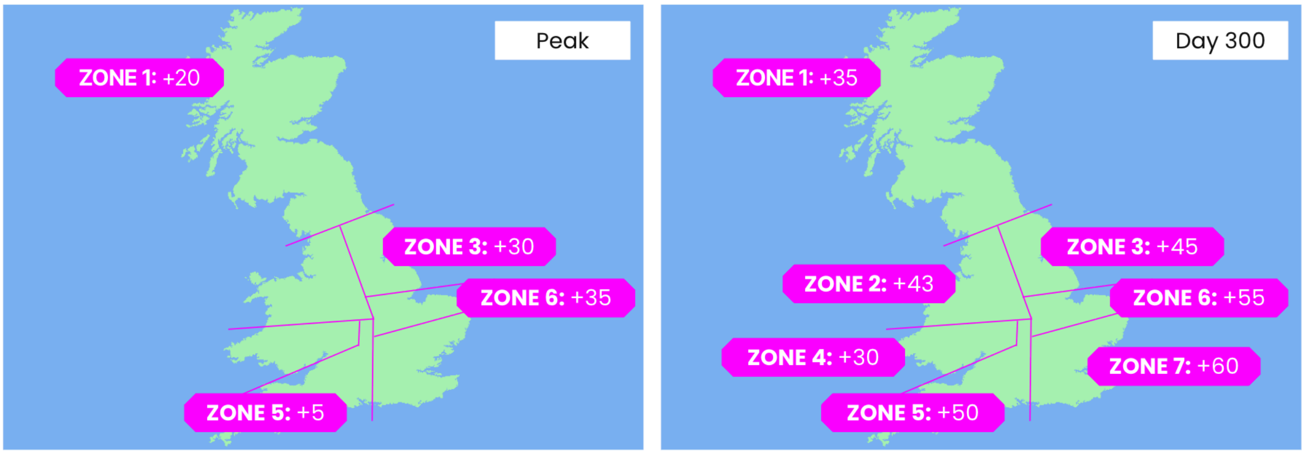

Figure 10. Representative examples of exit flow increases (relative to baseline flows) achieved by the automated CLF, showing performance across demand levels (indicated by the gas day used in the top right corner of each image) and across RIIO zones. Similar results are expected across intermediate demand levels, but have not been demonstrated yet due to time and computation constraints. Analysis was carried out for 2026 scenarios where supply and demand have been forecasted according to the FES assumptions of high continental and system transformation.

4.4.1. Automated capability analysis case study

This case study details a representative automated CLF run for a peak day scenario, focusing on entry capability analysis for RIIO zone 2. The automated CLF progresses through a series of CLF and BSS steps, iteratively increasing flows and re-solving scenarios as required.

The progress of the run can be visualised by tracking the zone entry flow and the scenario's score over time (Figure 11). In the steps where the flow is increasing, the CLF is automatically applying a set flow increase and balancing flows across the network. Conversely, in the steps where the flow plateaus, the scenario's score has risen above the failure threshold, and the BSS is attempting to re-solve it.

The process stops when the BSS can no longer find a solution for the scenario. At this point, the highest flow with a passing score is returned as the maximum achieved capacity for the zone.

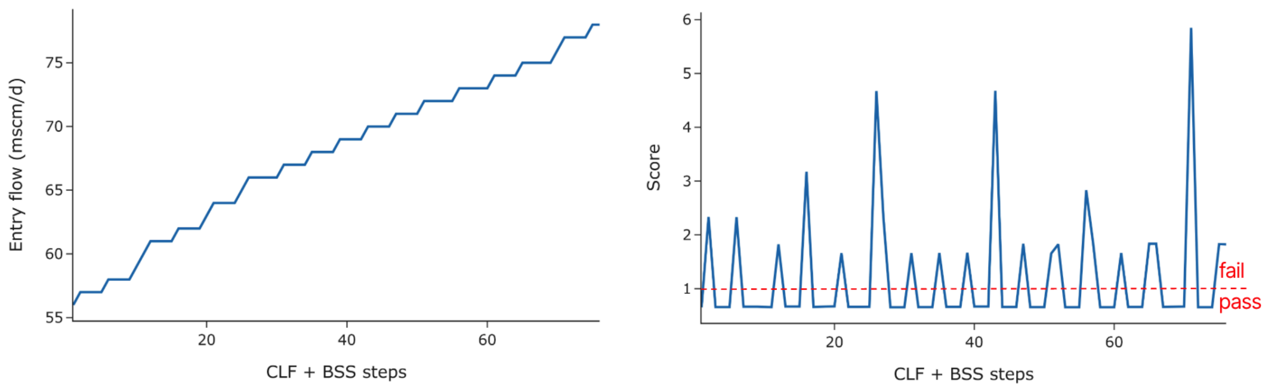

Figure 11: Entry flows (left) and scenario scores (right) for an example automated CLF run investigating the entry capability of the network in RIIO zone 2 for a peak day scenario in 2026.

4.5. Areas for further development

The CLF was developed as a proof of concept tool and has demonstrated its feasibility. To become a fully useful tool, several areas require further development.

Update demand adjuster tool: One crucial step is to update the demand adjuster tool to its latest version, as the tool was in the process of being updated during the build of this component.

Add support for removing flow from the target zone: The tool also needs to be expanded to support removing flow as well as adding it. This is important for exit analysis scenarios where minimum supply assumptions can make even base demand flows unsolvable, and for resilience testing where limited asset availability may prevent base flow scenarios from being solved.

User validation: To ensure the CLF meets its intended purpose, user validation is needed to confirm that all domain-specific technical requirements have been met, and to integrate any that are currently missing. Improving robustness is another priority, which involves better handling of edge cases and varied data input quality. Examples of this include managing situations where the minimum supply assumption scenario cannot be solved, and handling flow adjustments that exceed maximum physical values.

Handle noise in scenario solving: The system needs to address noise in scenario solving, where the BSS might fail to solve a scenario in one run but succeed in another due to random chance. This could be mitigated by introducing retry or extended exploration support into the automated CLF so it can progress after encountering failures.

Support for flexibility in supply flow adjustment: While the current implementation of supply flow adjustment follows a strict order, gas analysts often need to move more dynamically through the adjustment order to push the RIIO zone's capability as far as possible. This is necessary to avoid out-of-zone constraints that are not relevant to the capability of the target RIIO zone.

User testing: Finally, the CLF will require frequent and in-depth user testing to optimise the automated flow analysis strategies and best support the analysts' workflows.

4.6. Summary and Conclusion

Based on the project's results, it is clear that automation of capability flow analysis is feasible. Although there's still a need to improve the tool's reliability and speed through improvements to the BSS, and to reach the higher flow levels achieved by gas analysts, the overall performance of this PoC tool is encouraging. Pushing for these harder use cases, such as analyst-level flows and handling resilience scenarios, will present the next set of challenges.

The individual automation components, particularly the flow adjuster, are already showing that they could save analysts a significant amount of time. Distributing flow, which currently takes minutes, takes only seconds with the flow adjuster tool. This also improves reliability, as the manual supply adjustment process in SIMONE is error-prone and difficult to validate.

The integrated CLF and BSS demonstrates the impact of the BSS when used to power a larger feature. The successful flow increases across all RIIO zones, gas days, and for both entry and exit scenarios shows this tool can automate network analysis at a scale that immediately covers a significant number of time-consuming manual tasks.

Even if the tool cannot match analyst-level performance on the most challenging use-cases, it still represents a major boost to analyst productivity and the overall scale of network analysis that can be carried out. With clear next steps identified for improvement, from updating functionality to better matching analyst workflows to optimising the algorithm itself, the feature is in a promising position and is clear that it has the potential for significant future impact.

5. Derisking the Future

5.1. Local Component Demonstrator

The Local Component Demonstrator (LCD) has been developed as a rapid, agile solution to provide a user interface for direct interaction with the individual underlying components of the wider FastPress tool. This interface operates via software installed on an individual user's laptop, leveraging their local instance of SIMONE. This approach was chosen to navigate the significant initial setup hurdles associated with deploying cloud infrastructure within NESO, which would have created a high barrier to entry for providing such an interactive environment to gas analysts.

It is important to differentiate the LCD from the future cloud deployment of FastPress. The LCD's primary purpose has been to offer an immediate, interactive environment to receive feedback from gas analysts. In the sessions undertaken during this phase, subtle inaccuracies within the complex processes were revealed, and subsequently addressed. Ultimately, this has allowed the development team to thoroughly investigate and plan for the most effective integration of FastPress tools into the existing workflow of analysts.

The developed Start Point Finder (SPF), Limit Violation Checker (LVC) and Flow Adjuster[9] components have been successfully integrated into the LCD. Each component can be configured by the user to process an input scenario of their choosing, producing an interactive output page for exploration of results.

5.1.1. Start Point Finder

Within the LCD, the Start Point Finder exposes the initial step of the wider Base Scenario Solver process to a user, wherein an input scenario is evaluated against the set of known solved scenarios to find a subset of those scenarios deemed most similar.

This is presented as a scrollable list of similar scenarios, with high-level information (scenario name, percentage similarity, number of operational compressors within the scenario) being displayed alongside each list entry. An example of this is shown in Figure 12.

[9] Part of the Capability Limit Finder (CLF)

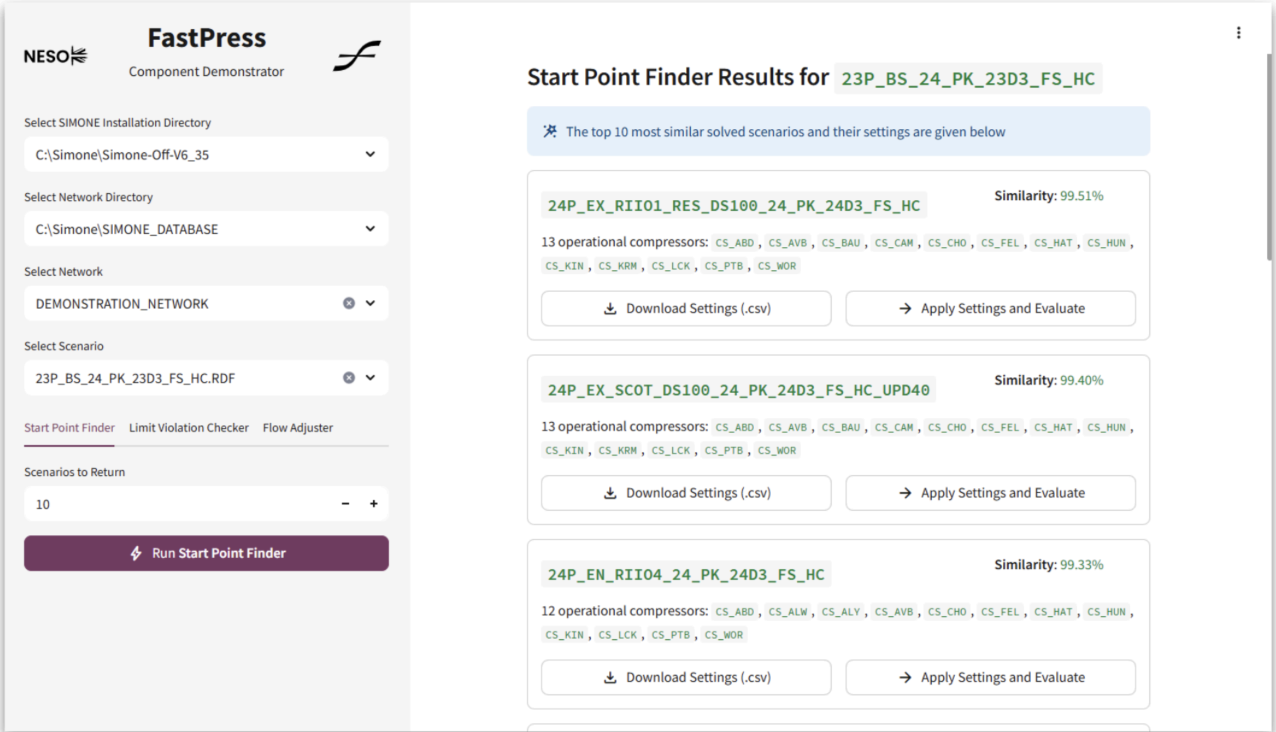

Figure 12. An example of the Start Point Finder results page, showing a list of similar scenarios

Two buttons are also given for each returned scenario. One enables the settings associated with a scenario to be downloaded as a CSV, for manual inspection in Microsoft Excel. The other button invokes a background task to apply the settings from the selected scenario to the scenario selected as input, before displaying a modal dialogue window for the user to download the resulting RDF file. Additionally, the Limit Violation Checker is immediately executed on the modified scenario, with results shown directly within the modal. This is shown in Figure 13.

