DNA - An integrated open-source optimization platform for thermo-fluid systems

Pierobon, Leonardo; Wronski, Jorrit; Elmegaard, Brian; Haglind, Fredrik; Bell, Ian H.

Published in:
Proceedings - The 55th Conference on Simulation and Modelling (SIMS 55)

Publication date:
2014

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
DNA - AN INTEGRATED OPEN-SOURCE OPTIMIZATION PLATFORM FOR THERMO-FLUID SYSTEMS

Leonardo Pierobon, Jorrit Wronski, Brian Elmegaard and Fredrik Haglind
Technical University of Denmark
Department of Mechanical Engineering
2800 Kongens Lyngby
Denmark

Ian H. Bell
University of Liège
Thermodynamics Laboratory
4000 Liège
Belgium

ABSTRACT
This paper presents developments and new features added to the simulation tool Dynamic Network Analysis. This open-source software is the result of ongoing development at the Department of Mechanical Engineering, Technical University of Denmark since 1988. Ever since, it has been employed to model dynamic and steady-state energy systems and is now available for the most common operating systems (Windows, Mac OS and Linux). Emerging interest in novel plant technologies, high-temperature heat pumps, refrigeration absorption modules, and in energy system optimization has stressed the necessity to extend the capabilities of the software, while at the same time decreasing computational time. Dynamic Network Analysis can now solve non-convex optimization problems by virtue of the fully-embedded genetic algorithm. Moreover, the thermophysical fluid property library has been extended with more than 110 fluids by interfacing CoolProp, a high-accuracy open-source property package for pure and pseudo-pure fluids, as well as humid air. The new features are unveiled in one case study where the optimization of an air bottoming cycle unit recuperating the exhaust heat from an offshore power system is performed by taking advantage of CoolProp’s table-based property interpolation scheme.

Keywords: DNA, CoolProp, Evolutionary algorithm, Thermophysical properties

NOMENCLATURE
\(a\) Constants in Eq. 2 or air
\(F_h\) Fin height [mm]
\(F_l\) Fin length [mm]
\(h\) Specific enthalpy [kJ·kg\(^{-1}\)]
\(l_{exh}\) Length gas side [m]
\(p\) Pressure [bar]
\(T\) Temperature [K]
\(v\) Specific volume [m\(^3\)·kg\(^{-1}\)]
\(\bar{X},\bar{Y}\) Arrays of optimization variables
\(x_y\) Tabulated and normalized fluid properties
\(\Delta\) Difference
\(\rho\) Density [kg·m\(^{-3}\)]

INTRODUCTION
Simulation of thermodynamic systems has become one of the main pillars of engineering work, university education as well as public and private research. Many software products are deeply integrated into existing workflows and increasing simulation quality has led to decreasing experimental efforts while accelerating the pace of implementation and development of new and optimised thermodynamic pro-
cesses. Such processes involve a number of non-linear phenomena, e.g. the phase change of a working fluid or transitions between laminar and turbulent flow regimes, calling for robust solution strategies and integrators.

Software for modelling thermo-fluid systems typically involves an ecosystem of at least three interacting software components. First, a system interpreter (a) converts the users’ input and creates a modified system of equations along with bindings to external libraries. These equations are then processed by a numerics component (b) containing solvers and integrators, which typically consist of custom-made special-purpose algorithms and modified versions of popular mathematics libraries like the Basic Linear Algebra Subprograms (BLAS) [1], its high-level companion the Linear Algebra Package (LAPACK) [2] and the Harwell Subroutine Library (HSL) [3]. The third component is the fluid property package (c) whose calculations are the bottleneck of the entire solving procedure while discontinuities in the therophysical properties and their derivatives challenge the solvers’ stability. For dynamic simulations, a fourth functional (d) unit can be introduced to provide access to data for varying operating conditions including controller inputs. Common software packages in the field of thermo-fluid simulations implement one or more of the four aspects, (a) through (d), mentioned above. If needed, missing features are often supplemented by accessing an independent library.

