Comparing Hydrogen and Jet-A for an N+3 Turbofan with Water Recirculation using Gradient-Based Optimization

Peter N. Atma*, Andrew H .R. Lamkin†, and Joaquim R. R. A. Martins‡
University of Michigan, Ann Arbor, MI, 48109

Advances in commercial propulsion technology led to the development of efficient high bypass ratio turbofan engines with larger overall pressure ratios and internal temperatures. Current trends suggest that geared ultra high bypass ratio turbofans are the next generation of commercial propulsion systems. Furthermore, the emphasis on decreasing emissions has driven the exploration of hydrogen-powered aircraft, adding to the already challenging design space. Carrying and burning hydrogen introduces complexity and weight penalties that we must offset using the fuel's thermodynamic and chemical properties. In this study, we create a closed-loop water recirculation system with a zero-dimensional thermodynamic model and compare the benefits between jet-A and hydrogen fuels. We perform a gradient-based optimization parameter sweep to explore the trade-offs between performance and emissions using both fuels with water recirculation. The results quantify the design space for next-generation propulsion concepts that can take advantage of hydrogen fuel's advantageous thermodynamic properties to reduce emissions and improve performance.

I. Introduction

The effects of climate change are pushing the aviation industry towards hydrogen-fueled propulsion systems as a solution to reduce emissions. N+3 technology estimates for turbofan engines that burn hydrocarbon fuels suggest that higher efficiency can be achieved by designing ultra high bypass ratio (UHBR) engines with small cores and high overall pressure ratios (OPR). Higher OPR and smaller cores challenge the limits of compressor and turbine design, placing an upper bound on potential performance and emissions improvements. Switching to hydrogen as the primary fuel source reduces carbon dioxide emissions immediately, but adds complexity and weight that offset the benefits. However, hydrogen is a versatile fuel with advantageous chemical and thermodynamic properties that can be exploited to increase the performance and reduce emissions. We introduce a closed-loop water recirculation model that demonstrates the possible efficiency gain when hydrogen is used for purposes other than combustion.

Water recirculation is the process of extracting water from the exhaust stream of a propulsion system and injecting it upstream of the combustor as finely atomized droplets. NASA, Boeing, and Rolls-Royce studied this concept and suggested that this technique reduces the NOx emissions as much as 47 percent [1]. Additionally, water recirculation improves fuel efficiency and thrust output with lower combustion temperatures that can improve the lifetime of turbine blades and reduce noise [1]. Traditional propulsion systems that burn hydrocarbon fuels would require external water storage on the aircraft because they do not produce enough in the exhaust for recirculation. The added weight of tanks, pumping, and ducting makes this concept infeasible for a conventional aircraft. The main product of hydrogen combustion is water vapor and can thus be recovered from the exhaust stream. Recirculating water vapor from the exhaust of hydrogen combustion reduces the requirement for storage tanks and allows for the creation of a closed loop system inside the propulsion cycle.

Zero-dimensional cycle modeling is an efficient tool for predicting the initial design, performance, and emissions of new propulsion concepts. Zero-dimensional analysis uses a first-principles approach with a chemical equilibrium analysis (CEA) thermodynamics solver [2] that considers the molecular species of different fuels. The industry standard for thermodynamic cycle analysis is the Numerical Propulsion System Simulation (NPSS) framework [3]. NPSS is a modular object-oriented library that models engine components as individual blocks with several thermodynamic solvers. Hendricks and Gray [4] created a new tool called pyCycle with the same functionality as NPSS with analytical derivatives for each engine component and thermodynamic solver [5]. pyCycle is built on top of the OpenMDAO framework [6] to enable gradient-based optimization and leverage hierarchical nonlinear solver structures for robustness.

*MSE Student, Department of Aerospace Engineering, AIAA Student Member
†Ph.D. Candidate, Department of Aerospace Engineering, AIAA Student Member
‡Professor, Department of Aerospace Engineering, AIAA Fellow

In this work, we analyze the thermodynamic benefits of a closed-loop water vapor recovery and water injection system in a high-bypass turbofan engine. We develop pyCycle components for water injection and vapor recovery to quantify the benefit of a closed loop recirculation system. We use gradient-based optimization to minimize fuel burn subject to performance requirements using both jet-A and hydrogen at a range of flight conditions. The optimized results show the trade-off between complexity, performance, and efficiency for jet-A and hydrogen fuels.

This work is organized as follows. First, in Section II, we introduce the turbofan model and explain the water injection and water recovery components. In section III the implementation of the multipoint optimization problem is discussed. Finally, we present the optimized results and discuss the design space in section IV.

II. Methodology

A. Engine Model Overview

The UHB turbofan model is the NASA advanced technology "N+3" engine [7]. The N+3 reference cycle represents a UHB ratio geared turbofan that could be available in the 2030 to 2040 time frame. The flow path consists of an inlet that directs ambient air through a fan, followed by a duct that splits the flow into a core and a bypass stream, each ending in a core and bypass nozzle, respectively. The low pressure system is split into two mechanical subsystems. First, the fan is connected to the gearbox that reduces the shaft speed to decrease the fan tip speeds. Second, the gearbox attaches to the low-pressure shaft that connects to the low-pressure compressor (LPC) and low-pressure turbine (LPT). The high pressure compressor (HPC) is connected to the high pressure turbine (HPT) by the high pressure shaft.

We introduce the closed-loop water recovery system as a feedback loop that transports water from the exhaust to upstream of the compressors. The recovery system injects vaporized water into the core stream that reduces the combustion temperature due to heat absorption. The vapor recovery component is placed directly before the core nozzle to extract water from the exhaust and recycle it back to the injector.

In this section we explain the implementation and assumptions of the water recovery model. We present the full engine layout and provide details on the multipoint zero-dimensional modeling approach. The component flow interface and mechanical connections, including the water injector and water extractor, are depicted in Figure 1.