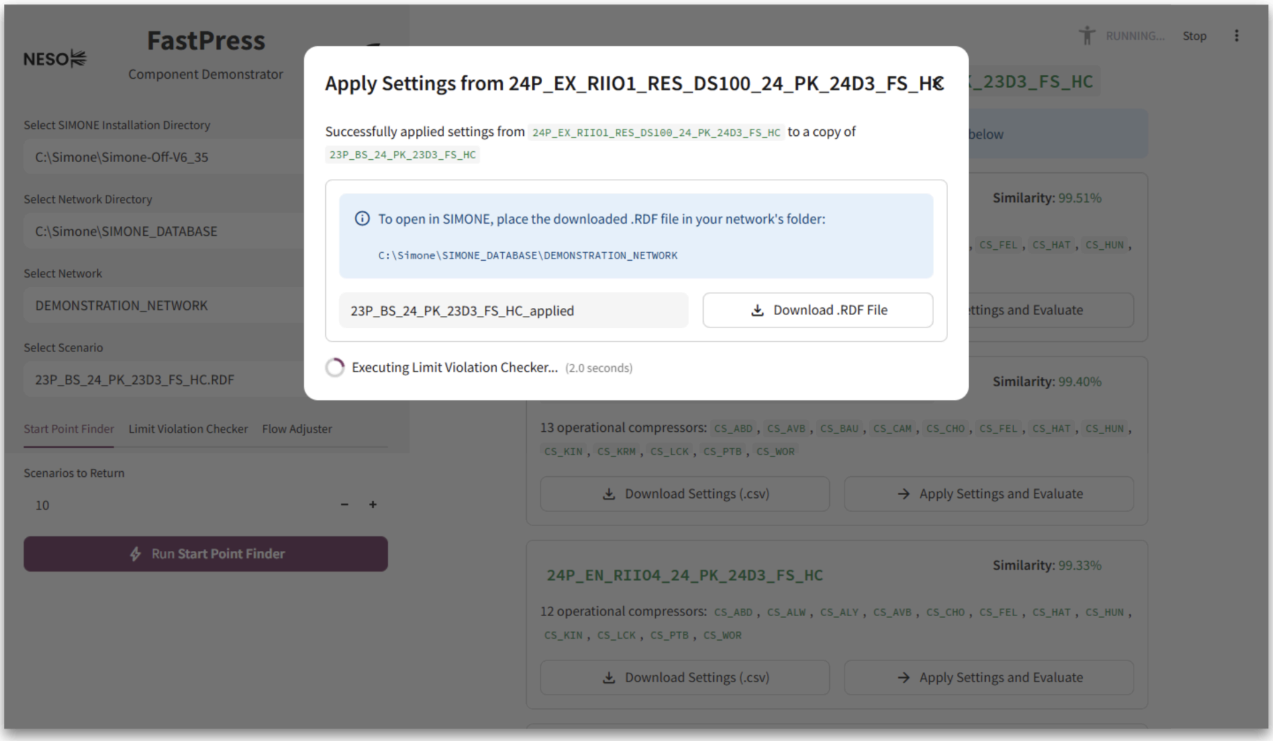

Figure 13. An example of the modal dialogue window, showing settings from a scenario returned by the Start Point Finder being applied to the input scenario, before subsequent evaluation via the Limit Violation Checker

5.1.2. Limit Violation Checker

The Limit Violation Checker (LVC) component within the LCD allows a user to execute the same limit checking and scoring procedure performed as part of the Base Scenario Solver against an input scenario of their choosing. A user simply selects their desired scenario, clicks the "Run Limit Violation Checker" button, and is then presented with results after a short wait (usually less than 10 seconds).

A high-level view of the scenario's pass/fail performance against 6 key factors is presented to the user once processing completes. These factors are:

- **Execution:** whether the scenario has been successfully executed by SIMONE
- **Linepack:** whether there is a mismatch in linepack (target vs. actual), with a violation threshold of 1 mscm/d
- **Supply/demand:** whether there is a mismatch in supply vs. demand, with a violation threshold of 0.01 mscm/d
- **Supply nodes:** whether or not any supply nodes in the network have breached their defined limits (e.g. pressure limits)
- **Offtake nodes:** similar to supply nodes, whether or not any offtake nodes in the network have breached their defined limits

- **Compressor stations:** whether any limits associated with compressor stations have been breached (e.g. upper temperature limits)
- **Compressor units:** whether any limits associated with compressor units have been breached (e.g. a unit has a status SURGE or CHOKE, which indicates it is running outside its operation envelope)
- **Regulators:** whether any limits associated with regulators have been breached (e.g. flow limits)

Any assets that have indeed breached their associated limits are presented in asset-specific tables below the high-level pass/fail overview. These tables allow the user to quickly identify which asset and which limit has been breached, and by how much. An example of this is shown in Figure 14.

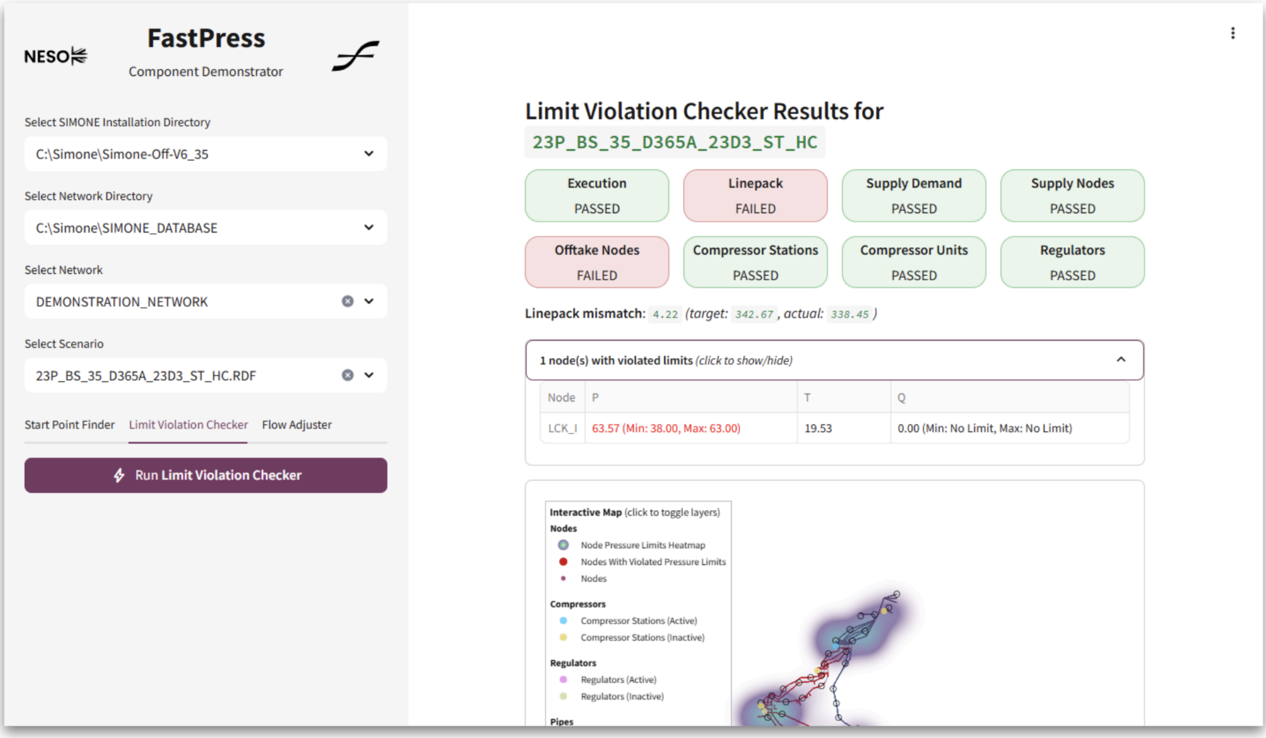

Figure 14. An example of high-level results output from the Limit Violation Checker

Further down the results page is an interactive map view of the network, with assets and asset-specific limit information overlaid. This map can be zoomed and panned in a familiar fashion to modern map-based web applications, with more information viewable when hovering the mouse cursor over a specific network asset.

A variety of data layers are provided, which can be toggled on/off using the control panel in the top-left. These data layers consist of:

- **Node pressure limits heatmap:** a blue/green/yellow heatmap of node pressures against limits, allowing a user to see geographical areas where node assets are pushing (or violating) their pressure limits

- **Nodes with violated pressure limits:** bright red markers highlighting those nodes which have violated their pressure limits
- **Nodes:** all other nodes within the network (without violated pressure limits)
- **Compressor stations with violated limits:** bright red markers highlighting compressor stations that have violated their top-level limits, or if corresponding compressor units have violated limits
- **Compressor stations (active/inactive)**: markers indicating compressor stations which aren't violating limits, with blue/yellow markers signifying compressor stations which are active/inactive respectively
- **Regulators with violated limits:** similar to compressor stations, bright red markers highlighting regulators that have violated their limits
- **Regulators (active/inactive):** again, similar to compressor stations, but with pink/dark yellow markers indicating regulators that are active/inactive respectively
- **Pipe pressure:** blue/red coloured edges within the network corresponding to pipes, coloured depending on the pipe's pressure (blue being low, red being high)

An example of this interactive map with all data layers visible is shown in Figure 15.

