# VISUALISATION OF THERMODYNAMICS AND ENERGY PERFORMANCE FOR COMPLEX BUILDING AND HVAC SYSTEM SIMULATION

Stephan Ledinger[1], Marcus Jones[2]
[1]Austrian Institute of Technology, Vienna, Austria
[2]Vienna University of Technology, Vienna, Austria

## ABSTRACT

Complex building and HVAC system simulation is usually carried out incrementally. The process of generating the results of interest from a simulation can be defined by the cycle in figure 1. Slowly adding new information to the already working model and checking whether these additions work as intended is important. During this procedure and for the final generation of results, the modeller wants to check the new results for plausibility and correctness as quickly as possible before proceeding to the next adaptation. The use of automation reduces the effort of visualising the new results. This allows the modeller to concentrate on the analysis and modeling phase and hence speed up the work flow.
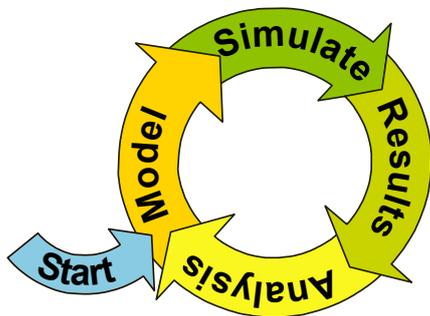
*Figure 1: An approach to simulation*

During the energy modelling and analysis of a complex buildings, new methods for visualising system and building energy performance were developed and employed to manage the complexity of the system. A key feature of the improved methods was the standardisation of simulation results (outputs) and the automation of post-processing these results. Automation should be applicable for not only one specific but as many building and system simulation tools as possible. This requires the standardisation conventions for naming and structuring the simulations results to be intuitive and clear. Additionally, the automation concept includes the possibility to predefine typical forms of diagrams, figures, charts, etc., and dynamically link them to the new simulation results after each simulation.

This paper provides an overview of the new prospects for building and system simulation, emerging out of this visualisation concept. The possible forms of displaying energy performance of a building and its HVAC system are automatically generated;

- time series of various variables like temperatures, flow rates, control signals, capacities, etc.

- three dimensional bar charts for energy balances systems over definable periods,

- animated psychrometric charts, showing all relevant states of a system for a definable time frame

- system overview diagrams that include dynamically linked state point information for an instant in time

The presented concept also has the intention to be applicable to other sources such as monitoring data. This allows, in addition to the streamlined simulation process, a fast validation of the modelled system.

## MOTIVATION

The motivation for visualising results from building performance simulations is to bundle the diversified information of a complex simulation model and bring this to a point where questions of interest can be answered also by persons not familiar with the simulation tool itself. The benefit of streamlining the visualisation process and incorporating definitions from the simulation data post-processing into the modelling process are:

- Increasing the traceability of the simulation from the beginning to the end, also for persons not related to the project

- Allowing to apply post-processing to other simulation models with a similar approach

- Decreasing the effort for debugging the simulation model

- Enabling to distribute work packages like the plausibility check of simulation results to persons familiar with the specific field of investigation but not with the simulation tool

## NAMING CONVENTIONS

The post-processing of simulation or monitoring data requires a certain standardisation of how this data looks like, meaning how the values are stored in which file type and how the post-processing routine recognises them. For the particular example simulation project the simulation environment TRNSYS (Solar Energy Laboratory, Univ. of Wisconsin-Madison 2009) version 17 was used. TRNSYS offers various options of formatting output data. This software structures the simulation model into so-called Types. In this particular case Type 25b with user-supplied units and scientific notation (16 decimals) was used to generate a data file for each state point of the simulation model. Figure 2 shows an example of these state point definitions in the simulation model. The naming convention defines which kinds of state points are available and how they have to be structured, see table 1 and 2. As an example, state point 'MoA1' in figure 2 has to have a printer, Type25b, connected, which generates a result file containing 3 columns:

- simulation time [hr]

- first parameter; Air temperature [°C]

- second parameter; Humidity ratio [kg/kg]

- third parameter; Mass flow of air [kg/hr]