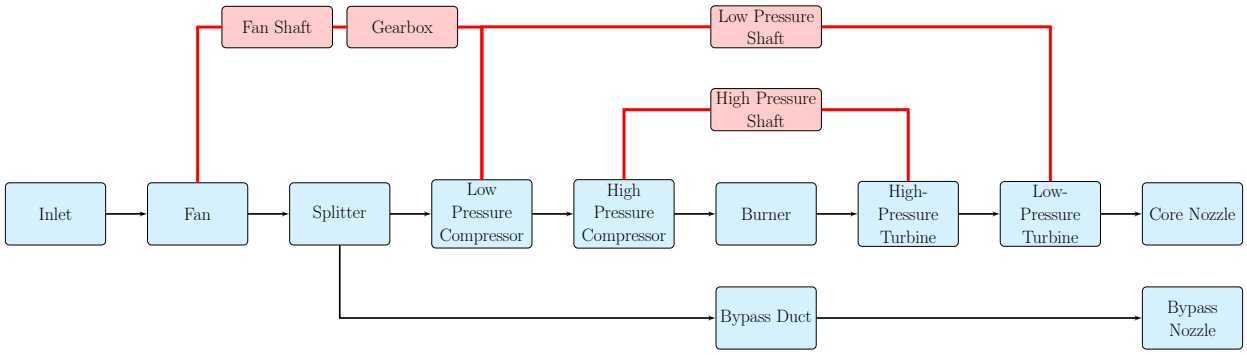

Fig. 1 Simplified layout of the N+3 engine cycle, adapted from Hendricks and Gray [4]. Black arrows are flow connections, red lines are mechanical connections, blue boxes are cycle elements, red boxes are shaft elements.

B. Water Recovery Model

We implemented the closed-loop water recovery system as a feedback system that extracts water from the exhaust stream and injects it upstream of the HPC. We chose this injection location based on claims from a study by NASA, Boeing, and Rolls-Royce [1] that water injection directly into the combustor is unnecessary. The water vapor recovery component sits downstream of the LPT and extracts water from the flow before it exits the core nozzle. The component flow interface and mechanical connections, including the water injector and water extractor, are depicted in Figure 2.

We modified the composition of the air mixture upstream of the combustor to account for humidity. pyCycle provides a `wet-air` dataset that introduces H_2O molecules to the composition of air. We prescribe atmospheric mass-specific humidity as a water-to-air ratio (WAR) that is defined as the ratio of H_2O to air in the reactants of the inflow mixture. Kalnay et al. [8] give the humidity values for each flight condition.

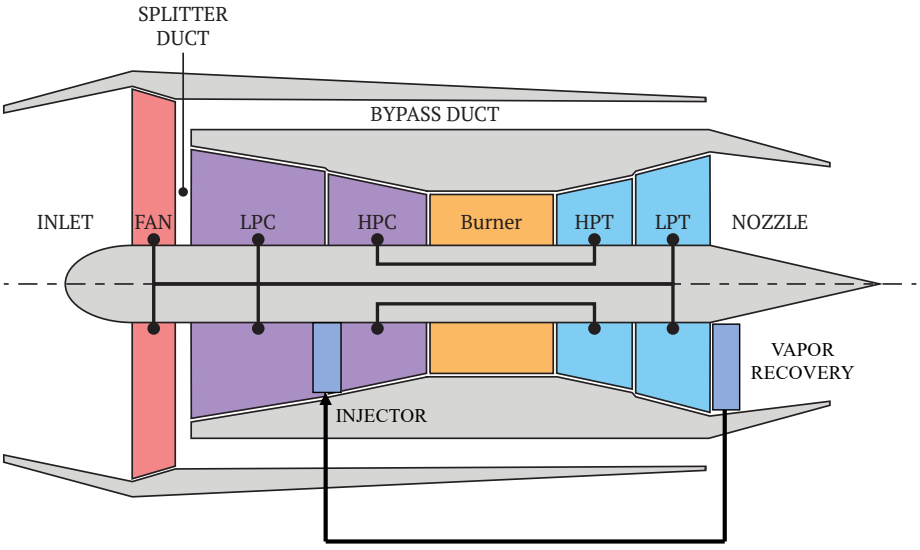

Fig. 2 The configuration of a high-bypass turbofan model with an integrated closed-loop water vapor recovery and injection system. The water vapor recovery system (extractor) extracts a fraction of the water in the core stream and reinjects it upstream of the high-pressure compressor. This diagram illustrates the feedback effect that this implementation has on the overall core flow.

We introduce two new thermodynamic models for water recirculation in pyCycle. A water injector adds water to the flow upstream of the HPC, while a water extractor diverts a portion of the water in the flow away from the exhaust stream. The injector operates similarly to fuel injection in the combustor. We determine a WAR that is analogous to the fuel-to-air (FAR) ratio in the combustor. This WAR is used to compute the chemical species present in the flow at the current thermodynamic state, determined by the incoming flow. The new species composition and thermodynamic state are determined using the pyCycle CEA solver [5]. The water injector inputs are the mass flow rate and mass fraction of water. We can solve for the mass flow rate on a mixture basis using the WAR, or directly by specifying the mass flow rate of water. A schematic of the injector is shown in Figure 3 where Y_{H_2O} is the mole fraction of water molecules.