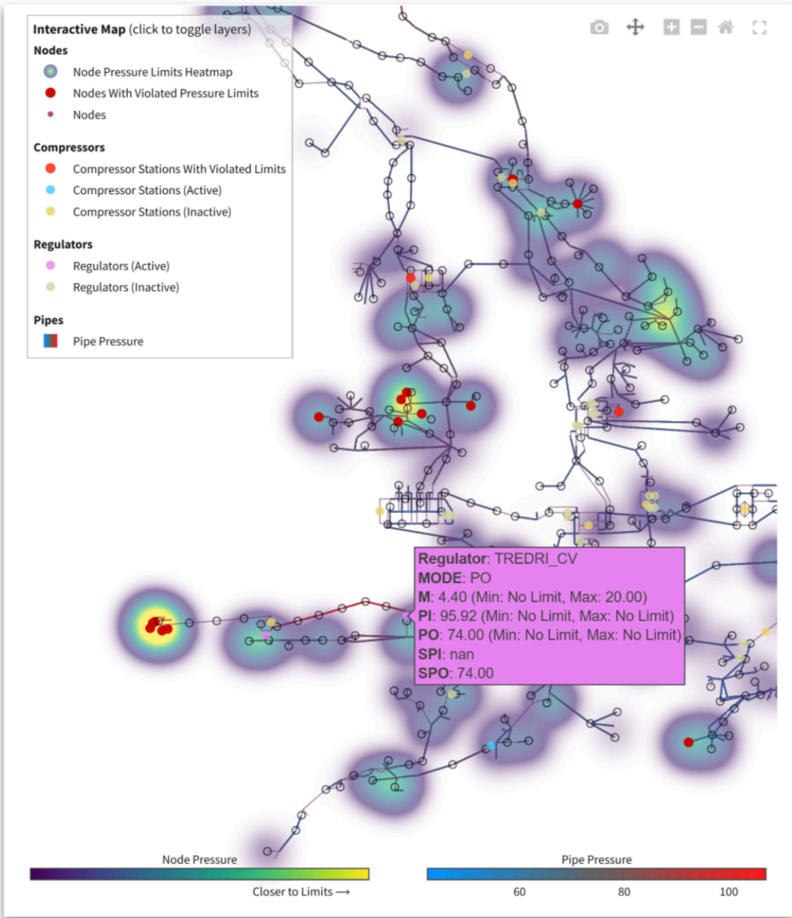

Figure 15. An example of the interactive map within the output of the Limit Violation Checker, with details for an individual regulator shown via hovering the cursor over the asset

5.1.3. Flow Adjuster

The Flow Adjuster component allows a user to experiment with adding a fixed amount of flow to an entry/exit scenario, automatically following the standardised procedure used by gas analysts in practice. When executing - after the user configures the target RIIO zone, the amount of flow to add and the scenario day - the underlying component process will read from an uploaded balance sheet to determine the correct flow change procedure to follow.

Figure 16 shows an example of what is displayed to the user when Flow Adjustment has completed. The user is given a quick summary of the maximum flow achieved, alongside a complete table of all changes to flows at each node in the network. Below this table is another interactive map, similar to that given in the Limit Violation Checker results page. Here, two heatmaps are overlaid, highlighting the nodes/areas which have resulted in the removal of flow (blue), and the addition of flow (red).

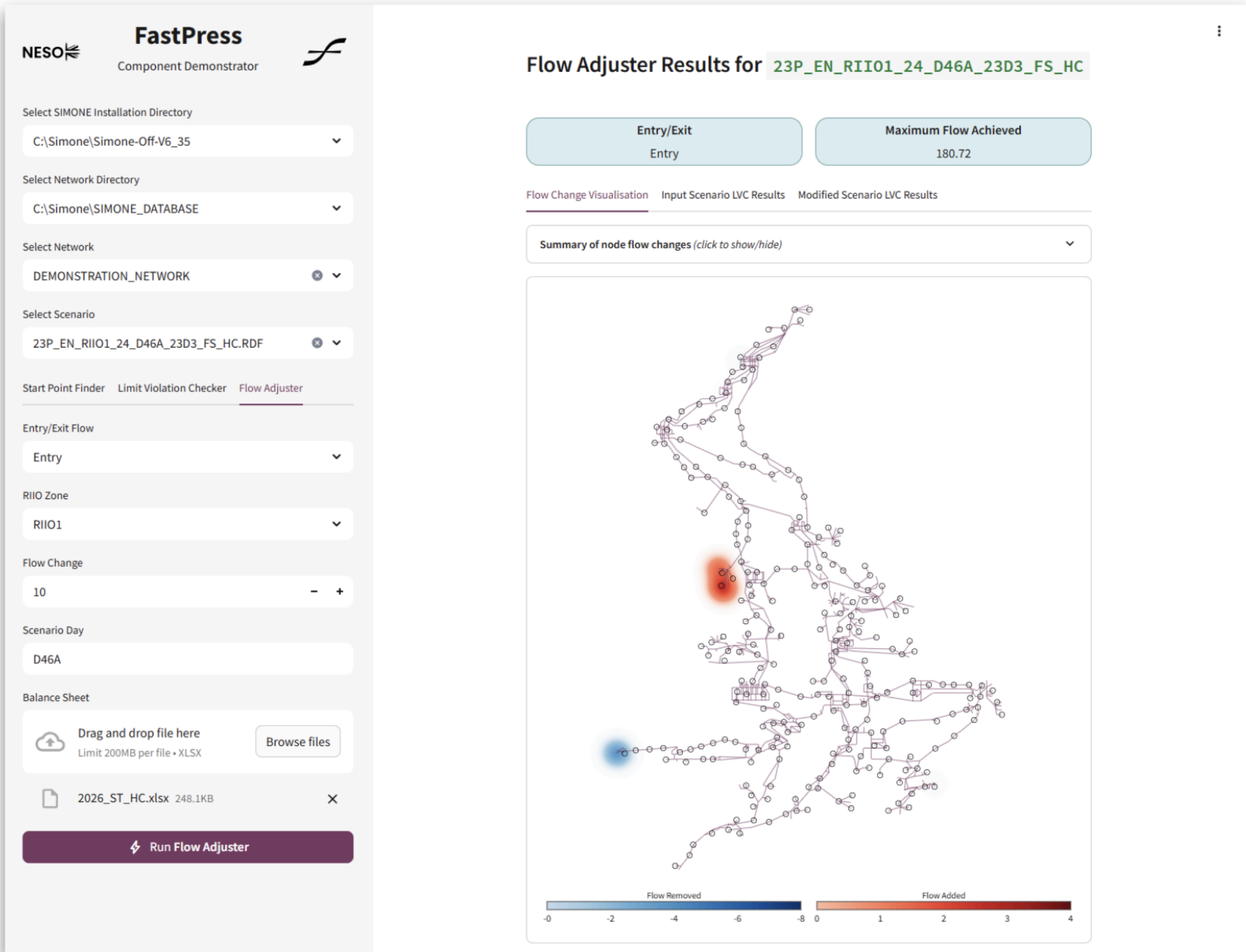

Figure 16. An example of displayed output from an execution of the Flow Adjuster component

For additional utility, both the input and modified scenarios are also automatically evaluated by the Limit Violation Checker, with results given in separate corresponding tabs within the Flow Adjuster results interface.

5.2. UX Design

The eventual end-goal for the deployment of FastPress consists of a bespoke web-based tool, the preliminary design of which was explored in the previous phase of the project. Alongside early visual designs, outputs from the previous phase established that the web application will support the asynchronous execution of processing jobs. Users will navigate a simple interface to configure and submit these jobs, returning later to inspect the results.

Throughout Alpha+, significant progress was made in refining previous early user interface designs (created in Alpha) to better meet the needs of gas analysts.

5.2.1. Progress in Alpha+

The Alpha+ phase commenced with a detailed heuristic evaluation of the existing UI designs, which identified several areas for improvement, including issues with progress visibility, inconsistent terminology, a lack of workflow clarity, and inadequate error recovery mechanisms. Based on these findings and direct feedback from NESO gas analysts, the interface was comprehensively redesigned. The primary objective of this redesign was to create a more intuitive user experience by simplifying workflows and ensuring the system's logic aligned more closely with the analysts' established methods and thought processes.

The redesign was an iterative process, characterised by regular review sessions with gas analysts to continuously refine the user experience. This collaborative approach was particularly beneficial in shaping the submission, tracking, and review of scenarios. As a result, the scenario submission process has been notably simplified, featuring an improved file upload and configuration system, complemented by enhanced validation and guidance to proactively reduce user error. Furthermore, the presentation of results has been improved to facilitate quicker and more accurate interpretation of outputs. This includes the introduction of clearer data tables and new visual indicators to highlight key information at a glance.

To support decision-making and enhance the explainability of the outputs, initial explorations into more visual and summary-level indicators were undertaken, including preliminary work to improve the utility of heatmaps. The design and refinement efforts were specifically focused on the user experience for inspection of Base Scenario Solver (BSS) and Capability Limit Finder (CLF, i.e. entry/exit flow analysis) results.

To facilitate asynchronous review and gather broader stakeholder feedback, video walkthroughs and structured prototype flows were produced and shared throughout.

5.2.2. UI Page Designs

Realistic mockups for five distinct pages have been created during this phase. The first of which is intended to act as the "home" page of the web application, and consists of a single searchable, filterable table of all experiment jobs previously submitted to the FastPress tool. Within each table row, the user can see high-level information about experiments:

- The experiment name (a custom name describing the submission)
- The type of the experiment (e.g. Base Scenario Solver, or Entry Flow Analysis)
- The date/time at which it was submitted
- The user that submitted the experiment
- The status of the experiment (i.e. whether or not it has completed)

The final design of the home page is shown in Figure 17.

