Aerothermal Performance of an Open Source Methane-Fueled Rocket Engine

Citation for the original published paper (version of record):

Hansson, A., Nyberg, A., Capitao Patrao, A. et al (2025). Aerothermal Performance of an Open Source Methane-Fueled Rocket Engine. FT2025: Proceedings of the 12th Swedish Aerospace Technology Congress, 215

(article starts on next page)

FT2025
SWEDISH AEROSPACE AT THE CROSSROADS
The 12th Swedish Aerospace Technology Congress
FT2025 in Stockholm
October 14-15, 2025

Aerothermal Performance of an Open Source Methane-Fueled Rocket Engine

Alexander Hansson[1], Atle Nyberg[2], Alexandre Capitao Patrao[3], Jan Östlund[4] and Tomas Grönstedt[5]

[1]*Student, Chalmers University of Technology, Gothenburg, Västra Götaland*
E-mail: Alexander.ha@live.se
[2]*Student, Chalmers University of Technology, Gothenburg, Västra Götaland*
E-mail: Atle.a.nyberg@gmail.com
[3]*Senior Engineer, GKN Aerospace, Trollhättan, Västra Götaland*
E-mail: Alexandre.CapitaoPatrao@gknaerospace.com
[4]*Technical Authority, GKN Aerospace, Trollhättan, Västra Götaland*
E-mail: Jan.Ostlund@gknaerospace.com
[5]*Full Professor, Chalmers University of Technology, Gothenburg, Västra Götaland*
E-mail: tomas.gronstedt@chalmers.se

Abstract

This study investigates the feasibility of integrating all relevant physics into a single simulation of a rocket engine nozzle, including the supercritical coolant, the supersonic reacting flame, along with the heat transfer between them. Traditionally, multiple simulations coupled with a thermal boundary condition would be utilized, which is time-consuming as it requires manually iterating between the simulations. A holistic simulation approach within one simulation was therefore developed. By comparing two CFD codes, STAR-CCM+ and Fluent, their suitability for this holistic simulation approach was evaluated based on accuracy, time consumption, and the required amount of manual input needed to complete a holistic simulation.

The methodology involved setting up comprehensive models in both CFD codes, ensuring that all relevant physical phenomena were accurately represented. The simulations were run under identical conditions to ensure a fair comparison. The results indicated that both codes produced outputs consistent with previous studies and with each other, validating the holistic approach. However, STAR-CCM+ demonstrated greater efficiency, making it more suitable for practical applications.

These findings suggest that a single, integrated simulation approach can significantly streamline the design and analysis process for rocket engine nozzles, potentially leading to more efficient and cost-effective development cycles.

Keywords: CFD, Supercritical Methane, Reacting Species Transport, ANSYS Fluent, STAR-CCM+, Conjugate Heat Transfer.

1. Introduction

Methane-fueled (CH4) rocket engines offer an attractive trade-off in terms of performance and complexity of use. For performance, the common choice of fuel falls on liquid hydrogen (LH2) due to its high specific impulse (Isp) and remarkable cooling capacity [1]. Hydrogen has been used historically on the Space Shuttle and Delta IV rockets, and presently on the European Ariane 6 and Japanese H3 rockets. Additionally, hydrogen is the fuel for high performance upper stages such as the Centaur V (Vulcan) and the Upper Liquid Propulsion Module (Ariane 6). However, there are also a number of challenges associated with the use of hydrogen - namely its cryogenic storage temperature (~20 K) and low density [2], which impacts logistics, handling, launcher size, and turbopump design. Kerosene on the other hand is storable at ambient conditions and features a considerably higher density – leading to less complicated operations and a smaller launcher size. The drawbacks are primarily lower Isp and significantly lower cooling capacity compared to hydrogen. The lower specific impulse results in an increase in required propellant mass, while a lower cooling capacity leads to higher nozzle wall temperatures and a more constrained choice of structural material.