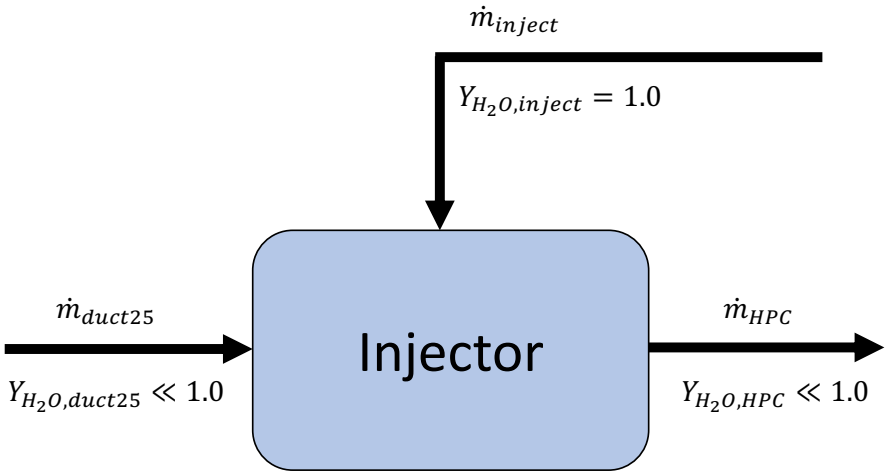

Fig. 3 Injector component schematic. Water from the extractor is simply injected into the core flow upstream of the high-pressure compressor.

The water extractor model diverts a fraction of a specific species from one flow path to another. The CEA solver

calculates the inflow composition and the extractor separates a specific species based on a mass fraction input. The composition of the core stream is updated to represent the remaining mixture after the extractor model removes the desired species from the incoming flow. We then solve for the thermodynamic state and flow path areas at the outflow of both extractor streams. A simple schematic of the extractor is shown in Figure 4 where Y_{H_2O} is the mole fraction of water molecules and $X_{H_2O,k}$ is the fraction of water that is recovered from the core stream.

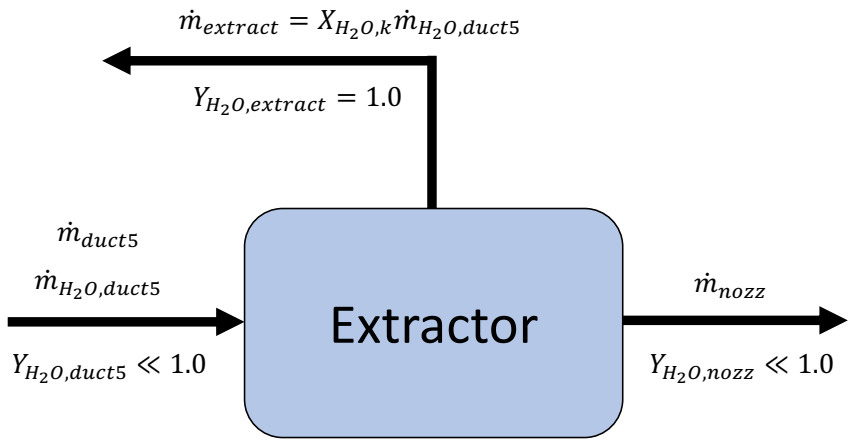

Fig. 4 Extractor component schematic. A certain mole fraction of the flow coming out of duct5 is comprised of water of which a certain specified fraction is extracted. The extracted water is routed back upstream to the injector and the rest is simply exhausted out the nozzle.

We connect one outflow stream of the extractor to the inflow stream of the injector to complete the water recirculation system. The model results in a mismatch between the mass flow rate upstream of the HPC and the mass flow rate exiting the core nozzle. To preserve conservation of mass, we treat the water recirculation as a nonlinear cycle that must converge before the engine calculation is physically balanced. We illustrate the water recirculation loop and the nonlinear solver connections in Figure 7.

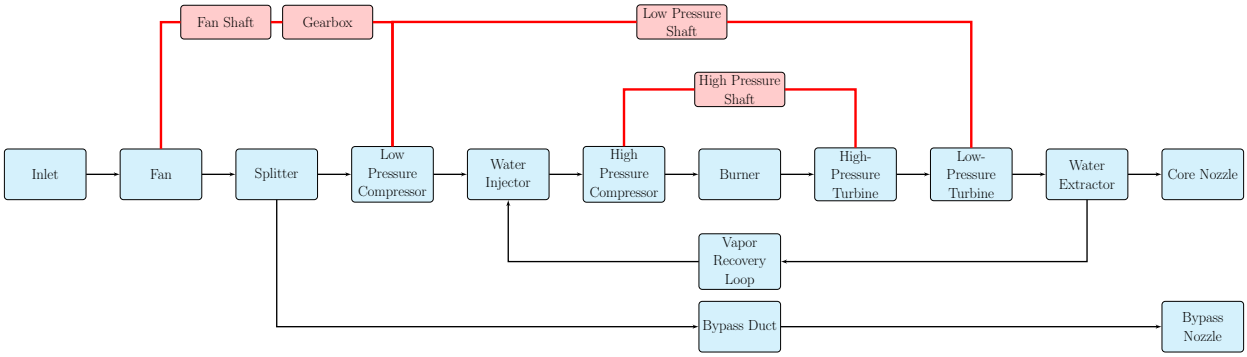

Fig. 5 Simplified layout of the N+3 engine cycle with the closed-loop vapor recovery. Black arrows are flow connections, red lines are mechanical connections, blue boxes are cycle elements, red boxes are shaft elements.

In pyCycle, the Cycle block contains all of the governing thermodynamic equations needed to model an engine such as CEA analysis, compressor and turbine models, and equations of state. These equations allow any arbitrary engine design to be assembled using the modular cycle elements. The Cycle block itself may not result in a valid model of the engine since some physical dependencies may not match. These physical dependencies are handled by the Balance block which solves a set of implicit state variables and nonlinear physical residual equations in the form:

$$\mathcal{R}_c = \sum_{i=1}^{n} x_i = 0 \tag{1}$$

where x_i is some physical variable such as a mass or energy flow rate. These residuals represent the physical conservation laws such as mass flow rate are converged to zero by the Solver. In pyCycle, a direct solver is used for linear equations and a Newton solver is used for nonlinear equations which utilizes the analytic partial derivatives provided by each OpenMDAO component. All residuals within the engine model are converged at the Cycle level for a given operation condition since all of the parameters downstream do not directly effect those upstream. However, with introducing the extractor and injector components, we add a feedback loop into the model where mass is removed downstream and added upstream. Therefore, the mass flow rates residuals of the closed-loop vapor recovery system shown below would need to be converged in the Multipoint Cycle explained in the next section.