There are a number of software for steady-state simulation. Many customised solutions are based on the general purpose computing packages SciPy [4] and Matlab [5] exploiting their powerful solvers and data treatment facilities. A popular equation-based application for static modelling is the Engineering Equation Solver (EES) [6]. Another package for similar purposes is CycleTempo [7], which also comes with its own fluid property database, representing the component-based modelling environments. Dynamic problems formulated in systems of differential equations require other approaches such as the ones implemented in Ascend [8], AspenONE [9], Dynamic Network Analysis (DNA) [10] and the different implementations of the Modelica language, e.g. Dymola [11] and OpenModelica [12].

Fluid property calculations are usually delegated to external libraries that are accessed on a per-call basis extending a small group of built-in fluid property correlations. A commonly used library for this purpose is REFPROP [13] developed by the U.S. National Institute of Standards and Technology (NIST). It can handle pure and pseudo-pure fluids as well as a large amount of mixtures of common working fluids. Another commercial software for a similar purpose is FluidProp [14], the property calculation engine behind CycleTempo. Furthermore, the TIL Media Suite [15] provides access to optimised routines tailored for dynamic simulations.

The only freely-available high-accuracy property library for mixtures, known to the authors is TREND [16]. Both, TREND and CoolProp [17] are competitive free fluid property libraries for pure and pseudo-pure fluids since they contain a large selection of relevant working fluids with CoolProp offering the unique features of tabulated property interpolation. The other open-source alternatives FPROPS [18] from the Ascend project and the Modelica-based packages ModelicaMedia [19] and HelmholtzMedia [20] only have a limited number of working fluids. Currently, DNA includes its own routines for a number of fluids like water/steam by Wagner and Pruss [21] with LiBr-Water solutions added by Pátek and Klomfar [22], some real gases and mixtures of ideal gases [23], refrigerants [24]. It also contains functions for calculating properties of solid fuels such as biomass and coal. In particular, functions for thermal radiation properties of combustion products and exergy may be mentioned as uncommon features.

Looking at the above, integrating DNA with CoolProp combines a mature steady-state solver and integrator with a computationally efficient fluid property database. Due to the enhanced simulation speed, DNA can now be used to perform optimisation and control development tasks as described below.

**DYNAMIC NETWORK ANALYSIS**

The present section discloses new features and capabilities of the simulation tool Dynamic Network Analysis. Particular emphasis is dedicated to the newly implemented capabilities of solving multi-variable optimization problems and of utilizing advanced techniques for rapid computation of therophysical and transport properties for a large variety of pure fluids.
New features

Dynamic Network Analysis is the present result of ongoing development at the Department of Mechanical Engineering, Technical University of Denmark, which began with a Master’s Thesis work in 1988 [25]. DNA may be described as a modelling language for describing thermo-fluid systems and as the compiler that interprets this language and makes it accessible for the numerical solver. It can also provide some fluid property data and therefore covers the aforementioned components (a) and (c), while partly fulfilling tasks done by the numerics component (b). In DNA, the physical model is formulated analogously to electrical networks by connecting the relevant component models through nodes and by including operating conditions for the complete system. The physical model is converted into a set of equations to be solved numerically. The mathematical problem can include both dynamic and steady-state mass and energy conservation balances for all components and nodes, as well as relations for thermodynamic properties of the fluids involved. The program includes a component library with models for a large number of different components existing within energy systems.

DNA is an open source tool written in FORTRAN. It is integrated with the open source editor Emacs and the open source compiler suite GCC both distributed by the Free software Foundation [26].

DNA solves both steady state and dynamic problems. For steady state problems algebraic loops due to mass balances in thermodynamic cycles are detected and removed by the compiler. The solver is a modified Newton method.

Dynamics occur in mass and energy balances as well as in control algorithms and in inertia of rotating shafts. This results in differential-algebraic equations which are solved by a backward differentiation formulae (BDF) implemented for variable step size by use of the Nordsieck formulation. The solver handles discontinuities by restarting at the point of crossing.