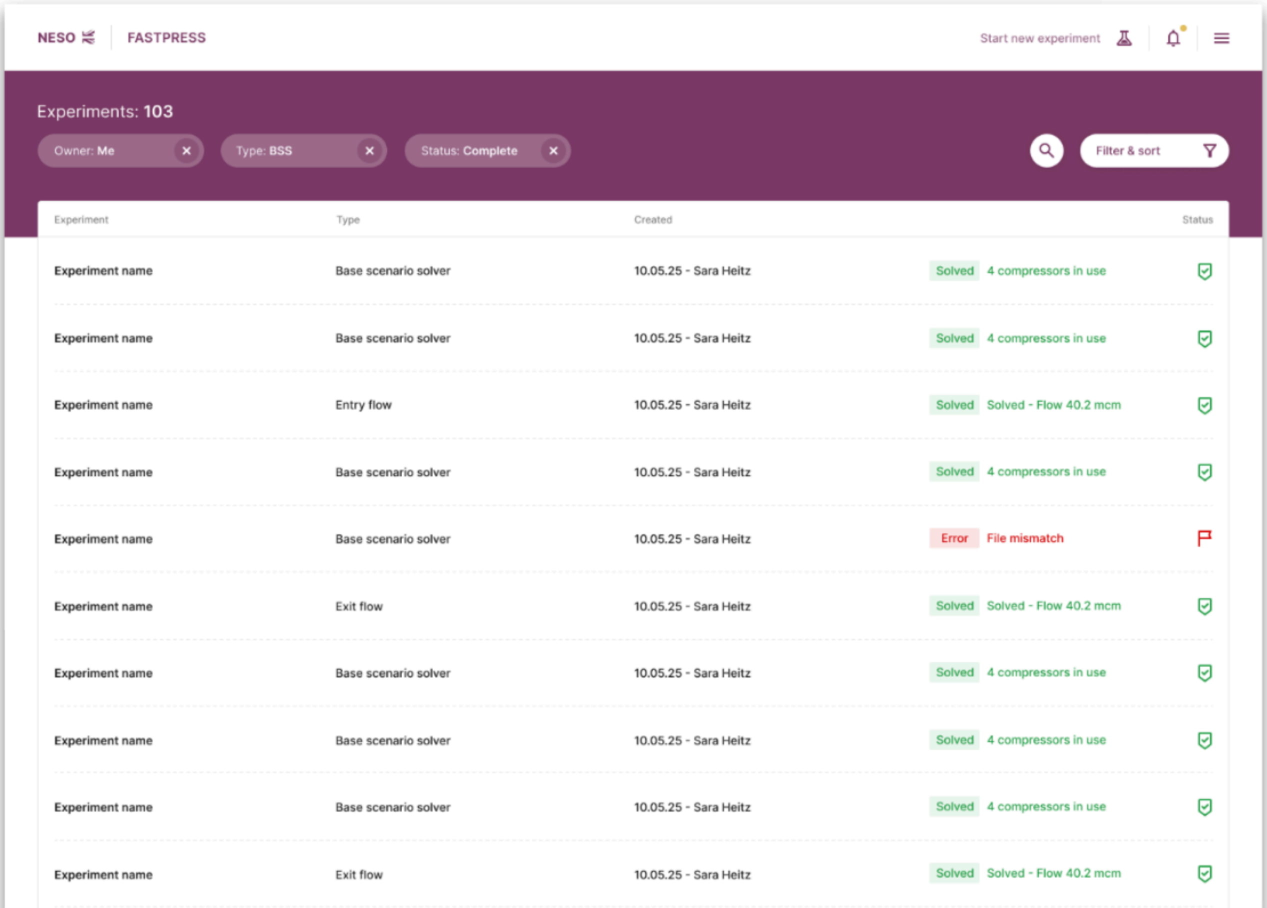

Figure 17. The UI's "home" page - a searchable, filterable table of previously submitted experiments

Supplementing this table is a separate page for applying advanced filtering to historic submissions. This was identified as a crucial element of a gas analyst's workflow - it is common for analysts to work with an overwhelming number of distinct analysis projects, so being able to quickly find submissions from the past is essential. The design of this page is given in Figure 18, and shows how a user can refine the visible table rows depending on multiple criteria.

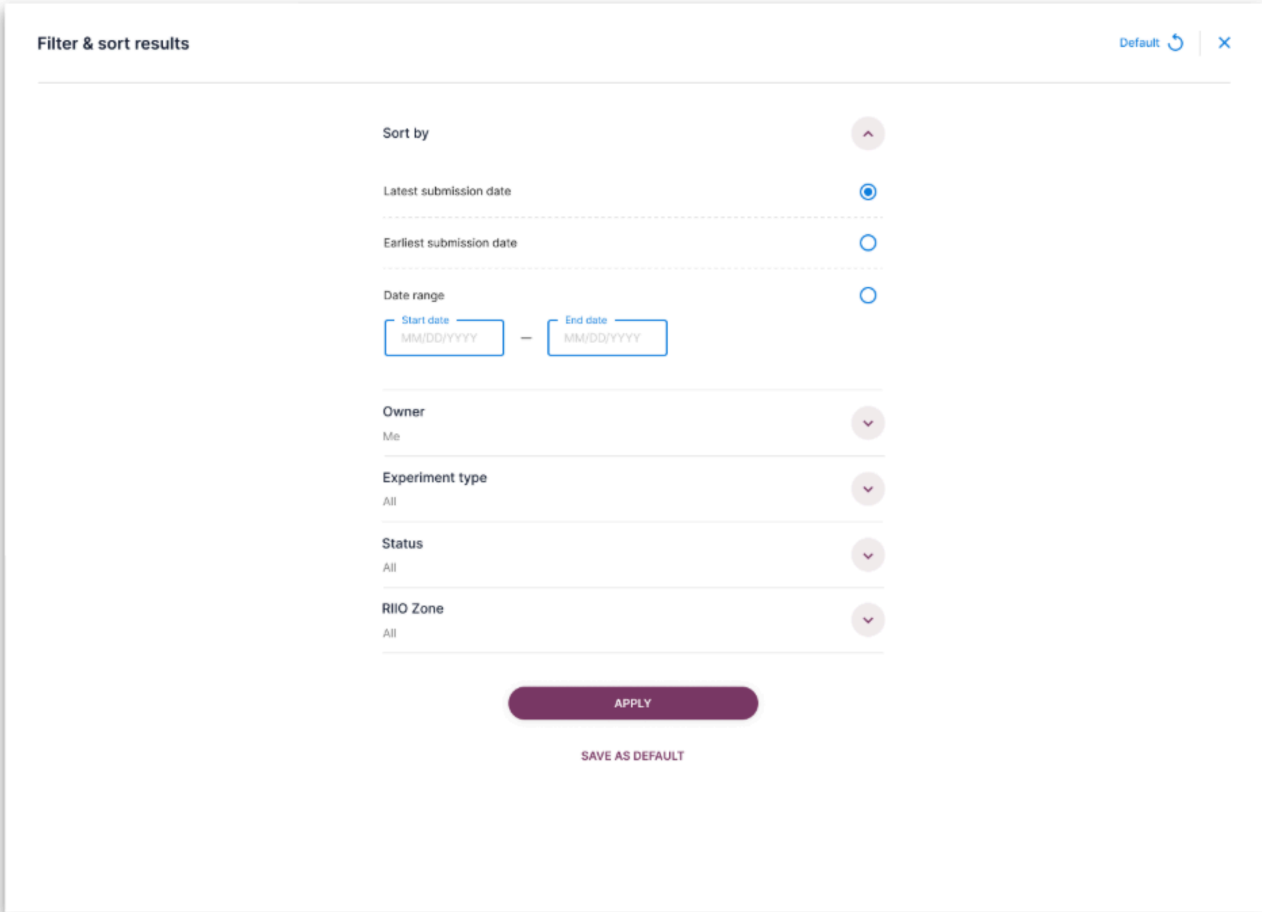

Figure 18. The advanced table controls page, used to filter the experiment table to find specific entries

The "new experiment configuration" page - shown in Figure 19 - has been designed to streamline the potentially complex process of submitting new processing jobs to the FastPress tool. Many nuances of the existing gas analyst workflow have been addressed here, by following a familiar process of scenario data input to that present in existing tools (i.e. SIMONE). Alongside the configuration of various submission metadata (e.g. a custom name and the type of submission), a user simply needs to drag their network folder into the web page, before selecting the desired scenario file(s) contained within.

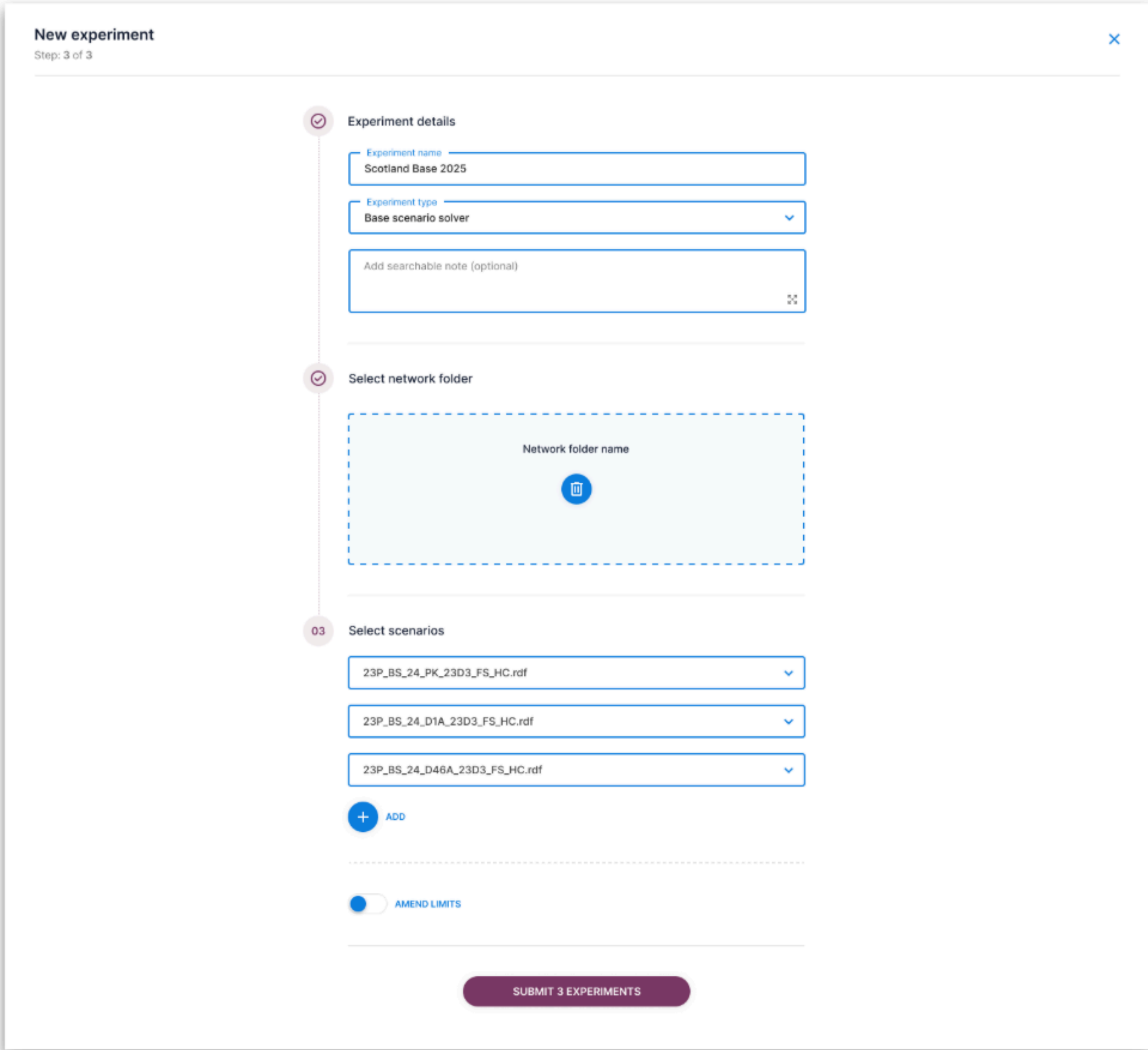

Figure 19. The new experiment configuration page

Upon completion of a submitted experiment, a user would be able to view results in an experiment-specific "results" page. For Base Scenario Solver experiments, the general design of this page is given in Figure 20. Here, a user can see an overview of the underlying algorithm's outputs, including the final solved scenario (if present). Feedback sessions with analysts have identified the highest-priority data to show here, including the number of compressors utilised by the solution, and a succinct table of how assets are performing against their limits (e.g. any violations, or any assets operating close to their limits).

Filling the majority of the screen space is an interactive map of the network, with geographical visualisations of asset performance overlaid, for the selected scenario. The main thing shown here is a heatmap revealing the location of assets operating close to, or beyond their limits. In addition, users would be able to hover/click on components of the network to reveal asset-specific information.