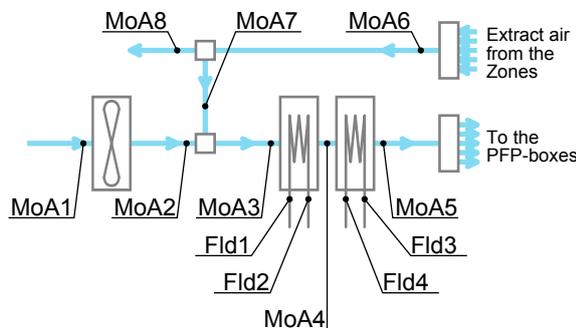*Figure 2: State point naming for a typical air handling unit scheme (VAV)*

*Table 1: State point types*

| Abbreviation | Description | Variables |
|---|---|---|
| MoA | Moist air flow | T, W, mf |
| Fld | Fluid flow | T, mf |
| Tnk | Stratified tank | Ef, T |
| ENel | El. energy consumption | Ef |
| ENth | Th. energy consumption | Ef |
| Irr | Irradiation on surface | Gb, Gd, Gg, Slp, Azm |

Accordingly, the simulation of the displayed system in figure 2 generates twelve output text files of which

*Table 2: State point variables*

| State Variable | Description | Unit |
|---|---|---|
| T | Temperature | [°C] |
| W | Humidity ratio | [kg/kg] |
| mf | Mass flow rate | [kg/hr] |
| Ef | Energy flow rate | [kJ/hr] |
| G | Irradiation | [kJ/hrm$^2$] |
| hf | Enthalpy flow rate | [kJ/hr] |

eight are moist air and four are fluid specific. Each file has a corresponding file name, which consists of the system name, the state point type, and the state point number in this system. Table 3 shows the header and the first data lines of such a file for the state point 'Fld1' in figure 2.

These naming conventions are intended to be applicable to any similar project, but were chosen in regard of the related project and therefore are subject to continuous improvement. The purpose of standardised names for state points and their containing variables is to provide consistent data to the post-processing routine. The data reading subroutine of the post-processing application will then be able to identify and structure each of the simulation result files according to it's file name and the variables according to the header.

The file header contains information about the kind of state point variable and unit, which is read in by the post-processing routine. Based on this information, one of the first procedures in the post-processing is the conversion of units. An example is the convention for the units of power generated by TRNSYS. Types in TRNSYS usually calculate power as [kJ/hr]. For a person, not familiar with TRNSYS, simulation results with that specific unit might cause confusion. Therefore, the post-processing routine converts all affected columns by dividing them by 3.6 to the equivalent average capacity over the time step in [kW].

Another useful immediate operation is the calculation of the enthalpy for each fluid and moist air state point. Based on the enthalpy flow at the state points around a component, the first law of thermodynamics, conservation of energy, can be automatically applied. In example, the enthalpy difference between 'MoA4' and 'MoA3' in 2 must be equal to the enthalpy difference between 'Fld2' and 'Fld1' and represents the capacity with which the first cooling coil is lowering the temperature and humidity of the air and vice versa the chilled water flow is heated by the warm air stream (equation 1).

$$hf_{MoA3} - hf_{MoA4} = hf_{Fld1} - hf_{Fld2} = P_{Cooling1} \quad (1)$$

## POST-PROCESSING

The post-processing routine was programmed in Matlab (The MathWorks Inc. 2007). After a simu-

*Table 3: Example - Fluid state point 1 'CentralAHU_Fld_1.out'*

| TIME | T | mf |
|---|---|---|
| HOURS | [°C] | [kg/hr] |
| +0.000000000000000E+00 | +1.000000000000000E+01 | +1.000000000000000E+02 |
| +2.500000000000000E-01 | +6.733439013590730E+00 | +1.640000000000000E+05 |
| +5.000000000000000E-01 | +6.708215989845478E+00 | +1.640000000000000E+05 |
| +7.500000000000000E-01 | +6.708215989845477E+00 | +1.640000000000000E+05 |
| +1.000000000000000E+00 | +6.708215989845477E+00 | +1.640000000000000E+05 |
| +1.250000000000000E+00 | +6.708215989845477E+00 | +1.640000000000000E+05 |
| +1.500000000000000E+00 | +6.708215989845477E+00 | +1.640000000000000E+05 |