The software capabilities are improved by linking DNA with the open-source genetic algorithm implemented by Carroll [27]. The code, readily available in Fortran language, is compiled together with DNA to form a unique software capable of solving non-convex optimization problems, to be set directly through the user interface. The genetic algorithm is preferred to gradient-based methods since it is less prone to end its search in local minima of the problem, usually converging towards global optima. This comes at the cost of an increased computational effort, due to the large number of evaluations of the objective functions [28]. For a more in-depth description of the key features of the genetic algorithm employed in this work the reader can refer to Goldberg and Holland [29].

Additionally, the fluid library is extended with over 110 compressible fluids given as Helmholtz-free-energy-based equations of state (EOS) and over 50 incompressible liquids and solutions by using a custom-built C-interface to pass property calls to the open-source thermophysical property library CoolProp [17]. Furthermore, an integration with CoolProp also enables the usage of all fluids and mixtures available in REFPROP [13].

Matlab interface

Matlab is an important tool in many technical applications. It is used in control applications, often in combination with the Simulink tool. DNA has been demonstrated integrated with Matlab as a function that Matlab can call when needed to obtain data from the plant that the DNA model is emulating. This means that the complete DNA application more or less will be considered as a Simulink block with inputs and outputs. The inputs are control signals and the outputs are measurements one sampling time ahead, when DNA has calculated the new state. Matlab then may change to the control signals and run DNA again. The integration is implemented via the Matlab external interface (Mex) [30, 31]. One feature which has been wanted for the development of DNA as a Mex routine has been that it should run on both Microsoft Windows and Linux. For this reason GCC is used as compiler [32].

The algorithm used is:

Call Mexdna from Matlab with the DNA input file as argument. The call is done via a Mex gateway file.

Read the input file and initialize the DNA model

Return to Matlab

while Simulating do

Change boundary conditions for control signals, load requirements and disturbances for the model.
Call Mexdna with sampling time step as input
Simulate until sampling time is reached
Return measurements (simulation results) to Matlab
end while

To pass variables between Matlab and the external program, a gateway routine is required. This routine handles conversion between Matlab array pointers and the external types. In order to keep all variables in memory while the execution control is in Matlab, the persistent arrays feature is used.

Fluid properties

Nowadays, many engineering applications make heavy use of advanced mathematics to solve design and control problems. Especially iterative methods and differential equations involve numerous calls to the connected fluid property library. Hence, computational efficiency is of the utmost importance.

Helmholtz-free-energy-based equations of state (EOS) provide the most accurate source of thermodynamic data for many relevant working fluids [33]. However, evaluating state points to provide properties based on random inputs is a computationally expensive task. The EOS is usually provided in a temperature-density-explicit form and those two are not very common iteration variables for a simulation software. Interpolation methods can improve the speed of evaluation drastically. The described software uses two different interpolation methods: a) Tabular Taylor Series Expansion (TTSE) as described by Miyagawa and Hill [34] and Watanabe and Dooley [35] and b) bicubic interpolation published in the work of Keys [36]. To work efficiently, both methods require a regularly spaced grid of state points and a couple of derivatives with respect to the two independent variables.

The following examples illustrate the procedures employed to obtain the property values using the interpolation methods. Using TTSE, the temperature can be obtained from the often used inputs pressure and enthalpy with the expansion

\[
T = \Delta h \left( \frac{\partial T}{\partial h} \right)_p + \frac{\Delta h^2}{2} \left( \frac{\partial^2 T}{\partial h^2} \right)_p + \Delta p \left( \frac{\partial T}{\partial p} \right)_h + \frac{\Delta p^2}{2} \left( \frac{\partial^2 T}{\partial p^2} \right)_h + \Delta h \Delta p \left( \frac{\partial^2 T}{\partial p \partial h} \right) + T_{i,j}
\]

where the derivatives are evaluated at the point \(i, j\), and the differences are given by \(\Delta p = p - p_j\) and \(\Delta h = h - h_i\). The nearest state point can be found directly due to the regular spacing of the grid of points. For an improved representation of the \(p-v-T\) surface, bicubic interpolation can be used. In the bicubic interpolation method, the state variable and its derivatives are known at each grid point. This information is used to generate a bicubic representation for the property in the cell, which could be expressed as

\[
T(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij}x^iy^j
\]

where \(a_{ij}\) are constants based on the cell boundary values and \(x\) and \(y\) are normalized values for the enthalpy and pressure, for instance. The constants \(a_{ij}\) in each cell are cached for additional computational speed.