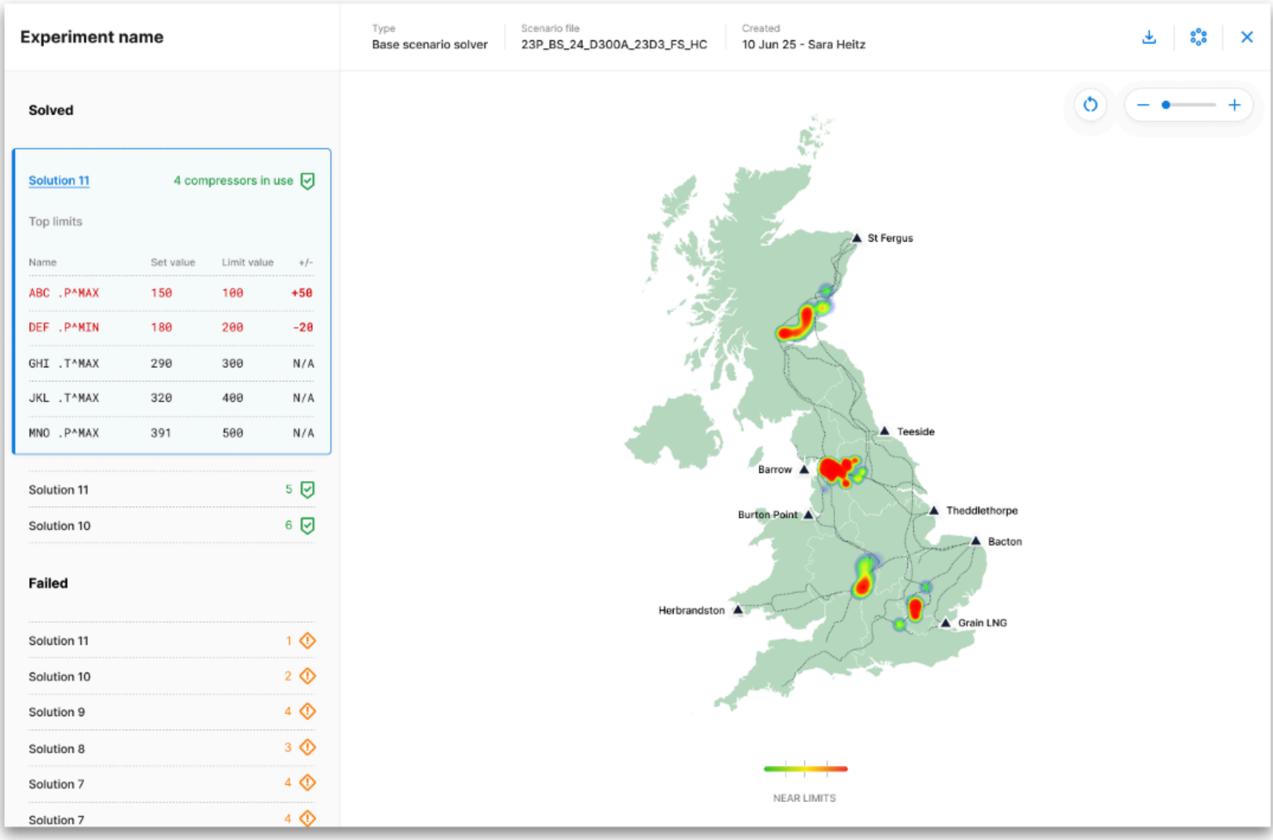

Figure 20. The "results" page for a Base Scenario Solver experiment

For Entry/Exit Flow Analysis experiments, the design of the results page is similar but includes specific information relevant to flow inside/outside the RIIO zone associated with the experiment in question, as shown in Figure 21. One key addition here is the "flow slider" element at the bottom of the page. This is designed to allow the user to intuitively move through the steps of the underlying Capability Limit Finder algorithm, as flow changes.

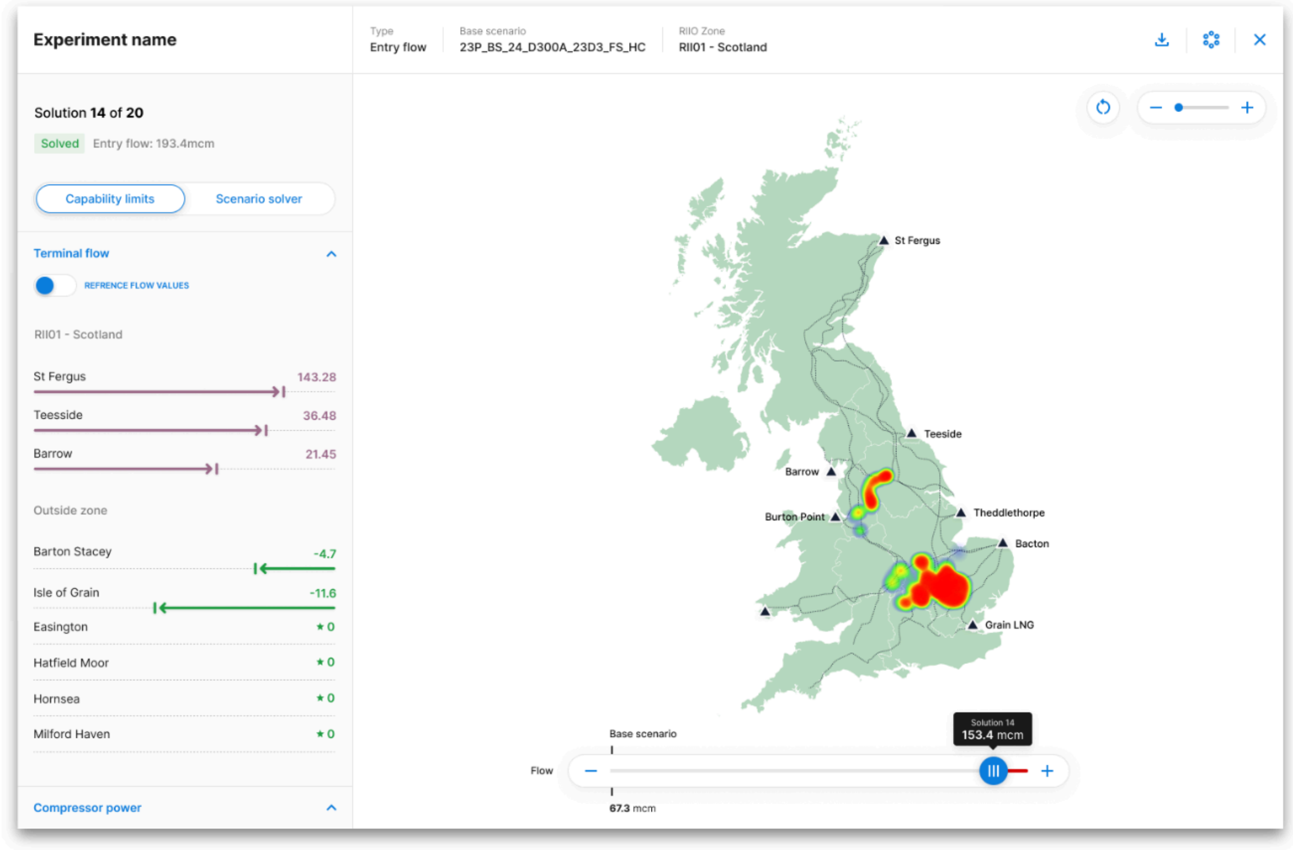

Figure 21. The "results" page for an Entry Flow experiment

Figure 22 shows an example of how asset-specific information could be visualised within the user interface through the interactive map. Here, an individual compressor station has been selected by the user, revealing visual overviews of the associated multi-junction and compressor envelope, alongside a table of all other information relating to the asset.

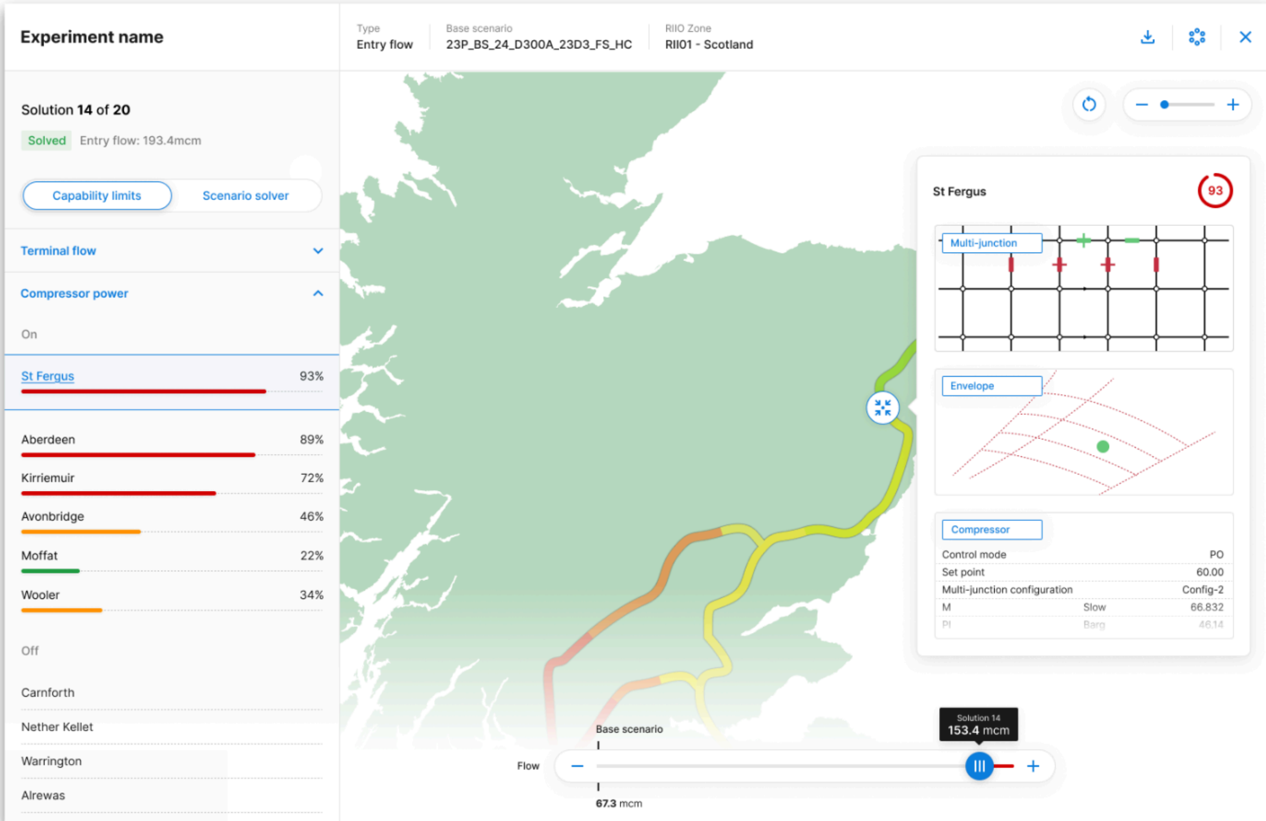

Figure 22. Conceptual designs for displaying individual asset information

5.3. Representative Cloud Deployment

5.3.1. Target State cloud application

The eventual end-goal of the project is to deploy the FastPress tool to NESO's Azure cloud instance. During the Alpha+ phase, a high-level solution architecture has been developed, which details the different Azure components and their interactions, shown in Figure 23.