*Table 4: Example - Custom point calculation for balances*

| Name | State type | Unit | Calculation | |
|---|---|---|---|---|
| Bal_Power_1, | hf, | [kW], | Cooling | = VAV.MoA.4 - VAV.MoA.3 |
| Bal_Power_2, | hf, | [kW], | Dehumidification | = VAV.MoA.5 - VAV.MoA.4 |
| Bal_Power_3, | hf, | [kW], | ZoneLoad | = VAV.MoA.6 - VAV.MoA.5 |
| Bal_Power_4, | hf, | [kW], | Fan | = VAV.MoA.2 - VAV.MoA.1 |

lation run, the modeller starts this Matlab-script and, once defined correctly, all the necessary procedures to finally generate readable plots are done automatically. These definitions require locating the simulation results and description files. Description files in this case are simple text files with strings that are read in and interpreted by the Matlab-script as commands. Two of these files were used in the example simulation project. The first, table 5, was used to add descriptive text to the state points in the Matlab data structure. This information is used to identify the stored data in the structure and automatically label graphs et cetera. The second, table 4, was used to automatically execute calculations on the "raw" values like shown in equation 1 and adding the results as new points to the Matlab data structure. These, so called "custom points", can be easily used to calculate energy balances and other analytical operations. The resulting data is automatically stored and can be accessed by the analysis- and plotting script like any other state point from the simulation.

The read, processed and calculated data is structured according to the system names. Having this standardised structure, information for following analyses and generating graphs can be accessed easily. At this development stage of the post-processing routine, monitoring data has to be provided in the same kind of text files and header information. Future extensions of the script aim towards including a conversion routine for typical monitoring data. For more details of the post-processing programming see (Jones and Ledinger 2010).

## TYPES OF FIGURES

Having a standardised structure for all simulation data in Matlab, which allows addressing any of the defined state points in the model over the whole sim-

*Table 5: Example - Descriptive text for state points*

| State point | Descriptive text |
|---|---|
| VAV_MoA_1 | - Fresh air supply |
| VAV_MoA_2 | - Fresh air after fan |
| VAV_MoA_7 | - Recirculated air |
| VAV_MoA_3 | - Mixed air |
| VAV_MoA_4 | - Cooled air |
| VAV_MoA_5 | - Dehumidified air |

ulation time frame, is beneficial for creating various diagrams quickly. This applies especially to the analysis phase of a simulation project, when complications, errors or new points of interest arise and would, without the standardised state point printers, require a new simulation run with specially added printers at the specific system. The Matlab post-processing routing allows to include an analysis script for any specific project. Every time the simulation model is re-simulated, the whole extended post-processing routine is executed and automatically generates the previously defined diagrams with the new simulation results.

The wide spread usage of Matlab results in a community with solutions to numerous programming challenges. Examples for very different graphs can be found online and adapted to the own purpose. The process of writing a script for plotting diagrams with simulation and also monitoring data can be seen as picking and combining adequate modules of code which already exist. Some examples of different types of graphs that can be generated with this procedure follow.

**Time series**

In the analysis of a simulated system, mainly time series of several specific variables are used. The evolu-

tion of temperature, humidity, mass flows, energy consumption, et cetera. over time often is the adequate and important tool to analyse and illustrate the simulated performance of a system.

**Energy balances and bar charts**

For further investigations on the simulated model, energy balances are useful. Time series of the calculated energy rates can be integrated for any period of time and displayed in a bar chart (figure 3).
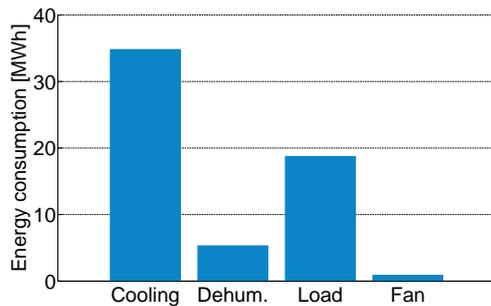


*Figure 3: Example bar chart*

**Three-dimensional bar charts**

Annual simulations often arise the need to split up bar charts into smaller time periods to display seasonal characteristics of systems in an overview. Therefor, three-dimensional bar charts can be used. As an example, figure 4 shows the cumulated energy consumption for several systems for the whole year broken down to months.
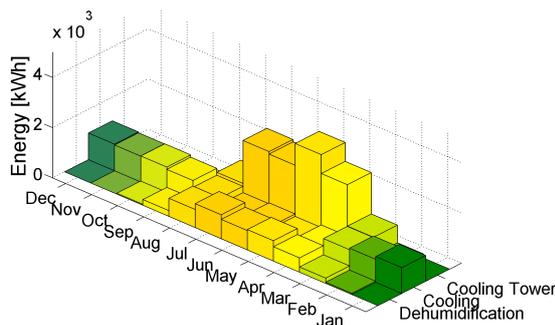