The increase in computational speed depends on the complexity of the EOS. Using IAPWS 1995 [21] as an example for the properties of subcooled water at 10 MPa, both methods yield an acceleration by a factor of more than 120 for one of the most involved equations of state when calculating density from \(p\) and \(h\) based on the current implementation in CoolProp. The measured computational time of approximately 1 µs per \(\rho(p,h)\)-call is in good agreement with figures reported by Johansen [37] for the same fluid, which also provides an in-depth comparison of numerous interpolation schemes.

As another example of the accuracy of these interpolation methods, the density of air is evaluated at 40000 random data points covering the entire fluid surface. Figure 1 presents the results of this analysis. These data show that the accuracy of the bicubic interpolation method is generally several orders of magnitude better than that of the TTSE method, though both yield acceptable accuracy for most technical needs.
CASE STUDY

The work considers as exemplary application the power plant installed on the Draugen oil and gas offshore platform, located 150 km from Kristiansund, in the Norwegian Sea. The reservoir was discovered in 1984 and started operation in 1993. The platform, operated by A/S Norske Shell, produces gas exported via Åsgard gas pipeline to Kårstø (Norway) and oil, which is first stored in tanks at the bottom of the sea and then exported via a shuttle tanker (once every 1-2 weeks). The normal power demand is around 19 MW and it can increase up to 25 MW during oil export. To enhance the reliability and to diminish the risk of failure of the power system, two turbines run at a time covering 50% of the load each, while the third one is kept on standby, allowing for maintenance work. Despite the low performance, this strategy ensures the necessary reserve power for peak loads, and the safe operation of the engines.

The system layout

Figure 2 shows the layout of the power system with the additional air bottoming cycle (ABC) module which recuperates the exhaust heat produced by gas turbine A. The engines B and C are not reported. Note that the bottoming cycle unit should have the capability to harvest the waste heat alternatively from the other two engines, thus ensuring high performances when switching the gas turbines on operation. The twin-spool engine employs two coaxial shafts coupling the low pressure compressor (LPC) with the low pressure turbine (LPT) and the high pressure compressor (HPC) with the high pressure turbine (HPT). The power turbine (PT) transfers mechanical power through a dedicated shaft to the electric generator (GEN). Natural gas is the fuel utilized in the combustion chamber (CC).

As regarding the ABC part, the first compressor (AC1) intakes ambient (1 $\rightarrow$ 2) air which then cools down in the intercooler (2 $\rightarrow$ 3), in this way diminishing the specific work during the compression process. The second compressor (AC2) increases further the pressure of the working fluid (3 $\rightarrow$ 4) which then harvests the exhaust energy from the gas turbine in the finned-plate heat exchanger (4 $\rightarrow$ 5). As reported in Kays and London [38], this device offers higher performances and compactness for gas-to-gas heat transfer processes compared to shell-and-tube and flat-plate heat exchangers. The air then expands through the air turbine AT2 (5 $\rightarrow$ 6) which drives
the second compressor, and afterwards through the air turbine AT1 (6 → 7) which is mechanically connected to AC1. Finally, the power air turbine (PAT) produces useful power by driving the electric generator (7 → 8).

**Heat transfer equipment**

The finned-plate heat exchanger serving the ABC power system consists of a stack of plates where the hot and cold fluids flow in the free space between the plates, typically in a cross flow arrangement, see Figure 4(a). The plates are equipped with a number of fins with the purpose of augmenting the surface area and of attaining larger heat transfer area-to-volume ratios, i.e. high compactness. The fins may have different shapes, e.g. wavy fins, offset fins and offset strip fins. The latter configuration (see Figure 4(b)) is the most widely adopted and it is thus the one selected in the current work. It is worth mentioning that FPHEs have to operate at lower temperatures (840 °C) and pressures (8.3 bar) [39] compared to shell-and-tube heat exchangers, depending on the process utilized to bond the metal plates. Given the boundary conditions for the ABC unit (see Table 2), these limitations are fulfilled in the present analysis.