$$\mathcal{R}_{c,\dot{m}} = \dot{m}_{extractor} - \dot{m}_{injector} = 0 \tag{2}$$

C. Multipoint Model

The N+3 engine model as it is implemented in pyCycle has many complex thermodynamic and performance connections between different operating conditions. Therefore, the N+3 engine model computes the engine performance at 4 different operating conditions in order to resolve the complexities between each operating condition on the final design. The operating conditions that considered are top-of-climb (TOC), rotating takeoff (RTO), sea-level static (SLS), and cruise (CRZ). The SLS condition represents the static performance requirements for the engine, the RTO condition represents the performance and cooling requirements at the point of rotation during takeoff, the TOC operating condition represents the performance requirements during a climb, and the CRZ operation condition represents the criuse performance of the engine. To account for each of these operating conditions, the N+3 model uses a technique called Multipoint Design Point (MDP) modeling to converge the model to an engine design that satisfies the requirements at each of these operating conditions. One of these operating conditions is specified as the design point, which in this case is TOC, and the other operating conditions are considered the off-design points. The design point for the model generally sets all of the engine sizing such as areas and pressure ratios across components. The off-design points are essentially the size of the engine at the design point but operating at the corresponding required thrust, cooling, and environmental conditions. The design point of the engine model is converged using a linear Newton solver which then passes the geometric sizing variables such as cross-sectional areas and map scalars to the off-design points and similarly converges those. Then, a nonlinear Newton solver is used to converge the overall model with respect to the variable connections between each operating condition. An XDSM diagram of the nominal N+3 engine model is shown in Figure 6 [4].

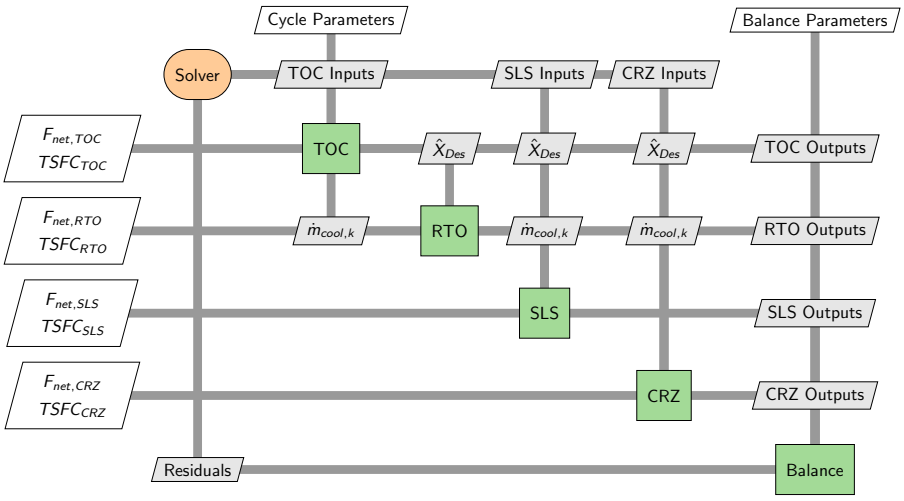

Fig. 6 N+3 engine multipoint setup XDSM diagram. This XDSM diagram shows how each of the operating conditions is coupled with each other and how the outputs are computed.

In the optimization problem described later, we set the mass flow rate of fuel, $\dot{m}_{fuel}$, as the objective function. Our comparison metric is $TSEC_{CRZ}$ but minimizing $\dot{m}_{fuel}$ is a better posed optimization problem since we will also have thrust constraints. Therefore, the water mass fraction that will most impact the objective function the most is at the CRZ operating condition and we thus want to size the area of the injector and extractor at CRZ instead of TOC like the rest of the areas are. Normally, the cross-sectional areas are set at the design point (TOC) and are connected to the off-design points. To size the areas of the injector and extractor at CRZ, we only connect the state variables across these components at the corresponding operating condition and set the areas of both components at the CRZ point. These connections are shown graphically in the XDSM diagram in Figure 7.

In the N+3 model implementation presented in the pyCycle paper, the bypass ratio (BPR) at TOC is computed using a Balance component. The Balance component for BPR_{TOC} uses the exit velocity ratio of the core and bypass nozzles which is given as:

$$V_{ratio} = \frac{V_{core,ideal} C_{v,core}}{V_{bypass,ideal} C_{v,bypass}} \tag{3}$$

where V_{ideal} is the exit velocity of a nozzle and C_v is the velocity coefficient of the nozzle which represents non-ideal effects. This adds a constraint where the core exit velocity is higher than the bypass exit velocity.

D. Performance Metrics

Past implementations of engine cycles and example engine cycles in pyCycle generally have used either JetA or JP-7 as a fuel. However, in this work we are interested in comparing the performance of similar engines with different types of fuel, specifically hydrogen due to the many benefits it offers in terms of performance and thermal properties. Fortunately, pyCycle comes ready to use hydrogen with cataloged exhaust products for hydrogen fuel. The fuel type is simply changed from Jet-A to H2 in the pyCycle model.

For measuring the efficiency of jet engines, thrust-specific fuel consumption (TSFC) is generally used since it is a represents how low the fuel burn is for a given level of thrust. However, when comparing Jet-A and hydrogen fuels this is not such a good metric since a given mass flow rate of hydrogen has an energy content almost 3 times that of Jet-A. Therefore, a new metric for comparing the relative efficiencies of engine running on Jet-A versus hydrogen is thrust-specific energy consumption (TSEC) which multiplies TSFC by the lower heating value (LHV) of the fuel

$$TSFC = \frac{\dot{m}_{fuel}}{F_{thrust}} \tag{4}$$

$$TSEC = \frac{\dot{m}_{fuel} LHV}{F_{thrust}} = TSFC \times LHV \tag{5}$$

With all of the sub-models of the N3 engine presented, the complete engine cycle XDSM diagram is shown in Figure 7.