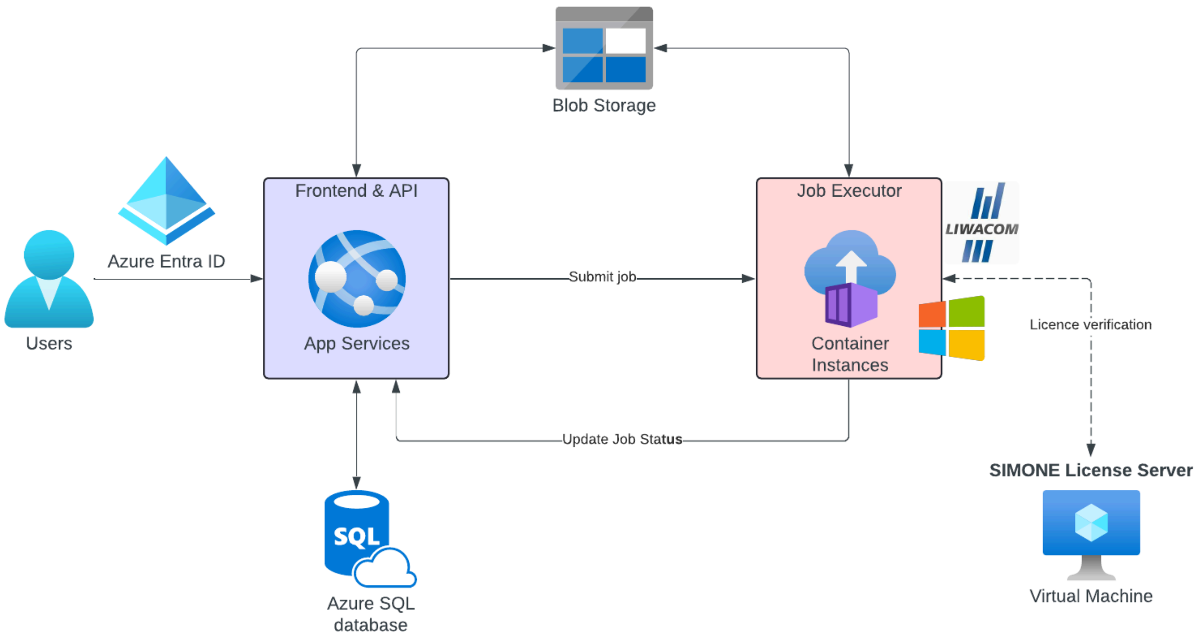

Figure 23. High level overview of intended cloud architecture

The overall design of the FastPress cloud application uses a queue-based asynchronous processing architecture. Users of the FastPress application would access the tool's user interface via a web browser, authenticating using Azure Entra ID with their NESO credentials. The frontend application and Application Programming Interface (API) would be deployed using Azure App Services.

When analysts submit jobs through the front-end, they would be added to a queue and processed sequentially by the Job Executor. This component uses Azure Container Instances running a Windows container with the SIMONE application. While Linux containers are typically preferred in cloud computing, the version of SIMONE used in NESO necessitates the use of Windows Containers. This requirement limits cloud compute options, with Azure Container Instances being the most suitable solution identified.

For job execution, the Job Executor must communicate with the SIMONE license server to validate license keys. The SIMONE license server is currently in the National Grid Azure instance, with a planned migration to the NESO Azure instance later this year. If this migration is delayed, existing patterns may need to be used to "reach back" to the National Grid cloud for SIMONE license verification.

Azure Container Instances offer horizontal scalability for parallel job processing, with the number of SIMONE license keys dictating the maximum number of parallel instances. Scalability will be configured to match the available SIMONE license keys. To optimise job processing efficiency and reduce completion time, the number of licenses used per job can be adjusted.

A robust and scalable data architecture is proposed, using Azure services for efficient metadata management and data processing. Specifically, an Azure SQL database will serve as the central repository for storing metadata associated with the management and tracking of analyst's jobs. This metadata will include various details such as job definitions, execution parameters, status updates, and historical records, which can be viewed by analysts via the user interface.

Data associated with individual job submissions will be sent to Azure Blob Storage as it can handle large amounts of unstructured data with high availability and durability. The Job Executor will actively monitor Azure Blob Storage. When a new job is detected, the Job Executor will retrieve this data for processing.

5.3.2. Representative deployment approach

During the Alpha+ phase, we were provided access to a Sandbox Landing Zone on the NESO cloud instance to experiment with the FastPress cloud deployment to derisk future work for the full cloud deployment described in the previous section.

The FastPress application would use a conventional queue-based asynchronous processing architecture. The main challenge to this deployment was the feasibility of installing and running SIMONE on Azure compute, given that running Windows containers in a cloud environment is a non-standard approach.

We were able to successfully install SIMONE on a Windows Azure Container instance during this phase, which mitigates the main risk for a full FastPress cloud deployment in the future. This installation is achieved via Azure DevOps Pipelines which programmatically install SIMONE onto a Windows Docker container, which are pushed to an Azure Container Registry in our Sandbox Landing Zone and subsequently deployed to Azure Container Instances. A separate code repository for this work is available in the NationalEnergySO Azure DevOps project.

We opted not to attempt to run the FastPress application in the cloud during this phase due to its reliance on the SIMONE license server. This licence server is currently located on the National Grid Azure cloud instance, not the NESO cloud with the Sandbox Landing Zone. Connecting from the NESO cloud to the SIMONE license server on the National Grid Cloud would be a complex undertaking with limited benefit, given the SIMONE license server is slated for migration to the NESO cloud later this year.

6. Future Work - Beta Phase

6.1. Background

A proposed Beta phase is explored in further detail in the remainder of this section, articulating the goals and initial delivery plans, while also highlighting the associated benefits, dependencies and risks. The Beta phase, following on from Alpha+, will focus on continued exploration and innovation, building upon existing work and incorporating additional features identified with NESO users. This phase is specifically divided into three development areas:

1. Real World Benefits
This involves transitioning developed features onto the current network model and into production. Our team will develop these preliminary features into a robust, fully embedded, and user-friendly tool with a seamless front-end for gas analysts. Significant engineering work will also be undertaken to embed the tool into NESO's cloud environment, helping to leverage efficiencies from scaling computation and centralise data.

2. High Impact Innovation
Building on the current BSS, which only solves static scenarios, a tool will be developed to address transient scenarios, which form the majority of analysts' work. Additionally, an 'Infrastructure Timeline Optimisation' feature will be created to identify network assets for repurposing or retirement, expanding on the pipe removal recommender initially developed to a PoC level in the previous Alpha phase.

3. Product Improvements and Formalisation
This area focuses on improving the efficiency of existing and new features, such as the CLF and future features developed . An 'Auditing Tool' will also be developed to streamline the auditing process for solved scenarios by wrapping around current features.
This phase is further detailed in the remainder of this section, outlining its goals and initial delivery plans, as well as associated benefits, dependencies, and risks.

6.2. Deployment Roadmap

6.2.1. Features

Transient Scenario Solver: Building upon the BSS, this tool would solve transient scenarios
Outputs

- An algorithmic approach to transient scenario solving that minimises the need for analyst involvement and automates existing manual processes
- It will be incorporated into the flow analysis workflow, automating a crucial part of the process.

Impact

- The automation provided by this tool will significantly reduce the workload of repetitive tasks for gas analysts, allowing them to focus on more complex and high impact problems.
- It has the potential to uncover new solution methods for gas network planning.

Infrastructure Timeline Optimisation: The tool is designed to identify gas network assets, such as pipelines and compressors, that can be repurposed or decommissioned, along with the optimal timing for doing so.

Outputs

- It will identify assets that can be repurposed, measure the impact of removal and find the optimal configuration and timing of assets for removal.
- It will offer a timeline or a sequential plan for removing assets.

Impact

- It will enable enhanced asset reuse.
- Potential to help defer or avoid costly capital expenditure (capex) on new infrastructure development.
- Potential to minimise the repair and maintenance costs of existing infrastructure (by identifying assets for decommissioning), thereby reducing operational expenditure (opex).
- It will free up time for maintenance staff.
- Pipes can be identified for repurposing of line pack storage.

Network Transition: Migrate FastPress tool and features to the common network model

Outputs

- Network Migration: Systematically adapt FastPress tools, data, configurations, and dependencies to the current network.
- Process Documentation: Create comprehensive documentation including guidelines, specifications, and troubleshooting.

Impact

- Enhanced Accessibility: Gas analysts gain immediate access and use to FastPress tools
- Preservation of Feature Performance: Maintains FastPress feature performance and functionality across network changes.

SPF Model Update: This update enhances the SPF model by enabling the incorporation of newly solved scenarios.

Outputs

- An SPF tool that can be continuously updated with newly solved scenarios, both before and after network transition

Impact

- Accelerates the solution process and enhances solution quality by improving solver performance (both static and transient).
- Elevates solution quality when the SPF is used independently.
- Maintains FP feature performance despite network changes.

Auditing Tool: Auto-generation of the scenario audit sheet

Outputs

- Wraps around existing features to streamline auditing.
- Auto-generates audit sheet content.
- Ensuring that key components/aspects meet the required criteria (pass / fail).
- It provides an "inspection" user interface for senior analysts, leveraging the Limit Violation Checker (LVC) and the Solution Scoring Tool (SST).

Impact

- Automates auditing to reduce errors and frees up senior analyst time.
- It improves the interpretability of results by using FastPress tools.

6.2.2. Plan

A high-level implementation plan for Beta is outlined below, showing the three stages and the features to be developed within each. Please note that the sequencing of feature development may change, as the Beta start date is not yet confirmed and delivery is dependent on Faculty's internal resourcing.