*Figure 4: Example three-dimensional bar chart*

**Peak analysis**

In HVAC system simulations the electrical energy consumption of the main system components are of interest. Other than the annual energy balance of the system, a peak analysis can give information about the maximum electrical capacity that a single or a sum of components consume within the duration of the simulation. As a possible example, figure 5 shows the aver-

age electrical power consumption of a vapor compression chiller in a histogram.
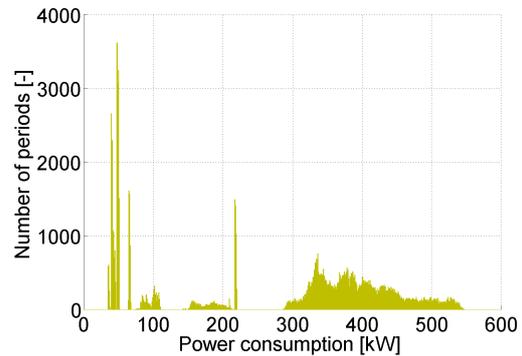


*Figure 5: Example Histogram*

**Psychrometric charts and animations**

Analysing ventilation systems requires psychrometric charts. The post-processing script allows to automatically plot such graphs for a specific system at any instant in the simulation time, once the state points of the system that should be contained are defined. An example of such a graph is given in figure 6. Additionally, an animation, showing this psychrometric chart for a specific time frame can be generated. Interactively displaying all the important state points over several days allows new ways of assessing system performance.
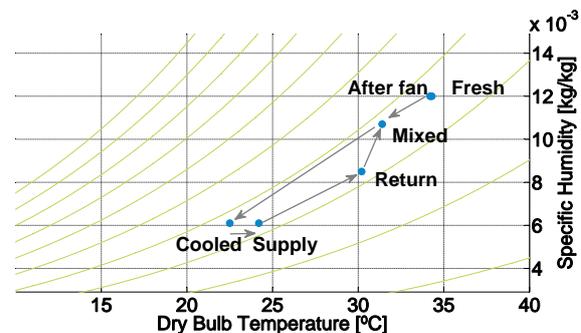


*Figure 6: Example psychrometric chart*

**Scheme visualisation**

Psychrometric charts are useful to evaluate the performance of air based systems. For a more general view, it is helpful to have the actual values of state points next to each of them in a scheme of the system at an instant of the simulation time. Once a scheme of the system is drawn and the desired text boxes are added to the correct location, the post-processing script can parse the relevant information into the text boxes and update the whole graph. An example of

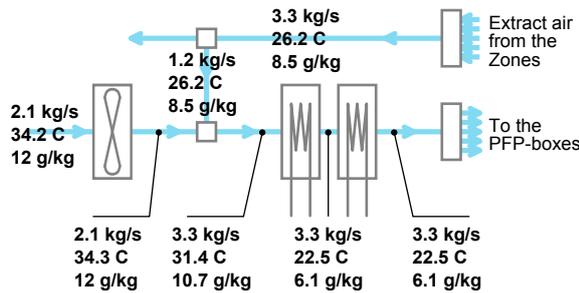such an automatically updated scheme is shown in figure 7.



*Figure 7: Example scheme visualisation*

## CONCLUSION

The visualisation of thermodynamics and energy performance for complex building and HVAC simulations requires advanced usage of data post-processing routines. A standardisation of simulation output files was applied to enable the programmed scripts to be used for other simulations as well as for monitoring data. New methods in post-processing and scripting using Matlab allow the modeller to access and visualise the vast amount of of generated simulation data. The applied routines allow automation of post-processing and visualisation, creating, for example, time series, bar charts, psychrometric analysis quickly and reliably.

These methods enable improved debugging and a higher quality of statistical analysis of the data. This results in a smoother simulation process, see figure 1, and a more detailed understanding of complex systems modellers are tasked to evaluate.

## REFERENCES

Jones, Marcus, and Stephan Ledinger. 2010. "Pushing the limits of simulation complexity - a building energy performance simulation of an exhibition centre in the U.A.E." *Proceedings of Simbuild 2010 - 4th national conference of IBPSA-USA (Pending final acceptance)*.

Solar Energy Laboratory, Univ. of Wisconsin-Madison. 2009. *TRNSYS 17 - a TRaNsient SYstem Simulation program*.

The MathWorks Inc. 2007. MATLAB version 7.5.0.342 (R2007b).