Methane offers a compromise with respect to the advantages and disadvantages of hydrogen and kerosene. It is a better coolant and offers higher Isp than kerosene, and can be stored at mildly cryogenic temperatures. Density is lower than for kerosene but significantly higher than for hydrogen, resulting in lower required pumping power and the possibility to run both the CH4 and LOX turbopumps on a single turbine [3]. In recent years, there has been an addition of several new methane fueled engines, some operational (SpaceX Raptor, Blue Origin BE-4, LandSpace TQ-12), and some under development (ArianeGroup Prometheus). This new development has also led to the design and release of open-source designs from academia such as the RFZ model from the RETALT project in order to further research and collaboration.

This paper carries out aerothermal simulations of a Nozzle Extension (NE) modeled on the flamewall contour of the TIC nozzle of the RFZ model engine [4]-[5]. This NE serves as a computational test case for calculating its performance and for comparing the simulation tools ANSYS Fluent and STAR-CCM+. The NE consists of cooling channels with a rectangular cross-section and the cryogenic methane is modelled as a real fluid, while the flame is simulated as a chemically reacting flow. Between these two domains a conjugated heat transfer model is setup for the NE wall. Results such as performance (I_{sp}), nozzle flow profiles, and wall temperatures are presented.

Computational fluid dynamics (CFD) is an invaluable tool when designing and evaluating NE. In order to accurately simulate the nozzle, a wide range of physical phenomena need to be accurately accounted for, such as the chemically reacting flow, the cryogenic coolant and the heat transfer between them. Due to the extreme conditions the typical approach is to split the task up into multiple simulations, coupling them through boundary conditions to account for the heat transfer. This has become a manual and time consuming approach, highlighting an area that can be improved in the method.

2. Methodology

The following section outlines the methodology in this paper, including how the simulations were carried out, the numerical methods relied upon and their evaluation.

2.1 Cases

The project was split up into three simulations in order to gradually increase simulation complexity. The cases increase in complexity until reaching the final, holistic simulation with a representative rocket nozzle geometry. This method allowed for the development of a simulation method during the computationally light cases which would then be verified and tested in the later cases. The cases are described in subsequent sections (2.1.1 to 2.1.3), while the CFD approach used is outlined in section 2.2.

2.1.1 Case 1: Flow over Flat Plate

A two-dimensional simulation featuring two fluid domains separated by a flat plate of steel. One domain contains the hot supersonic flow of a rocket flame and the other the cryogenic, supercritical methane, see Figure 1 for a representation of the geometry. This case was treated as a test case for the study of how the two CFD codes differed in methods and schemes. It was also used for the development of the method to simulate both fluid flows simultaneously (i.e cryogenic coolant and supersonic flame). The main areas of interest were the factors contributing to the heat transfer within and between fluids and solids. Previous work was studied to find differences between the CFD codes and their claims tested in the context of this case. A previous project [6] indicated the turbulence models, when using k-ω SST, as a primary contributor to the difference in results yielded by the two CFD codes. Thus investigating the methods and constants utilized by the two codes was the main focus after the development of an efficient simulation setup.

The mesh for the first case was generated using ANSYS ICEM, by creating a structured hexa-mesh with one-to-one face matching between the domains. This mesh was iterated upon to find the required layer number and height to achieve a suitable resolution of the viscous and thermal boundary layer (a wall y^+ between 0.1-1) which was utilized in the creation of the third case's mesh.

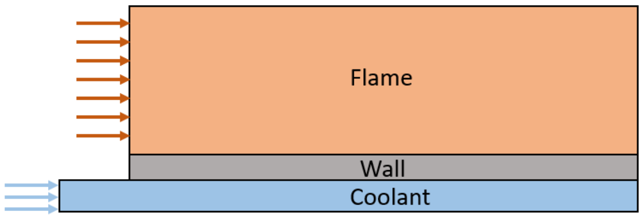

Figure 1: Flat plate geometry and regions. The figure is not to scale.