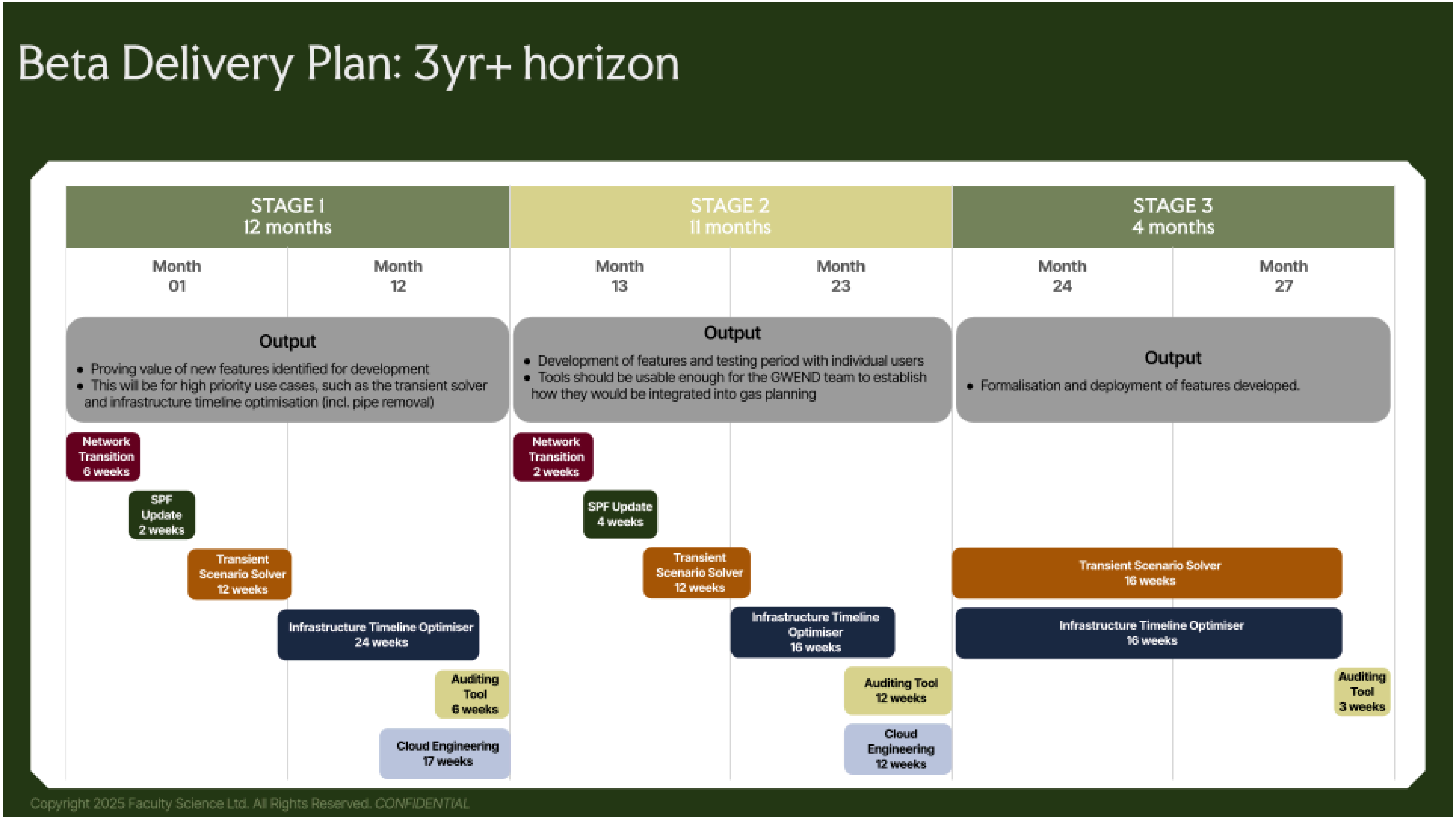

Figure 24. Beta feature development timeline

6.3. Beta Benefits

6.3.1. Beta Benefits and how they Translate to the End Consumer

Internally the tool provides direct benefits to NESO by helping to streamline the workflow for gas analysts and significantly reducing the amount of manual effort required. It automates time-consuming tasks like setting up and solving scenarios, checking for violations, and iterating flows, which allows a greater number of scenarios to be solved much faster. This frees up valuable analyst time to focus on more complex and high-impact problems that wouldn't be possible under current operational constraints. The project's time-saving benefits, along with its ability to help avoid infrastructure and constraint costs and facilitate asset repurposing, are a key focus of the Beta phase.

However, the FastPress Beta phase is also centered on realising tangible benefits that directly (and/or indirectly) impact the ultimate consumer. While the project’s technical advancements are built to be for internal use, their primary purpose is to enhance National Gas Transmission's (NGT) ability to operate a more efficient, resilient, and cost-effective gas network. These improvements are crucial because, in a regulated market, operational efficiencies and cost reductions for the network operator translate directly into lower bills for consumers.

The FastPress tool allows NESO to better plan for future demand and supply changes, by rapidly analysing a greater number of complex scenarios and identifying potential network constraints before they become costly issues. This proactive approach helps to avoid the significant expenses associated with reactive interventions, such as those required to manage real-time network imbalances. These reactive costs are ultimately paid for by the consumer through network charges. By providing NGT with the data and speed to avoid these costs, FastPress directly benefits households and businesses across the country.

Furthermore, the benefits extend to long-term infrastructure planning. By allowing NGT to optimise its long-term investment strategy, the FastPress tool helps to ensure that capital is deployed only where it is truly needed. It can help to identify opportunities to repurpose existing assets, such as pipelines, for new energy vectors like hydrogen, which will be essential for the UK's energy transition. This ability to avoid or defer new, costly infrastructure projects means that NGT can operate with a leaner, more effective network, reducing the need for significant capital expenditure that would otherwise be passed on to consumers.

In summary, the beta benefits of the FastPress tool are designed to deliver real, quantifiable value to the consumer. We have identified three primary benefit channels that will contribute to this goal:

1. ***Time Savings:*** Freeing up valuable analyst time to focus on strategic and more complex scenarios.
2. ***Avoided Constraint and Infrastructure Costs:*** Optimising network operations and long-term planning to prevent costly reactive and capital spending.

3. ***Repurposing of Existing Assets:*** Enabling a more sustainable and cost-effective network by identifying and repurposing infrastructure for the future.

6.3.2. Benefit Channels Identified

Time Savings

The FastPress tool will provide gas analysts with enhanced resource capacity by significantly reducing the time required to solve scenarios.

The existing process is time-consuming and manual, often taking months to complete an entire report. The tool automates many of these steps, such as setting up scenarios, checking for violations, solving scenarios, and iterating flows, allowing a greater number of scenarios to be solved in a significantly faster time period. This frees up valuable analyst time, which can then be redirected toward more strategic and complex scenarios that would not otherwise be possible under current operational constraints. Without this tool, the desire to increase the number of scenarios run would require a much greater time and staff investment.

Avoided Constraint and Infrastructure Costs

By enabling the analysis of a greater number of scenario demand days, the project provides more comprehensive data points and increases the accuracy of regulatory reporting.

In the interim, this will lead to better-informed decisions for NGT, allowing them to proactively address potential network issues and, consequently, decrease reactive constraint costs.

In the long term, this improved foresight will allow NGT to make more efficient planning decisions regarding infrastructure investment. The tool will enable NGT to run the network more tightly, helping to avoid unnecessary or poor capital expenditure on new infrastructure projects, while also saving on their associated carbon emissions generated.

Repurposing of Existing Assets

A key feature of the Beta phase is the development of the Infrastructure Timeline Optimisation (ITO) feature, which identifies pipelines, compressors, and other assets that can be repurposed or decommissioned. This enables NGT to operate a leaner network with fewer operational assets.

The benefits here are multi-faceted. It reduces the need for new infrastructure development, which avoids associated costs and carbon emissions. It also minimises the repair and maintenance expenses for existing infrastructure, freeing up maintenance staff and potentially reducing network outages. This channel directly supports a more sustainable and cost-effective network.

6.3.3. How we will Achieve these Benefits

The benefits outlined in the section above, ultimately leading to a reduction in cost to consumers, will be achieved through productionising and deploying the features developed in the Alpha+ phase, and developing and deploying new features in Beta. The below details how the benefits would be realised, with a visualisation of represented at the end of the section.

Time Savings

The primary way we achieve time savings is by automating the manual, repetitive tasks currently performed by gas analysts. The process is broken down as follows:

- *Automated Scenario Solving*: The BSS uses a genetic algorithm and the suite of associate features developed to date to find optimal settings and solve scenarios without manual intervention.
- *Expanding to Transient Solutions*: Beyond static scenarios, the Transient Solver developed in the Beta phase will automate the solving of transient scenarios.
- *Flow iteration*: The CLF tool will automate and assist in finding the entry and exit capability limits of the network.
- *Batch Solving*: Using the BSS, analysts can submit multiple scenarios to be solved in a single batch. The tool then works in the background, freeing up the analyst's time for other tasks. This allows for the efficient solving of scenarios overnight or on weekends, leveraging SIMONE licenses outside of normal working hours.
- *Streamlined Auditing*: A new Auditing Tool will automatically generate audit sheets and check against codified limits, which will save senior analysts time in reviewing and validating solved scenarios. The tool's output will provide clear indicators:
  - Pass/Fail: For quick flagging of scenarios.
  - Maybe: To prompt analysts for feedback, contributing to the tool's continuous learning and improvement.

Achieve Avoided Constraint and Infrastructure Costs

The features developed in Alpha+ and the Transient Solver developed in Beta will contribute to savings in constraint and infrastructure costs, through improving the quality and volume of data used for decision-making.

- Increased Data Volume: The efficiency gains from automated solving will allow for a significant increase in the number of scenario demand days that can be analysed. This means moving from a limited number of static/transient scenarios solved to a much more comprehensive dataset.
- Enhanced Regulatory Reporting: By having a richer dataset, NGT's regulatory reports will be more accurate and robust. This better informs Ofgem and leads to more precise system requirements.
- Improved Decision-Making: With more accurate data and a deeper understanding of the network's behavior, NGT can make better planning decisions, both in the short term and

the long term. This proactive approach helps to avoid reactive constraint costs that would otherwise be incurred to manage network imbalances.

- Optimised Investment: In the long term, this improved foresight allows NGT to run the network "tighter." This means avoiding unnecessary capital expenditure on new infrastructure by making more efficient long-term investment decisions.

Repurposing of Existing Assets

This benefit is achieved through the development of the new Infrastructure Timeline Optimisation feature, a key focus for the Beta phase.

- Asset Identification: The tool will analyse the network to identify assets, such as pipelines and compressors, that are redundant or could be repurposed. This provides a data-driven basis for strategic asset management.
- Timeline and Plan: The tool will offer a sequential plan for removing or repurposing these assets. This provides NGT with a clear roadmap for operating a leaner network.
- Multi-faceted Savings: By enabling asset repurposing, the project delivers multiple benefits:
    - Reduced Capital Expenditure: It decreases the need for new infrastructure development.
    - Minimised Operational Costs: It reduces ongoing repair and maintenance costs for decommissioned assets.
    - Environmental Benefits: It minimises the embodied carbon and GHG emissions associated with building new infrastructure.