The design approach to size the heat transfer equipment is the *effectiveness – NTU method* implemented as reported in Yousefi et al. [40]. The heat transfer coefficient and the pressure drop on each side of the finned-plate heat exchanger (see Figures 4(a) and 4(b)) are calculated in accordance to Manglik and Bergles [41]. The design model of the FPHE was verified by comparison with the geometrical data reported in Yousefi et al. [40]. The differences between the model results and the values reported in the references are within 4.0% in terms of both overall heat transfer coefficient and pressure drop.

**The optimization problem**

A feasible implementation of bottoming cycle units offshore requires to design compact and light modules, while at the same time exploiting the entire waste heat recovery potential. For this case study, the optimization problem is formulated in two main steps: i) find the cycle parameters maximizing the net power output of the ABC unit, and ii) define the geometry of the finned-plate heat exchanger which minimizes its weight given the optimal cycle variables.

The genetic algorithm carries out the two tasks by acquiring the array of the parameters and the upper and lower bounds limiting the possible values for the vectors of the optimization variables $\hat{X}$ and $\hat{Y}$, which read
Table 1: Lower and upper bound for the variables involved in the optimization of the air bottoming cycle unit depicted in Figure 2.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Lower - upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outlet pressure $p_2/p_1$</td>
<td>-1.2 - 3.2</td>
</tr>
<tr>
<td>Outlet pressure $p_4/p_3$</td>
<td>-1.2 - 3.2</td>
</tr>
<tr>
<td>Fin height [mm]</td>
<td>50 - 200</td>
</tr>
<tr>
<td>Fin length [mm]</td>
<td>20 - 100</td>
</tr>
<tr>
<td>Gas side length [m]</td>
<td>1.0 - 3.0</td>
</tr>
</tbody>
</table>

\[ \bar{X} = \left[ \frac{p_2}{p_1}, \frac{p_4}{p_3} \right] \quad \text{and} \quad (3) \]

\[ \bar{Y} = \left[ F_{h,a}, F_{l,a}, F_{h,exh}, F_{l,exh}, l_{exh} \right]. \quad (4) \]

In Equation 3, $p_2/p_1$ and $p_4/p_3$ are the pressure ratios of the two air compressors, see Figure 2. The variables $F_{h}$, $F_{l}$ and $l_{exh}$ are the fin height, the fin length and the length of the exhaust gas side. The subscripts "a" and "exh" refer to the air and the exhaust stream side.

The objective functions for steps i) and ii) are the electric power output of the generator serving the ABC unit and the weight of the finned-plate heat exchanger calculated from the design analysis. Table 1 lists the upper and lower bounds of the optimization variables. Note that the values related to the geometry of the heat exchanger are set accordingly with the figures reported in Shah and Sekulić [39] and Yousefi et al. [40].

Table 2 lists the parameters which are maintained constant during the two successive optimizations. As the use of integer variables in the optimization routine is beyond the capabilities of the algorithm, the fin per meter and the number of plates are kept constant. The thickness of the finned-plate heat exchanger and the construction material (stainless steel) are retrieved from Bolland et al. [42].

As the gas turbine can operate on a wide range of both liquid and gaseous fuels, the terminal temperature of the exhaust gases exiting the FPHE is set to 160 °C. This precaution prevents condensation of corrosive compounds when fuels such as crude oil, heavy fuel oil and naphtha are combusted. In the geometric optimization of the finned-plate heat exchanger it is verified that the velocity on the hot and cold side of the heat exchangers is within 1 m·s⁻¹ and 40 m·s⁻¹, while an upper limit of 3 kPa is selected for the pressure drops. If such test is not passed the objective function is set to a large number, i.e. $10^6$.

The parameters of the genetic algorithm are specified according to De Jong and Spears [43]: generation size 1000, crossover rate 0.6, and mutation rate 0.001. As for the population number in the first optimization and in the weight minimization, the specified values are 5 and 50, respectively. These numerical quantities are selected so as to ensure the repeatability of the solution when different simulations are performed. The process terminates when the average change in the objective function of two successive iterations is lower than the specified tolerance ($10^{-4}$) or when the number of generation reaches the maximum value.