Table 1 N+3 engine model parameters. Atmospheric parameters specify the flight conditions and humidity of the atmosphere and Lower Heating Values are given for each fuel to compute TSEC.

Parameter	Value	Units	Comments
h_{TOC}	0.00017	-	humidity ratio at TOC [8]
h_{CRZ}	0.00017	-	humidity ratio at CRZ [8]
h_{RTO}	0.009	-	humidity ratio at RTO [8]
h_{SLS}	0.009	-	humidity ratio at SLS [8]
LHV_{JetA}	18564.0	BTU/lbm	Lower heating value of Jet-A [9]
LHV_{H2}	51591.0	BTU/lbm	Lower heating value of H2 [10]

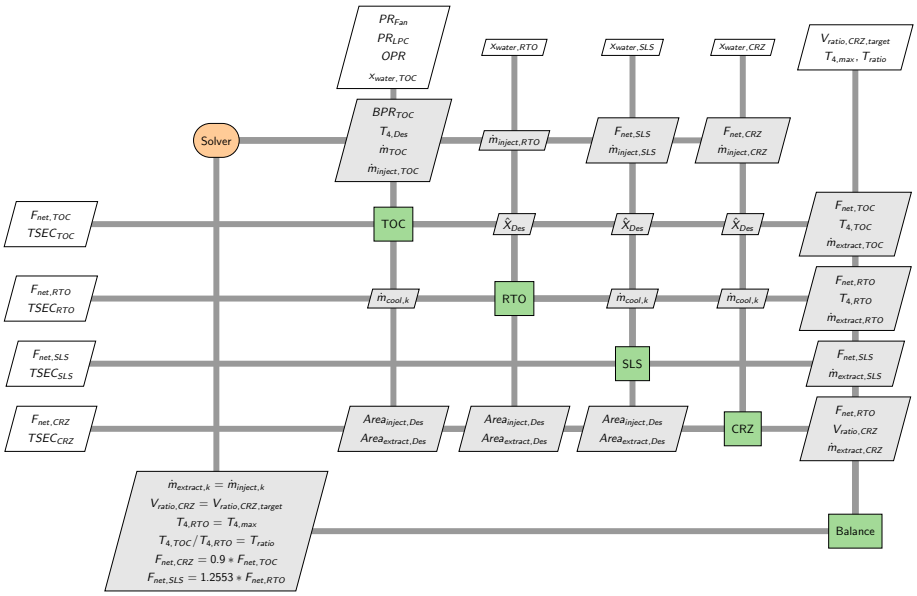

Fig. 7 Full model N+3 XDSM diagram. This XDSM diagram shows the multipoint coupling between the different operation conditions and shows how the water recovery fractions are used to solve for water mass flow rates.

III. Optimization Problem

A. Problem Statment

We will perform gradient-based design optimization of the engine model with the multipoint architecture that considers each of the flight conditions experienced by commercial aircraft. The objective is to minimize the TSEC since that is our performance metric for comparing the two types of fuels. However, since we are constraining net thrust at each operating condition this optimization problem becomes ill-posed. Therefore, we will minimize fuel flow rate, W_{fuel} at the cruise condition subject to net thrust and engine diameter constraints since this will accomplish the same goal with a better posed problem. An XDSM diagram of the optimization problem with the multipoint formulation is shown in Figure 8.

This optimization problem was run for both Jet-A and H2 fuels using the parameters given in Table 1. The optimization problem objective function, design variables, and constraints are shown in Table 2 below.

B. Optimization Software

OpenMDAO supports the use of different optimization packages including Scipy and PyOptSparse. PyOptSparse is an object-oriented framework for formulating and solving nonlinear constrained optimization problems. PyOptSparse was chosen as the main optimization package for this problem due to its wide range of open-source gradient-based optimizers and integration with [11].

SNOPT (Sparse Nonlinear OPTimizer) was the optimizer chosen for this work. SNOPT is a state-of-the-art optimizer used to solve large-scale optimization problems (linear and nonlinear). The software uses a sparse SQP (sequential quadratic programming) algorithm with a limited-memory quasi-Newton approximation to the Hessian of the lagrangian. SNOPT is particularly useful for nonlinear problems whose functions and derivatives are expensive compute. An augmented Lagrangian merit function ensures convergence from an arbitrary point. SNOPT allows the nonlinear constraints to be violated (if necessary) and minimizes the sum of these violations. Due to the robustness of the algorithm, SNOPT is a perfect optimizer to solve the highly complex Multipoint systek since the pyCycle engine cycle would occasionally reach a point that is non-physical and would thus cause the solver to fail. SNOPT is able to respond to these solver failures and backtrack to a region in the design space that is physical and continue the optimization [12].

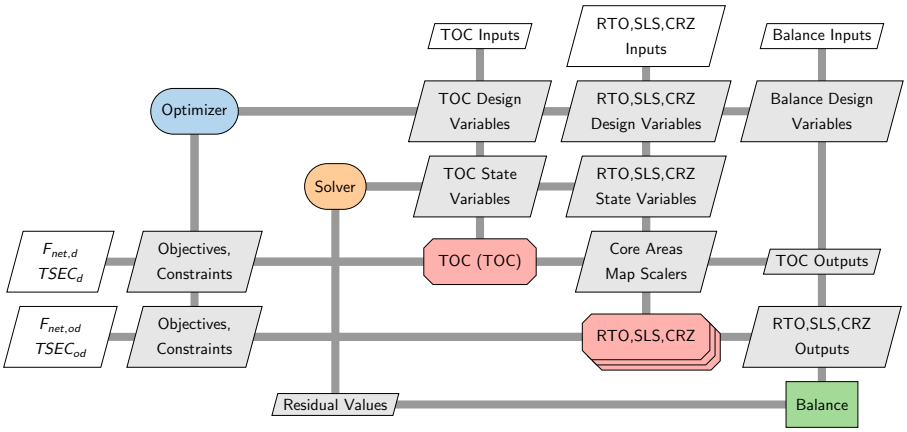

Fig. 8 XDSM diagram of the multipoint optimization problem with constraints. The XDSM diagram shows the variables and outputs within the optimization model and how each of these values is connected to the optimizer and design points.