2.1.2 Case 2: Representative Cooling Channel

A representative three-dimensional cooling channel geometry with wall roughness was adopted from the work of Pizzarelli et al. [7]. The simulation aimed to be an introduction to accurately simulating a dense, non-ideal methane under supercritical pressure, where the results would be compared to Pizzarelli [7] and Ricci et al. [8] for verification against experimental results. This case was included to determine the accuracy of the CFD codes' simulation of supercritical and cryogenic fluids. In this case the equivalent sand grain roughness was found following the same method as was presented in [8], by completing a simulation without heat transfer (cold case) and matching the pressure loss of experimental data. This project used test "0A" as a reference for the cold case and test 2 for the hot case where heat was added. The geometry for the second case can be seen in Figure 2, where the boundaries colored blue were those set to be symmetric. The simulations of the second case utilized a mesh provided by GKN Aerospace Trollhättan which was made using ICEM.

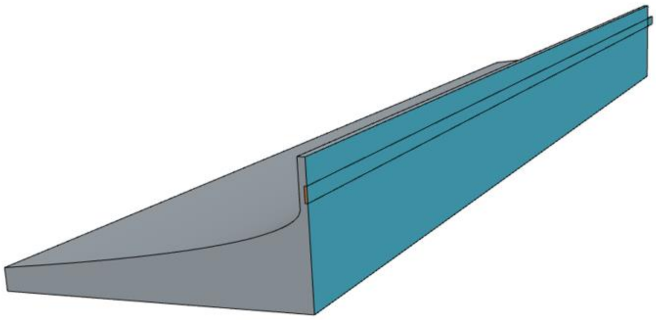

Figure 2: Cooling channel geometry.

2.1.3 Case 3: Representative Nozzle Geometry

This case entails a simulation of a generic rocket nozzle internally cooled with methane. The simulation followed the same CFD procedure as the first case (see section 2.2 for details), now with a more complex and three-dimensional geometry. The simulation was done with a nozzle flow made up of a multi-component gas resembling the exhaust gases of a rocket combustion chamber at the nozzle inlet. The mass fraction of each species were found with the NASA CEA code [9]. The final part of the project, a simulation approach containing all relevant physics inherited from the previous cases, was now simulated in a generic but representative nozzle geometry to verify if the method was reliable and accurate. The accuracy was determined from a comparison with the numerical approach of [4] along with the verifications from the second case of this paper. The suitability of the codes would also be determined from the amount of manual input and time required to complete such a simulation. Figure 3 displays the geometry for the third case showing the nozzle geometry, along with the thin cooling channel stretching across the NE. The blue surface are the periodic sides of the domain.

ANSYS ICEM was used to construct the mesh of the third case, using the findings from the first case to generate a preliminary grid. To ensure mesh independence a mesh study was conducted where finer meshes were generated until a difference in the flamewall temperature, channel pressure loss and coolant heat pickup differed less than 1% between meshes.

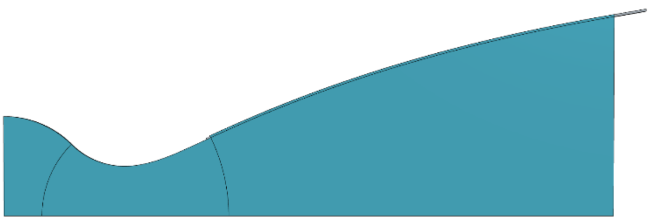

Figure 3: Rocket nozzle geometry.

All the cases were simulated in both ANSYS Fluent and in STAR-CCM+ with the same models, meshes and schemes to allow for a fair comparison between them. The comparison was done with properties relevant to the flame such as I_{sp}, velocity profiles, Mach contours and wall temperatures but also for properties relevant for the coolant like cooling channel wall temperatures, heat pickup and pressure loss. The goal of the project was to determine if a holistic nozzle simulation was viable but also which of the two aforementioned CFD codes would be most suitable for such an application. The comparison was based on both the accuracy and time needed for a code to produce results.

2.2 CFD

This section contains the numerical approach for the simulations throughout the project. All simulations were run in both STAR-CCM+ and ANSYS Fluent where the models, schemes and settings were kept as similar as possible, allowing for identification of differences between the CFD codes from the mismatch in results.