RESULTS

Table 2: Parameters assumed for the two-step optimization.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generator efficiency [%]</td>
<td>98</td>
</tr>
<tr>
<td>Ambient temperature [°C]</td>
<td>15</td>
</tr>
<tr>
<td>Ambient pressure [bar]</td>
<td>1.013</td>
</tr>
<tr>
<td><strong>Gas turbine</strong></td>
<td></td>
</tr>
<tr>
<td>Electric power output [MW]</td>
<td>16.524</td>
</tr>
<tr>
<td>Thermal efficiency [%]</td>
<td>31.3</td>
</tr>
<tr>
<td>Exhaust gas temperature [°C]</td>
<td>379.2</td>
</tr>
<tr>
<td>Exhaust gas mass flow [kg · s⁻¹]</td>
<td>91.5</td>
</tr>
<tr>
<td><strong>Air bottoming cycle unit</strong></td>
<td></td>
</tr>
<tr>
<td>Water inlet temperature [°C]</td>
<td>5</td>
</tr>
<tr>
<td>Water outlet temperature [°C]</td>
<td>40</td>
</tr>
<tr>
<td>Pressure drops air side [bar]</td>
<td>0.03</td>
</tr>
<tr>
<td>Outlet temperature $t_5$ [°C]</td>
<td>330</td>
</tr>
<tr>
<td>Exhaust temperature $t_{11}$ [°C]</td>
<td>160</td>
</tr>
<tr>
<td>Compressor mechanic efficiency [%]</td>
<td>99.8</td>
</tr>
<tr>
<td>Compressor isentropic efficiency [%]</td>
<td>86</td>
</tr>
<tr>
<td>Turbine isentropic efficiency [%]</td>
<td>90</td>
</tr>
<tr>
<td><strong>Finned-plate heat exchanger</strong></td>
<td></td>
</tr>
<tr>
<td>Fins per meter air side [m⁻¹]</td>
<td>500</td>
</tr>
<tr>
<td>Fins per meter gas side [m⁻¹]</td>
<td>500</td>
</tr>
<tr>
<td>Number of plates [-]</td>
<td>60</td>
</tr>
<tr>
<td>Fin thickness [µm]</td>
<td>200</td>
</tr>
<tr>
<td>Plate thickness [µm]</td>
<td>200</td>
</tr>
</tbody>
</table>

Figure 5 shows the optimal value of the objective function, i.e. the net power output of the ABC mod-
ule, at each generation for the optimization of the thermodynamic cycle. The electric power remains between 2730 kW and 2750 kW for the first 180 generations. The largest improvement is attained at generation 179 where the objective function reaches a value of 2756 kW. Considering the first generation as a reference, the total gain in power across the optimization process is around 30 kW which corresponds to a relative improvement of 1.1%.

Figure 6 reports the value of the optimization variables, i.e. the pressure ratios of the two air compressors, as a function of the generation number. Accordingly with the trend of the objective function, the pressure ratios reach the best values, i.e. 1.685 and 2.543, after 340 generations. The final quantities differ from the ones of the first generation by 8.1% and 3.7%, respectively.

Figure 7 reports the evolutionary trend of the weight of the finned-plate heat exchanger for the second optimization. It is underlined that the pressure ratios of the compressors are now fixed to the optimal values. It can be noted that the algorithm employs all the available generations to abate the value of the objective function. The largest improvements occur during the first ten generations where the target parameter decreases from 40 $10^3 \cdot \text{kg}$ to 35 $10^3 \cdot \text{kg}$.

Figure 8 shows the evolutionary trend of one optimization variable, i.e. the fin height on the exhaust gas side, involved in the routine. The largest variations occur during the very first evolutions where the fin height of the exhaust gas side diminishes from 75.5 to 84.7 mm. The last variation ($\approx 1.0 \text{ mm}$) occurs at iteration number 846.