Table 2 Multipoint optimization problem definition. The 4 design variables are the water recovery fractions of the core exhaust stream at each operation condition. The objective function is the thrust specific energy consumption at the cruise condition with a net thrust constraint at the design point, TOC.

	Variable/Function	Description	Units	Quantity
minimize	$\dot{m}_{\text{fuel,CRZ}}$	Fuel flow rate at CRZ	lbm/s	1
with respect to	$X_{\text{H2O,TOC}}$	Water recovery fraction at TOC	-	1
	$X_{\text{H2O,RTO}}$	Water recovery fraction at RTO	-	1
	$X_{\text{H2O,SLS}}$	Water recovery fraction at SLS	-	1
	$X_{\text{H2O,CRZ}}$	Water recovery fraction at CRZ	-	1
	$\text{PR}_{\text{HPC,TOC}}$	TOC high-pressure compressor pressure ratio	-	1
	$\text{PR}_{\text{LPC,TOC}}$	TOC low-pressure compressor pressure ratio	-	1
	$\text{PR}_{\text{fan,TOC}}$	TOC fan pressure ratio	-	1
	$T_{4,\text{ratio}}$	TOC-to-RTO temperature ratio	-	1
	$T_{4,\text{RTO}}$	RTO combustor temperature	°R	1
	$V_{\text{ratio,CRZ}}$	RTO combustor temperature	°R	1
		Total		10
subject to	$F_{\text{net,TOC}} \geq 5800.0$	Target net thrust at TOC	lbf	1
	$D_{\text{fan}} \leq 100$	Maximum Fan Diameter	inch^2	1
		Total		2

IV. Results

The 4 optimizations mentioned in the optimation problem section were run and solved. The optimality, feasibility, and merit function for each optimization is shown plotted in Figure 9.

From this plot we can visually check that each optimization achieved optimality and feasibility, and that the merit function reached a steady-state value.

The optimization problems with both types of fuel but no water recovery ran smoothly and finished with 0/1 SNOPT exit codes meaning that the optimization finished successfully and met the optimiality conditions. The resulting values of the optimization problem for Jet-A and hydrogen without water recovery are shown in Tables 3 and 4, respectively. Similarly, the resulting values of the optimization problem for Jet-A and hydrogen with water recovery are shown in

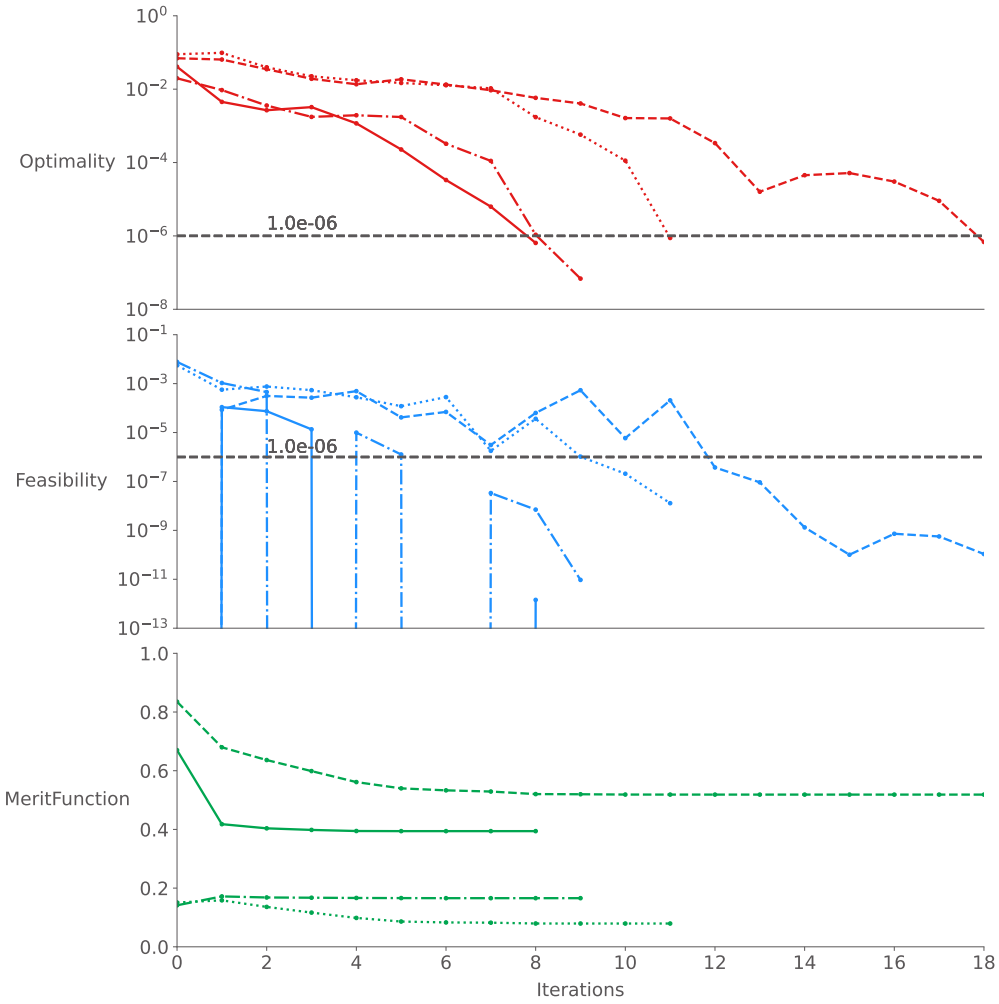

Fig. 9 The optimality, feasibility, and merit fuction for each optimization problem. Jet-A without water recovery is shown with the solid lines, the Jet-A with water recovery is shown with dashed lines, hydrogen without water recovery is shown with dash-dot lines, and hydrogen with water recovery is shown with dotted lines.