2.2.1 Thermophysical Properties

Liquid methane, used to cool the nozzle extension, enters the cooling circuit at supercritical pressure and subcritical temperature, exhibiting dense, non-ideal behaviour. As it absorbs heat and transitions to a fully supercritical state, ideal gas assumptions remain invalid throughout the flow. To account for this, the fluid was modelled as a single-phase real gas using thermophysical property tables provided by GKN Aerospace, which define pressure- and temperature-dependent values for simulation.

The reactions in the exhaust gases were described using a reduced "Zhukov-Kong Mechanism" [10] containing 23 gaseous species. The data contains tabulated material properties for each of the individual species along with data of the possible reactions. Experimental validation has been carried out by Zhukov that showed good agreement between the data and the experimental results within temperatures of 870-1700K and pressures of 2-500 atm (2.03-506.6 bar). This reaction scheme was imported into the “laminar flame concept” finite rate chemistry model.

2.2.2 Models and Schemes

In the first and last case the simulations used a steady, coupled, and pressure based solver. The RANS k-ω SST (Menter 1994) turbulence model [11] was used throughout the project. The energy equation was solved in order to capture the heat transfer in and between the materials. Imported reaction schemes were used for the flame and rgp-formatted files for modeling the coolant. In the second case the simulation was set up to resemble that of [8], resulting in the segregated solver using SIMPLEC for Fluent and SIMPLE for STAR-CCM+. This is because the SIMPLEC method was available for steady state simulations in STAR-CCM+. The second case included wall roughness modeling, utilizing the “Rough displaced origin” method in both CFD codes. The third case utilized the same settings as the first, now in a more complex geometry representative of a rocket thrust chamber with cooling. A three-dimensional structured mesh using periodic boundaries making up one degree of the entire thrust chamber was generated using ANSYS ICEM.

2.2.3 Simulation Strategy

To maintain numerical stability, the simulations were initially run with a minimalist setup, excluding effects such as wall roughness, chemical reactions and inter-domain heat transfer. The simulation complexity was incrementally increased by introducing one physical model at a time, allowing the flows to stabilize before the addition of new, more demanding, physics.

This method was used throughout the project and allowed for identification of errors within each model. The flame, for example, was initiated as a non-reacting, multi-component gas before activating reactions and the coolant domain wall roughness was ramped from 0 µm to the final value.

STAR-CCM+ was run on a cluster in server mode, in contrast to Fluent that did not feature that functionality during the duration of this project, which allowed the user to interact with the simulation as if the simulation was carried out on a local machine. This together with the ease of pausing/deactivating continua in STAR-CCM+ allowed for a temporary decoupling of the fluid regions, letting them converge without the changing influence of the other. This was used to iterate between regions to keep the simulations stable and less time consuming. These functionalities were not featured in Fluent which was instead run fully coupled.

3 Results

The following section contains the results from the simulations of the three cases along with the comparison with the work of [8] and [4].

3.1 Flat Plate

The flat plate case showed that the holistic simulation approach was possible in both CFD codes for a simple two-dimensional geometry, with similar results between the codes. The temperature profiles presented in Figure 4 were produced with a P_{rt} of 0.85 in both codes and shows an average difference of 13.8 K at the flamewall and 3.5 K at the hotwall.

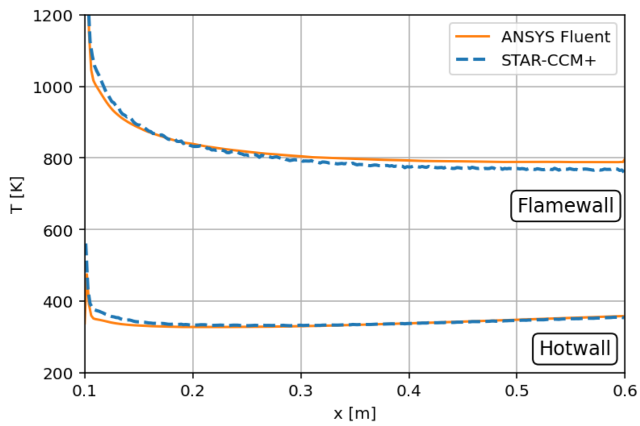

Figure 4: Wall temperatures profiles of the flame- and hotwall in each software.