Table 3 lists the results of the two-steps optimization of the ABC power system. The thermal efficiency of the ABC unit (12.6%) is calculated as the ratio between the electric power output and the heat rate exchanged by the FPHE.

**DISCUSSION**

As for the optimization of the ABC system, the power output and the thermal efficiency of the plant illustrated in Figure 2 increase from 16.5 MW to 19.3 MW and from 31.3% to 36.5%, respectively. Hence, for the thermal efficiency an increment of 5.2%-points is attained. The improvement appears to be smaller than the values reported in the open lit-
Table 3: Results of the two-step optimization. Air bottoming cycle unit and finned-plate heat exchanger design.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Air bottoming cycle</strong></td>
<td></td>
</tr>
<tr>
<td>Pressure ratio $p_2/p_1 [-]$</td>
<td>1.685</td>
</tr>
<tr>
<td>Pressure ratio $p_4/p_3 [-]$</td>
<td>2.543</td>
</tr>
<tr>
<td>Net power output [MW]</td>
<td>2.76</td>
</tr>
<tr>
<td>Thermal efficiency [%]</td>
<td>12.6</td>
</tr>
<tr>
<td><strong>Finned-plate heat exchanger</strong></td>
<td></td>
</tr>
<tr>
<td>Fin height air side [mm]</td>
<td>80.5</td>
</tr>
<tr>
<td>Fin length air side [mm]</td>
<td>30.8</td>
</tr>
<tr>
<td>Fin height gas side [mm]</td>
<td>87.3</td>
</tr>
<tr>
<td>Fin length gas side [mm]</td>
<td>66.5</td>
</tr>
<tr>
<td>Gas side length [m]</td>
<td>1.43</td>
</tr>
<tr>
<td>Effectiveness [%]</td>
<td>86.9</td>
</tr>
<tr>
<td>Heat transfer area [m²]</td>
<td>20463</td>
</tr>
<tr>
<td>UA-value [kW · m² · K⁻¹]</td>
<td>90.3</td>
</tr>
<tr>
<td>Core volume [m³]</td>
<td>42.0</td>
</tr>
<tr>
<td>Total weight [10³ · kg]</td>
<td>33.8</td>
</tr>
</tbody>
</table>

Furthermore, this work assumes conservative values for the temperature of the exhaust gases and at the inlet of the first air turbine so as to account for a reasonable size of the primary heat exchanger and to preserve the fuel flexibility of the topping unit. Operating gas turbines at lower temperatures effectively reduces corrosion caused by impurities in the fuel as well as the deposition of ashes [46].

As for the heat exchanger calculations, the values listed in Table 3 are in accordance, considering the different size of the plants, with the values obtained by Bolland et al. [42] for the design of a finned-plate heat exchanger of an ABC module recuperating the exhaust heat from the LM2500 gas turbine.

As reported in the previous section, the genetic algorithm computes the optimal pressure ratios of the two air compressors in 345 generations, given the boundary conditions listed in Table 2. Figure 9 shows a contour plot which highlights the impact of the two optimization variables on the net power output of the air bottoming cycle unit. The dark-red zones are characterized by low net power outputs, while the white ones define the regions of high values for the target function. The outcomes of the first optimization step are in-line with the results of such sensitivity analysis since the optimal pressure ratios lay in the white area of Figure 9.

To be noted that the first optimization handles a convex quadratic function with positive definite Hessian, see Figure 9. Under these hypothesis conditions.
gradient-based approaches such as the steepest descent method are ensured to detect the exact solution with high rates of convergence and with a fixed number of iterations. These properties are expected to be poorer for the genetic algorithm due to the stochastic approach and depending on the input parameters, e.g. populations size, mutation rate.

The benefits of evolutionary approaches are instead fully exploited when minimizing the weight of the finned-plate heat exchanger. In fact, the second optimization process has to deal with numerous discontinuities in the objective function, which arise when the solver hits the limitations of the velocity and of the pressure drops in the FPHE. Figure 10 shows the effect of variations in the height and length of the fins installed on the air side. The figures for the remaining three variables are kept at the optimal values listed in Table 3.