Tables 5 and 6, respectively.

Table 3 Optimization results of the N+3 engine with no water recovery using Jet-A as the fuel.

	Variable/Function	Value	Lower	Upper	Units
objective	$W_{fuel,CRZ}$	0.6394	-	-	$\frac{lbm}{s}$
variables	$PR_{fan,TOC}$	1.285	1.2	1.4	-
	$PR_{LPC,TOC}$	4.000	2.5	4.0	-
	$PR_{OPR,TOC}$	59.495	40.0	70.0	-
	$T_{4,TOC}/T_{4,RTO}$	0.913	0.8	0.95	-
	$T_{4,RTO}$	3220.570	-	-	°R
	$V_{ratio,CRZ}$	1.35	1.35	1.45	
constraints	$F_{net,TOC}$	5800.0	5800.0	-	lbf
	D_{Fan}	100.0	100	-	in^2

Table 4 Optimization results of the N+3 engine with no water recovery using hydrogen as the fuel.

	Variable/Function	Value	Lower	Upper	Units
objective	$W_{fuel,CRZ}$	0.2331	-	-	$\frac{lbm}{s}$
variables	$PR_{fan,TOC}$	1.288	1.2	1.4	-
	$PR_{LPC,TOC}$	4.000	2.5	4.0	-
	$PR_{OPR,TOC}$	59.153	40.0	70.0	-
	$T_{4,TOC}/T_{4,RTO}$	0.918	0.8	0.95	-
	$T_{4,RTO}$	3204.07	-	-	°R
	$V_{ratio,CRZ}$	1.35	1.35	1.45	
constraints	$F_{net,TOC}$	5800.0	5800.0	-	lbf
	D_{Fan}	100.0	100	-	in^2

Table 5 Optimization results of the N+3 engine with water recovery using Jet-A as the fuel.

	Variable/Function	Value	Lower	Upper	Units
objective	$W_{fuel,CRZ}$	0.6037	-	-	$\frac{lbm}{s}$
variables	$x_{H2O,TOC}$	0.115	0.0	-	-
	$x_{H2O,RTO}$	0.0	0.0	-	-
	$x_{H2O,SLS}$	0.0	0.0	-	-
	$x_{H2O,CRZ}$	0.3	0.0	0.3	-
	$PR_{fan,TOC}$	1.291	1.2	1.4	-
	$PR_{LPC,TOC}$	4.0	2.5	4.0	-
	$PR_{OPR,TOC}$	54.218	40.0	70.0	-
	$T_{4,TOC}/T_{4,RTO}$	0.923	0.8	0.95	-
	$T_{4,RTO}$	3400.0	-	-	°R
	$V_{ratio,CRZ}$	1.35	1.35	1.45	
constraints	$F_{net,TOC}$	5800.0	5800.0	-	lbf
	D_{Fan}	100.0	100	-	in^2

Table 6 Optimization results of the N+3 engine with water recovery using hydrogen as the fuel.

	Variable/Function	Value	Lower	Upper	Units
objective	$W_{fuel,CRZ}$	-	-	-	$\frac{lbm}{s}$
variables	$x_{H2O,TOC}$	0.066	0.0	-	-
	$x_{H2O,RTO}$	0.0	0.0	-	-
	$x_{H2O,SLS}$	0.0	0.0	-	-
	$x_{H2O,CRZ}$	0.17	0.0	0.17	-
	$PR_{fan,TOC}$	1.291	1.2	1.4	-
	$PR_{LPC,TOC}$	4.0	2.5	4.0	-
	$PR_{OPR,TOC}$	55.132	40.0	70.0	-
	$T_{4,TOC}/T_{4,RTO}$	0.927	0.8	0.95	-
	$T_{4,RTO}$	3204.07	-	-	°R
	$V_{ratio,CRZ}$	1.35	1.35	1.45	
constraints	$F_{net,TOC}$	5800.0	5800.0	-	lbf
	D_{Fan}	100.0	100	-	in^2