Important lessons were learned from the simulations – namely how each software can be used in an effective manner to reach convergence. An advantage that STAR-CCM+ showed was that the two fluid regions and solid domain could be deactivated (see regions in Figure 1). This enabled regions to be paused, essentially decoupling the two flows. The feature allowed for iterations between the flows, letting them find a stable solution before coupling them. This feature was present in Fluent but was a cumbersome process that is not recommended by the authors. Instead it is recommended that the simulation in Fluent remains fully coupled through the entire process.

Differences were also found between the CFD codes. Firstly, the default settings differed and needed to be configured correctly to produce similar results. Of these settings viscous heating was the most impactful as it was seen to decrease the wall temperature profile up to 200 K if disabled, this essentially meant that heat generated by the viscous dissipation wasn't taken into account in the energy equation. Secondly, the codes produced different amounts of turbulence at the inlets even though the same turbulent intensity and turbulent viscosity was specified, see Figure 5 for a comparison of the calculated viscosity at the inlet boundary for the flame and coolant regions. This was more important in this case since the geometry was flat and the turbulence entering from the inlet would exceed the production of turbulence in the domain. Additionally, the species averaging schemes differed, and were changed in STAR-CCM+ to match those of Fluent. Lastly, there were settings that yielded minor influence but contributed to a closer match in results, among them was: "Compressibility correction" in STAR-CCM+ that was deactivated as it worsened the resolution of the boundary layer.

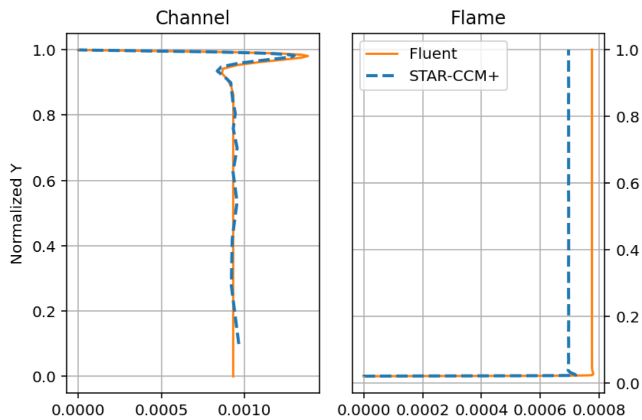

Figure 5: Turbulent viscosity in the two regions in each software along the normalized perpendicular distance from the wall.

Further discrepancies were found when refining the first layer heights of the mesh. Fluent had a standard geometric tolerance and minimum wall distance of $10^{-12}\ m$ while STAR-CCM+ used $10^{-6}\ m$. For the code to take account of cells smaller in size the minimum wall distance and interface tolerances needed to be decreased.

3.2 Case 2: Representative Cooling Channel Geometry

The second case replicated the experiments of [8] in CFD to validate the simulation method used for the methane in both CFD tools. The results in Figure 6 shows that both tools were able to produce similar wall temperature profiles compared to the experimental data, validating the simulation method of the methane for the rest of the cases.

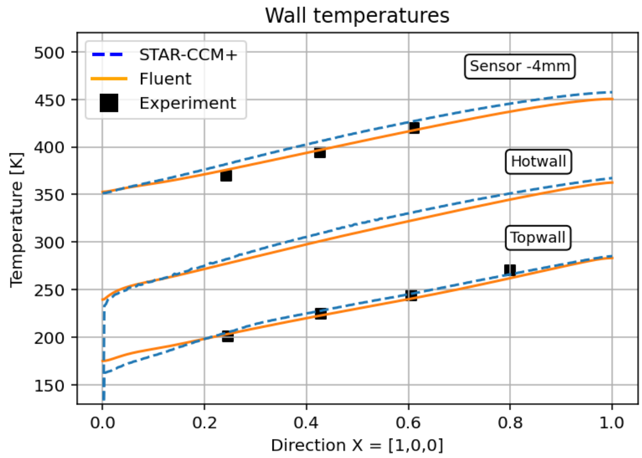

Figure 6: Wall temperature profiles for different positions.