In this case the white zones indicate regions where infeasible designs are attained as the pressure drops or fluid velocities exceed the predefined limits. Since in these cases the objective function is set to an arbitrary value ($10^6$), the bounds create discontinuities which can be promptly handled by the genetic algorithm. Moreover, stochastic methods are in this case preferable by virtue of the non-convexity of the target function, making gradient-based methods largely dependent on the starting point and more prone to convergence issues.

**OUTLOOK AND CONCLUSIONS**

**Fluid Properties**

In order to employ the interpolation techniques discussed above, tabulated data has to be generated from the EOS. The current implementation uses gridded data sets in terms of enthalpy $h$ and logarithmic pressure $\log p$ and logarithmic specific volume $\log v$ and temperature $T$, respectively. The minimum enthalpy $h_{\text{min}}$ is the enthalpy of the saturated liquid state at the minimum temperature defined by the employed EOS $T_{\text{min}}$. The enthalpy difference between saturated liquid and saturated vapour at $T_{\text{min}}$ is added twice to $h_{\text{min}}$ yielding $h_{\text{max}}$. The minimum pressure $p_{\text{min}}$ is defined by the saturated vapour state at $T_{\text{min}}$. $p_{\text{max}}$ is obtained from multiplying the reducing pressure of the EOS (often the critical pressure) with two. $T_{\text{max}}$ is defined in the same way, while minimum and maximum specific are extracted from the previously generated tables in $h$ and $\log p$. The regularly spaced grids are shown in Figure 11.

It is obvious from Figure 11, that the two ranges of tabulated data do not cover the same regions. Therefore, additional calculation steps are required to assure that the calculation stays within the bounds of both data sets.

Due to the high accuracy of the interpolation methods shown in Figure 1, the next version of CoolProp will cover the whole $T$, $p$ and $h$, $p$ space for an EOS with the same amount of data points. Maximum enthalpy is obtained from $T_{\text{max}}$ in the low pressure limit. Besides the obvious boundaries given by $T$, $p$ and $h$, $p$, a melting curve based on the Simon-Glatzel equation can be used to further reduce the application of the interpolation schemes. Special cases like with a negative inclination of the melting curve in $h$, $p$ coordinates, like water, still require the evaluation of the EOS in some cases. With those two data sets, a one-dimensional search in the tables allows for twelve different input pairs. Only the combinations $s$, $u$ and $s$, $v$ and $u$, $v$ will not be covered. Transport properties will be tabulated in $T$, $p$ with numerical derivatives instead of $T$, $v$ to play well together with the other tabulated data.

**Modelling and Optimisation**

This paper discloses new features and capabilities of the open-source simulation program Dynamic Network Analysis developed at the Technical University
of Denmark.

The tool has been interfaced to the open-source thermophysical property library Coolprop. The user can now model novel energy system technologies by leveraging the extended fluid library which includes more than a hundred compressible and incompressible fluids. Rapid and accurate calculations of thermodynamic and transport properties are delivered by virtue of advanced interpolation methods, i.e. tabular Taylor series expansion and bicubic interpolation. Moreover, Dynamic Network Analysis can deal with non-convex multi-variable optimization problem utilizing the fully embedded genetic algorithm.

The new features are here exploited to design an air bottoming cycle module to recuperate the exhaust heat from the gas turbine-based power system installed on an oil and gas facility in the Norwegian Sea. Optimizing the pressure ratios of the two air compressors evidences opportunities for improvements in efficiency and installed power capacity of 5.2 %-points and 2.76 MW, respectively. Furthermore, the application of the genetic algorithm enables to minimize the compactness and the weight of the heat transfer equipment of around 6.2 $10^3$·kg. The full potential of the evolutionary algorithm is exploited in the latter optimization owing to the large discontinuities in the objective function.

Closing Remarks

From a broader perspective, the implemented routines, returning the physical properties of real fluids and optimal plant designs delivered in an open-source environment, constitute essential features to tackle the modelling challenges posed by future energy system in a computationally efficient and accurate way.

REFERENCES


Figure 11: Interpolation tables in $h$, $p$ (solid) and $v$, $T$ (dashed) as generated by CoolProp v4.