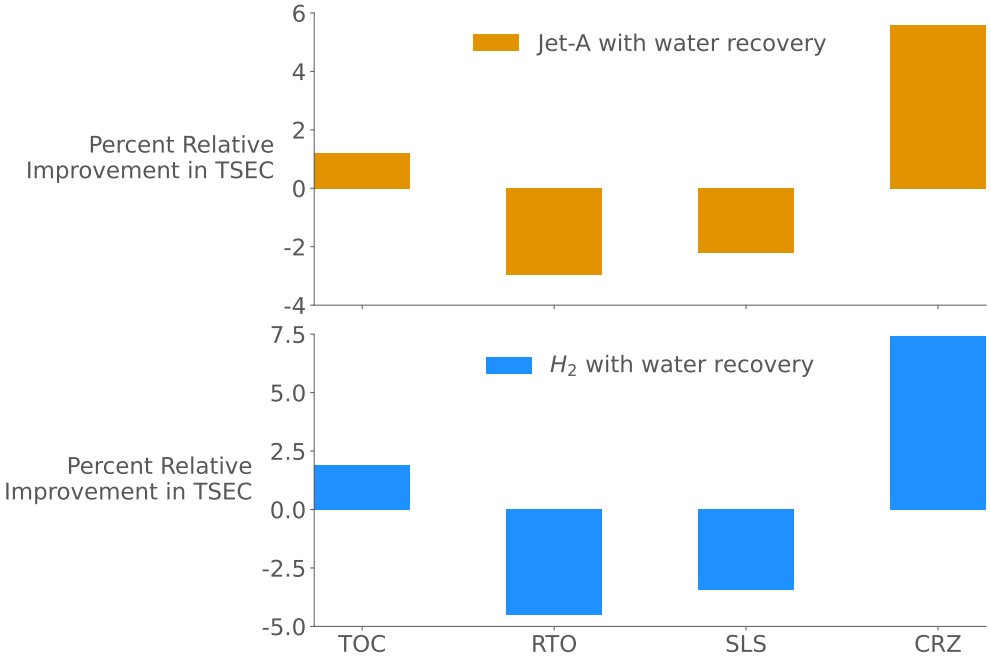

Fig. 10 Percent relative improvement in thrust specific energy consumption (TSEC) of the N+3 engine with water recovery. This plot compares the relative improvement of the two optimziation problems with water recovery compared to the optimzation problems without water recovery.

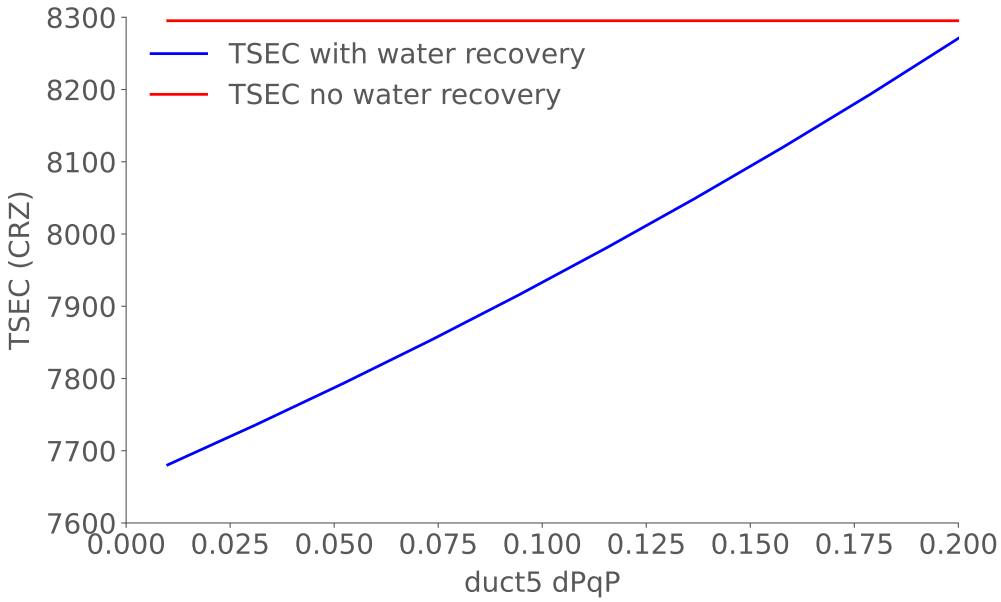

Fig. 11 Pressure loss sweep of Duct5 which is just upstream of the water vapor extractor component. The figure shows the optimized TSEC values for varying levels of pressure loss due to the condensation of water compared to the optimzation problem without water recovery. This shows the working space available for designing a water condensor while still gaining the benefits of water recovery.

V. Conclusion

VI. Acknowledgements

References

[1] Daggett, D. L., Fucke, L., Hendricks, R. C., and Eames, D. J., "Water Injection on Commercial Aircraft to Reduce Airport Nitrogen Oxides," Tech. rep., NASA, March 2010.

[2] Gordon, S., and McBride, B. J., "Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations," *NASA Rept. RP-1311*, 1994.

[3] Jones, S., *An Introduction to Thermodynamic Performance Analysis of Aircraft Gas Turbine Engine Cycles Using the Numerical Propulsion System Simulation Code*, NASA, 2007. TM-2007-214690.

[4] Hendricks, E. S., and Gray, J. S., "pyCycle: A Tool for Efficient Optimization of Gas Turbine Engine Cycles," *Aerospace*, Vol. 6, No. 87, 2019. doi:10.3390/aerospace6080087.

[5] Gray, J. S., Chin, J., Hearn, T., Hendricks, E., Lavelle, T., and Martins, J. R. R. A., "Chemical Equilibrium Analysis with Adjoint Derivatives for Propulsion Cycle Analysis," *Journal of Propulsion and Power*, Vol. 33, No. 5, 2017, pp. 1041–1052. doi:10.2514/1.B36215.

[6] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. doi:10.1007/s00158-019-02211-z.

[7] Jones, S. M., Haller, W. J., and Tong, M. T., "An N+3 Technology Level Reference Propulsion System," Tech. Rep. NASA/TM—2017-219501, NASA Glenn Research Center, 2017. URL `https://ntrs.nasa.gov/citations/20170005426`.

[8] Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., Zhu, Y., Chelliah, M., Ebisuzaki, W., Higgins, W., Janowiak, J., Mo, K. C., Ropelewski, C., Wang, J., Leetmaa, A., Reynolds, R., Jenne, R., , and Joseph, D., "The NCEP/NCAR 40-Year Reanalysis Project." *Bulletin of the American Meteorological Society*, Vol. 77, 1996, pp. 437–471.

[9] "Jet Fuel Characteristics," , 2002. URL `https://www.smartcockpit.com/docs/Jet_Fuel_Characteristics.pdf`.

[10] "Fuels - higher and lower calorific values," , 2003. URL `https://www.engineeringtoolbox.com/fuels-higher-calorific-values-d_169.html`.

[11] Wu, N., Kenway, G., Mader, C. A., Jasa, J., and Martins, J. R. R. A., "pyOptSparse: A Python framework for large-scale constrained nonlinear optimization of sparse systems," *Journal of Open Source Software*, Vol. 5, No. 54, 2020, p. 2564. doi:10.21105/joss.02564.

[12] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, 2005, pp. 99–131. doi:10.1137/S0036144504446096.