To achieve these results the turbulent Prandtl number was changed from 0.7 to 0.85 in STAR-CCM+ to match that of Fluent. Running the simulation with the default P_{rt} showed a roughly 25 K offset from the experiment and Fluent, indicating that a P_{rt} of 0.85 was a more realistic value for modelling the coolant. A P_{rt} of 0.85 was therefore used during the rest of the project.

3.3 Rocket Nozzle Extensions

The third case featured a full three-dimensional rocket nozzle with a cooling channel. The simulations showed a close match between the programs, which can be seen in Figure 7 of the wall temperature profiles. However one difference was that Fluent begins at a lower wall temperature than STAR-CCM+, although this was limited to only the first 10 cells. The profiles had an average difference of roughly 3 K at both walls. Additionally the mass flow averaged heat pickup of the coolant was 48.8K and 49.63K for STAR-CCM+ and Fluent respectively, the percentile difference being 0.17%.

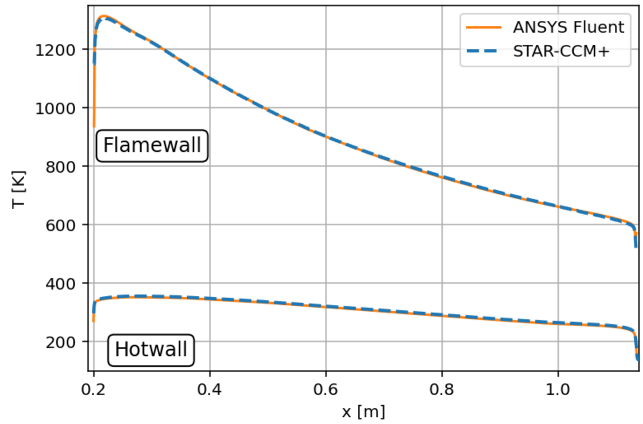

Figure 7: Wall temperature profile in each software.

An important difference between the CFD codes was that STAR-CCM+ got a higher wall y^+ for the flamewall, see Figure 8, indicating that STAR-CCM+ needs a finer mesh than Fluent to properly resolve the viscous boundary layer at the flamewall. The comparison was only made for the same mesh, but practically the results mean that STAR-CCM+ will require a finer boundary resolution, increasing the computational demand and the simulation time compared to Fluent.

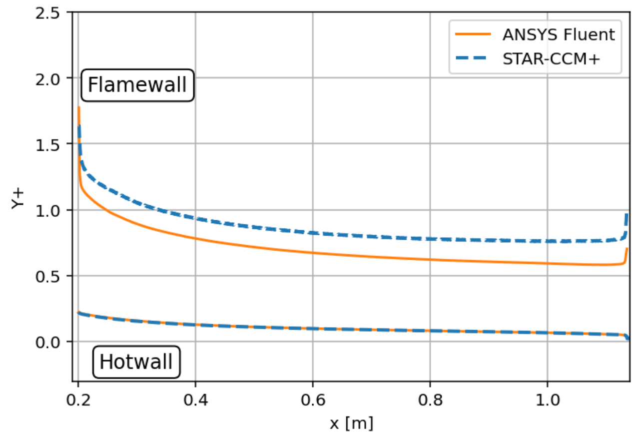

Figure 8: Wall Y+ profiles in each software.

Figure 9 presents a Mach contour of the rocket nozzle for both Fluent and STAR-CCM+, showing how similar the flows behave with iso-lines. Some differences can be seen close to the centreline, where there is a Mach gradient across the boundary in STAR-CCM+. This is also present for the simulation in Fluent but it occurs further downstream close to the exit of the nozzle.