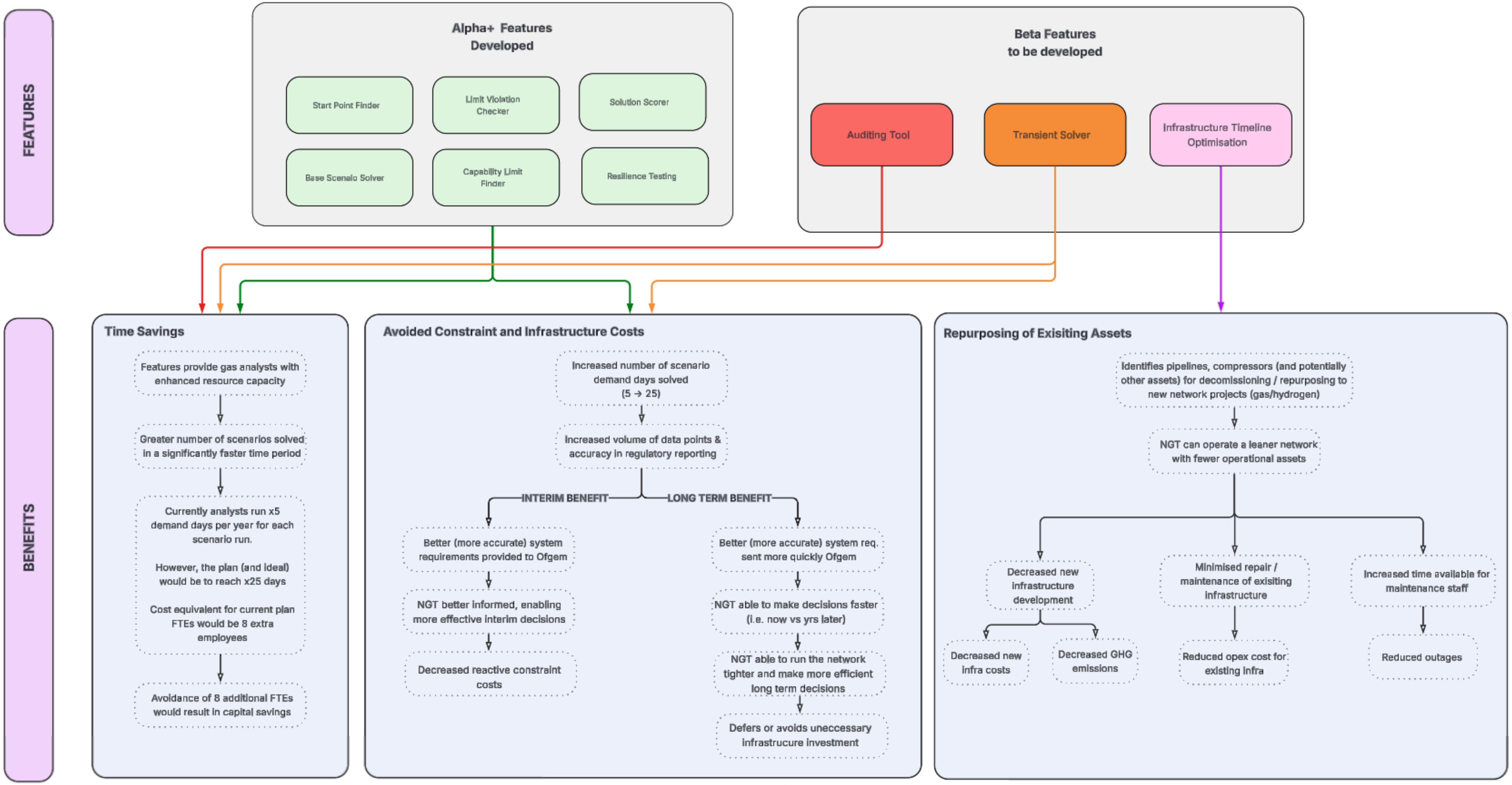

Figure 25. Workflow detailing the benefits derived from features developed in Alpha + and Beta phase

6.4. Dependencies and Risks

6.4.1. Dependencies

	Dependency	Description	Stakeholders
	Before Project Kick Off		
1	Access to NESO's systems	Each developer will require access to: • An active directory account • A VDI • A laptop	• Lauren Cooper (Innovation Business Partner) • Sam Williams (PMO contractor)
2	Single point of contact from NESO cloud/ infrastructure team	There are many stakeholders who Faculty need to engage with 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.	• Simon Wright (Senior Platform Manager - Cloud) • Sam Williams (PMO contractor)
3	Developer access to a Windows Machine	In past phases of the project, permissions within VDI and laptop were very restrictive. We had issues installing software required to complete the project and had to spend 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	• Sam Williams (PMO contractor)

		• Terraform • PNPM • Docker Some of which require administrative access to the machines we use.	
4	Access to developer laptops	While development occurs outside of the cloud, as it will for a majority of the proposed beta phase, testing and development speed will largely be limited by the hardware developers have access to. In our experience both VDIs and non-developer laptops are significantly slower at running SIMONE and the tools we have developed so far. Therefore access to developer laptops is necessary to develop at a reasonable pace	• Sam Williams (PMO contractor)
5	Approval for SIMONE License usage in FastPress Tool	In the past, we have received approval to utilise SIMONE Licences for the development of the FastPress tool. As the existing licences are reserved for business-as-usual activities for the Gas Network Planning Analyst, any 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.	• Aneta Ivanova (SIMONE License Manager) • James Carlin (Lead Architect) • Sam Williams (PMO contractor)

6	Access to SIMONE licenses with the SIMONE-API extension	The Infrastructure Timeline Optimisation tool will require the SIMONE-API extension feature package to work properly (by editing the network), and obtaining this requires buying or renting a new licence.	• Aneta Ivanova (SIMONE License Manager) • Sam Williams (PMO contractor)
	During Project (Week 1 - 4)		
7	Cloud subscription, network integrations and permissions	The required Azure set up and permissions have been listed in the following section Fastpress Project: Alpha Repo... We anticipate the provisioning of resources and permissions may require multiple iterations of testing, and so have reserved 4 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) • Sam Williams (PMO contractor)
	During Project (Week 5 onwards)		
8	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.	• Luke Cutler (Gas Network Planning Manager) • Ben Sloman (Data Science Manger) • Sam Spinnael (Gas Analyst + Project Manager)
	During Project (Week 20 onwards)		
9	Access to users for User Acceptance Testing	To ensure the deployed tool meets the User and Non-Functional	• Gas Networking Planning Analysts

		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 20 of the project. Exact plans will be formulated during the course of Beta.	

Table 4. Beta phase project dependancies

6.4.2. Risks

	Risk Description	I	P	S	Mitigation
1	Access to Software and Hardware - Not getting access to NESO's systems (hardware, software and cloud) with adequate permissions to complete product build. In past phases of the project, obtaining the necessary account/software access took approximately 4-5wks. 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. Additionally, obtaining developer access to a laptop will be beneficial for the speed of development. Some examples of installation we require are; Azure CLI, Terraform, PNPM, Docker and SIMONE Installer. Some of which require administrative access to the machines we use.	5	5	25	We anticipate the provisioning of hardware and software installations may require multiple iterations of testing, and so have reserved 4-5 weeks in the initial phase of the project for teams in NESO and Faculty to work together. There are many stakeholders who Faculty interface with in order to resolve all access issues. 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. Current Alpha+ access and permissions not to be terminated and the end of the phase and continued into Beta, mitigating the need to re-apply for access/permissions.

2	Docker Desktop is required for local development of containerised cloud compute, however in the Alpha+ phase this has not been possible to install on developer laptops due to NESO IT security policies. This is a key blocker for future development of cloud infrastructure if unable to be installed.	5	5	25	Work with NESO IT developer teams to establish a route to installation. A representative from the NESO IT team is essential to ensure this is handled within an appropriate time manner.
3	New features may be computationally complex and expensive which have not been accounted for in the initial iteration of cloud architecture design. This may cause the need to re-architecture to ensure scalability and potential engineering rework.	4	4	16	To mitigate the risk of features being computationally complex and expensive, we have planned for the engineering workstreams to include re-architecture and prototyping at the initial phases of the project. This will allow us to test variations of the architecture by making informed assumptions to calculate potential cost implications and scalability concerns. The assumptions made for each architecture design will be documented and tested later on in the project.
4	Poor engagement during initial user acceptance testing	3	3	9	Engage users early to ensure the tested features match their needs Ensure to make detailed plans and communicate expectations to those involved during the User Acceptance Testing phase of the project.
5	Development Landing Zone: Unclear direction as to how FastPress becomes eligible for a Development Landing Zone within the corporate NESO cloud - a requirement to develop functional early deployments of the FastPress tool.	5	5	25	Clear direction needs to be provided by a dedicated representative from the NESO cloud team

	Currently we have been provided an isolated Sandbox Landing Zone, which does not provide connectivity to NESO's SIMONE License Server instance, which is required for all components of the FastPress tool to function.				
6	Migration of existing code base from the NG ESO-DAP-AAEs to the NESO ADO environment. A delay in enabling this transfer will impact the engineering timeline	2	4	8	Clear direction needs to be provided by a dedicated representative from the NESO cloud team
7	Migration of the SIMONE Licence Server from the NG to the NESO cloud. If this does not occur during the project lifetime then the FP tool will not function.	5	2	10	NESO and NG need to ensure the migration completes.
8	Issues setting up the cloud deployment of SIMONE, caused by a lack of options for Windows compute deployment on Azure	5	1	5	Attempt alternative deployment ideas, e.g. with/without Docker, different Azure services, permanent compute. Resolved via experimentation, enabled by provisioning of Sandbox
9	The Transient Solver requiring excessive computation time to solve. The transient solver is applied in the time dimension, which can greatly increase the complexity and runtime of solutions	3	3	9	We will select and design algorithmic approaches that are computationally efficient and scalable We will build on the existing BSS work to apply learnings about performance improvements that we have previously delivered. We will test and optimise for run time of solutions early in development
10	Volume of licence requirements: A deployed tool will likely require an increased number of licenses to scale and	2	4	8	We will quantify license requirements for the finished product early as part of our value calculation work

	provide timely solutions. NESO will need to bear the cost of this				We will build the system to be as efficient as possible to minimise the number of licenses required
11	Design Risk: Designs may not be aligned with existing or upcoming unified NESO design standards	2	2	4	We will sync with the relevant NESO teams early and establish acceptance criteria
12	A move to 64 bit SIMONE licenses from NESO may cause issues with data compatibility and require a rework of our current interface with SIMONE which assumes a 32 bit software	2	2	4	Fair warning of such a move would allow us to anticipate and test any changes that need to be made to the tool

Table 5. Beta phase project risks

7. Conclusion

The Alpha+ phase of the FastPress project has successfully demonstrated the feasibility of an automated, algorithmic approach to solving static SIMONE scenarios. By building on the foundation of the previous Alpha phase, we have achieved a number of key deliverables that move the project from a proof-of-concept to a pre-production level of performance.

Firstly, the Base Scenario Solver (BSS) and Limit Violation Checker (LVC) have been significantly developed to a pre-production performance level from a data science viewpoint. This has allowed the tool to consistently and reliably solve static scenarios, a crucial and time-consuming task for gas analysts. The BSS, which uses a genetic algorithm to find optimal settings, has been refined to successfully solve a range of test scenarios, demonstrating a performance on the same order of magnitude as manual analyst solve times. The LVC, a core component, has been validated to accurately check scenarios against operational limits.

Secondly, we have developed two new features, the Capability Limit Finder (CLF) and Resilience Testing (RT), to a proof-of-concept (PoC) level. The CLF successfully automates entry and exit capability analysis. While it hasn't reached its maximum potential yet, it is already increasing flows, and promising results suggest maximum or near-maximum capacity can be achieved with further development. This feature has been shown to work across all RIIO zones and various demand levels. The RT feature, even at a PoC stage, showcases resilience testing by solving scenarios under compressor unavailability, demonstrating the ability to reconfigure solutions and allow for resilience testing both in base scenarios and in entry/exit capability scenarios.

Furthermore, we successfully deployed the Local Component Demonstrator (LCD) onto an individual user's laptop, providing an interactive user interface for direct feedback, which allowed us to clarify nuances of technical requirements, identify future changes to improve usefulness, and understand how the tool can already fit into user workflows as-is. From an engineering standpoint, a key milestone was also achieved by successfully installing and running SIMONE on a Windows Azure Container instance. This achievement mitigates the main risk for a full cloud deployment and confirms the technical viability of the intended target architecture.

Finally, we have completed the initial scoping and understanding of features for the upcoming Beta phase. This includes a high level plan, incorporating new features such as the Transient Scenario Solver and the Infrastructure Timeline Optimisation tool. The Beta phase will also focus on transitioning the developed features to the new common model for the network, improving efficiency, and formalising the tool for production use within the NESO cloud environment. The work completed in Alpha+ provides a robust foundation for this future development, paving the way for a more productive and efficient workflow for gas analysts.

8. Appendices

8.1. 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
ITO	Infrastructure Timeline Optimisation
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
SST	Scenario Scoring Tool