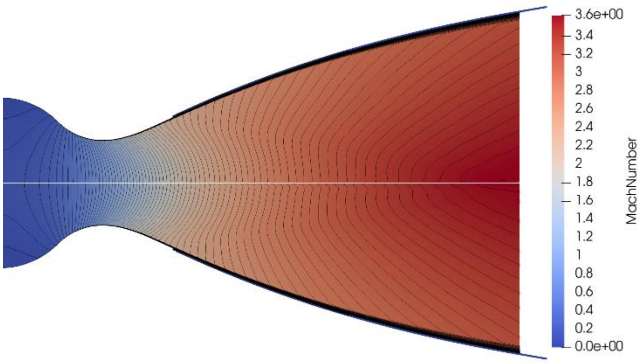

Figure 9: Mach contour of the rocket nozzle where the top part is from Fluent and the bottom from STAR-CCM+.

The simulation results were also compared to the simulations where the parabolic TIC geometry was sourced from. The reference simulation was conducted with a constant wall temperature of 500 K, which differs from the simulation conducted in this report. Therefore, some difference in performance was expected but the comparison was still useful to ensure that the performance is on the right order magnitude and that the two CFD codes produces similar results. See Table 1.

Table 1: Comparison of performance for the CFD codes against the reference, result presented in percentage of reference.

Variable	Fluent	Star	Reference
F_N [kN]	-0.16%	-0.21%	809.3
I_{SP} [m/s]	-3.8%	-3.7%	3487
$\dot{m}$ [kg/s]	+3.8%	+3.8%	231.8

4 Discussion

The Mach contour plot in Figure 9 shows a difference at and connecting to the central axis of the NE. The geometry was sourced from [4] which had a similar occurrence for nozzles with inferior design leading to disruptions in the boundary layer at the nozzle walls. With that in mind the error could be the result of an insufficient boundary layer resolution, refining the mesh could potentially reduce this error. This would also explain why the phenomena was more pronounced in STAR-CCM+ as it has been shown that it needs a slightly finer mesh.

When considering the wall temperature plots a remaining, although small, error can be seen. The first step would be to refine the mesh to reduce the error discussed in the previous paragraph. The difference in wall temperatures were shown to shrink through the cases even through the method remained the same. This could be the consequence of the differences in the inlet turbulence and the formation of the boundary layers for both regions. Due to the simple and straight geometry of the first two cases, the inlet turbulence might have been the primary source of turbulence in the flow. In the third case, with a bending and turbulence inducing geometry, that would no longer be the case, leading to production of turbulence and better agreement between the temperature profiles.

Both STAR-CCM+ and Fluent were shown to reliably reach an accurate solution after the adjustments found through the first case which allows for the choice between the CFD codes to be based on other factors. Time consumption is one, where STAR-CCM+ and Fluent needed about 4300 or 5500 core hours respectively to complete a simulation of the third case. STAR-CCM+ also featured the possibility of pausing regions along with having different model settings for the different regions giving it an edge in the simulation. STAR-CCM+ cases did however require a longer set up before a simulation, partially due to it not supporting rgp-formatted tables therefore requiring the user to reformat the data. This problem was minimized with the use of a template file that only required the user to import a mesh and set the boundary conditions before simulation. With all this in mind the authors of this article generally considered STAR-CCM+ as the superior choice for these simulations due to its speed and user experience.

The possibility of running two meshes, an axisymmetric for the flame and a three-dimensional for the channel and coolant was never investigated but would be a method of decreasing the computational intensity of these simulations. If supported within a CFD code the cost of simulating the flame would be much lower and an alternative to running the whole geometry as a 3D mesh.

In this project only the coolant flow was verified using experimental data showing that the turbulent Prandtl number used in those simulations gave realistic answers. This check was not done for the flame, meaning a different Prandtl number could be more beneficial for the simulation. In this case, STAR-CCM+ would gain an edge as it allows for different turbulent Prandtl numbers for different continua, while Fluent does not.

5 Conclusion

Both CFD codes were able to perform a holistic simulation of a rocket engine nozzle including all relevant physics. STAR-CCM+ was slightly faster compared to Fluent. The manual input for both programs was low though higher for STAR-CCM+, this could however be minimized using a macro within STAR-CCM+. The results were accurate when comparing to the previous work of [4] and [8] but they have yet to be compared against GKN Aerospace’s internal methods. The final conclusion of the code comparison was that the simulations were viable though heavy and time consuming when performed in fully three dimensional meshes.

The two CFD codes feature both pros and cons when compared against each other. The iteration method used in STAR-CCM+ required a larger amount of manual input, this method was used to save time but a run using the same method as Fluent was never tested due to its higher time consumption. STAR-CCM+ cases took longer to set up due to the abundance of settings needing to be changed, this problem was partially solved by creating a template file containing all the necessary settings and files, thus evening out the amount of time required to set up a case. STAR-CCM+ did not support rgp-tables, they had to be reformatted using a python script and imported into the CFD code, which was a time-consuming procedure that was negated by using the template file. Despite the significant amount of manual input required for setting up a case within STAR-CCM+, it was considered the superior software when applying the holistic simulation approach. This is because it reached a nearly identical solution to Fluent in a significantly shorter amount of time while being more user-friendly.

Acknowledgements

This project was conducted as a collaboration between Chalmers University of Technology and GKN Aerospace Sweden in the form of a master thesis. GKN is a leading aerospace manufacturer, specializing in high performance engine components for civil and military crafts. The Swedish branch of GKN also specializes in the design and manufacture of rocket engine components for the European Space Agency (ESA) and the Arianne rockets. Among them is the rocket nozzle extension (NE).

Nomenclature

Designation	Denotation	Unit
y^+	Dimensionless wall distance	-
$P_{r,t}$	Turbulent Prandtl Number	-
I_{sp}	Specific Impulse	m/s
$\dot{m}$	Mass flow	kg/s
F_N	Net thrust	N
T	Static temperature	K

Abbreviations

Designation	Denotation
CFD	Computational fluid dynamics
rgp	Real gas property
NE	Nozzle extension

Bibliography

[1] G.P. Sutton and Oscar Biblarz, “Rocket propulsion elements”, John Wiley & Sons, 2011.

[2] D. K. Huzel and D.H. Huang, ”Design of liquid propellant rocket engines”, NASA-SP-125, 1967.

[3] A. Ianetti et al. ”Prometheus, a LOX/LCH4 reusable rocket engine”, in Proceedings of the 7th European Conference for Aeronautics and Space Sciences (EUCASS), Milan, Italy. 2017.

[4] S. Karl, T. Bykerk and M. Lauretti, “Design of a truncated ideal nozzle for a re-usable first stage launcher,” in HiSST: 3rd International Conference on High-Speed Vehicle Science & Technology, 2024.

[5] E. Moritz and T. Bykerk, "A standard model for the investigation of aerodynamic and aerothermal loads on a re-usable launch vehicle-second stage geometry", 3rd International Conference on High-Speed Vehicle Science and Technology (HiSST). 2024.

[6] Z. Ying, Z. Xingwang and Q. Chen, “Comparison of STAR-CCM+ and ANSYS Fluent for Simulating Indoor Airflows,” Springer, 2018.

[7] M. Pizzarelli, F. Nasuti, V. Raffaele and F. Battista, “Validation of Conjugate Heat Transfer Model for Rocket Cooling with Supercritical Methane,” Journal of Propulsion and Power, vol. 32, pp. 1-8, 2016.

[8] D. Ricci, P. Natale, F. Battista, E. D'Aversa, R. Pellegrini, H. Asakawa and H. Negishi, “Investigation on the Transcritical Behaviour of Methane and Numerical Rebuilding Activities in the Framework of ASI/JAXA Cooperation Project,” 2016.

[9] S. McBride and B. J. Gordon, Computer program for calculation of complex chemical equilibrium compositions and applications. Part 1: Analysis, NASA Reference Publication 1311, 1994.

[10] V. Zhukov, “Kinetic model of alkane oxidation at high pressure from methane to n-heptane,” Combustion Theory and Modelling, vol. 13, pp. 427-442, 2009.

[11] F. R. Menter, “Two-equation eddy-viscosity turbulence models for engineering applications,” American Institute of Aeronautics and Astronautics (AIAA), 1994.