A User-Friendly Dynamic Reactor Simulator Built in Microsoft Excel

Mario Luis Chew Hernandez
Industrial Engineering Department
Technological Institute of Superior Studies of Coacalco
Coacalco, Mexico
mario@tesco.edu.mx

Leopoldo Viveros-Rosas
Industrial Engineering Department
Technological Institute of Superior Studies of Coacalco
Coacalco, Mexico
leopoldo@tesco.edu.mx

Isaac Hernandez Arrieta
Industrial Engineering Department
Technological Institute of Superior Studies of Coacalco
Studies of Coacalco
Coacalco, Mexico
Isaac.subB@tesco.edu.mx

Abstract-Computer plant simulation is being used in all aspects of engineering, through many simulation software packages. However, almost all of them require licenses that must be purchased by engineering colleges intending to use simulation in their teaching. As public educational institutions everywhere are facing a scarcity of economic resources, they can resort to a freeware steady-state plant simulator, however, there is no availability of reliable, free dynamic plant simulators. In addition, published experiences on developing dynamic simulators use programming languages requiring paid licenses (e.g. Matlab) and thus have limited relevancy to schools struggling to cut expenses. This article first uses a set of typical college objectives to discuss the advantages of building their own dynamic simulators, and then shows the development of a user-friendly dynamic simulator of a batch reactor constructed entirely within Microsoft Excel, which, in contrast to the programming languages used in related reports, is already widely used by universities around the world.

Keywords-batch reactor; dynamic simulation; visual basic

I. INTRODUCTION

The typical bachelor Chemical Engineering degree program includes the subject of computer plant simulation in its last semesters [1]. One of the aims of the subject is that the student becomes familiar with commercial simulators, both dynamic and stationary. While some universities are able to purchase simulator licenses, this is not the case of most public institutions in developing countries, where financial resources are limited. As global trademark agreements have led to the prohibition of using unlicensed software in teaching, some institutions have resorted to free-ware software, of which reliable steady-state process simulators can be found (notably, the Cape Open Simulator [2]). However, no reliable, sufficiently tested, and license-free dynamic simulators are

currently at hand. Fortunately, the academic staff of most colleges has the skills required to develop and program process simulators containing most of the elements of professional software. This work shows, taking a batch reactor as an example, how such a simulator can be developed using the functionalities included in the widely used Microsoft Excel program.

The use of transient state simulation to promote industrial performance is well documented, both in its discrete-event version [3-5] and in its continuous form. Regarding the latter, literary reviews can be found in [6] on the application of neural networks in simulating chemical and biochemical processes and in [7] on the modeling of energy systems. Specific applications of simulation can be found in [8] where simulations to optimize plant design with respect to environmental objectives are applied and, more recently, in [9] that pressure dynamics to the simulation of a combined cycle gasification plant are incorporate. Authors in [10] simulated a petrochemical wastewater treatment plant, authors in [11] dealt with a desalination plant energized by ocean waves, authors in [12] optimized the drying process of phosphate pellets, authors in [13] emphasized on energy use when studying a benzene-toluene-xylene distillation system, and authors in [14] treated the control of the activated sludge process.

Regarding the software used to perform the simulations reported in the literature, the usage of simulation environments (e.g. Aspen) and high-level programming languages that provide routines facilitating simulation (e.g. Matlab) has recently increased, as opposed to the use of general-purpose languages (e.g. C++ or Fortran). For example, authors in [15] studied distillation tower sequences with Aspen Dynamics and Matlab Simulink, authors in [16] simulated a polymerization reactor using ChemCAD, and authors in [17] studied a heat

integrated reactor applying Aspen-HYSYS. Authors in [18] used Matlab and COSMOL Multiphysics to simulate heterogeneous catalysis, authors in [19] applied Aspen Dynamics for studying the control of a combustion system, authors in [20] modeled an activated sludge process using INSEL, and authors in [21] used Aspen and Matlab to evaluate a predictive control scheme for CO_2 absorption plants. Didactic applications of dynamic simulations are reported in [22], in which the authors used Aspen HYSIS to train engineers in statistical methods to analyze distillation systems and in [23] where the Honeywell Unisim Design software was applied to simulate an isopropyl alcohol plant and train its operators. Instances of reports on the development of process simulators for didactic purposes are found in [24], in which a steady-state free-ware simulator was designed to which third parties could make additions in a "wiki" scheme, called *Lazarus*, running on UML language, and [25, 26] in which educational software was developed to analyze the dynamics of exchanger networks. Finally, authors in [27] used an online simulator to train operators to manage overpressures in natural gas systems, authors in [28] built the "LABVIRTUAL" platform, with educational material including simulators, and authors in [29] developed a Matlab-based dynamic simulator for chemical reactors.

Previous reports on the development of dynamic simulators for training purposes use either a commercial simulator or an advanced programming language (Matlab). However, the usage of such software requires the purchase and continual renovation of licenses. This limits the relevancy of the mentioned approaches to cash-strapped public schools in developing countries, who may find simulators running on easily accessible, universally used programming languages, like the one sketched in [30], far more useful. The current work builds on the latter reference by discussing, from the perspective of institutional objectives, the convenience for schools to build their own simulators, and presenting a complete description of an Excel-based dynamic simulator of a batch reactor. The shown simulator contains many of the features offered by professional software (button-based operations, filling the blank inputs, dialogue boxes, etc.).

II. METHODS

This article has two main purposes, for which different methods are applied:

- The first is to show, the convenience for colleges on developing their own simulators. This is done through objective analysis [31]: a hierarchy of fundamental objectives is built and the probable impact of the alternative on the objectives is assessed.

- The second is to present the dynamic simulator of a batch reactor developed in Microsoft Excel.

III. RESULTS AND DISCUSSION

A. Analysis of Institutional Objectives

Generally speaking, public education institutions have two conflicting objectives: an academic one, related to the development of professional skills of their students and an economical one, which refers to minimizing cost. This is shown in the structure in Figure 1.
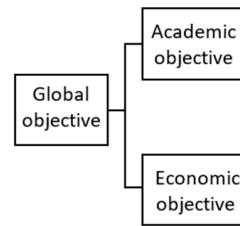
Fig. 1.   Fundamental objectives hierarchy.

When contemplating the purchase of an educational tool, an institution should balance its price and the way it may benefit student formation. This is shown in Figure 2, where the objective damaged by the alternative is pointed at by a dotted arrow while that which is benefited by a solid one.
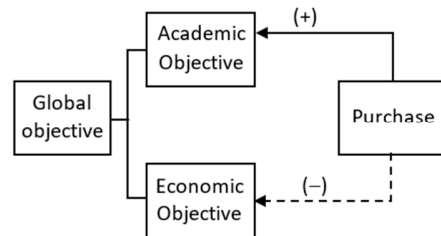
Fig. 2.   Purchase impact on objectives.

The academic objectives aim for the student to acquire relevant professional skills. This is shown in Figure 3, where $C_{i,j}$ denotes skill i of subject j (for example, $C_{1,QG}$ refers to skill 1 of the subject "General Chemistry"). Figure 3 represents three skills covered in the subject "Process Simulation": $C_{1,SP}$, $C_{2,SP}$, and $C_{3,SP}$. $C_{3,SP}$ is defines as "The student will use commercial process simulators" [1]. Figure 3 breaks down the $C_{3,SP}$ skill into its steady-state simulation element *SS* part and a dynamic simulation one, *Dyn*.
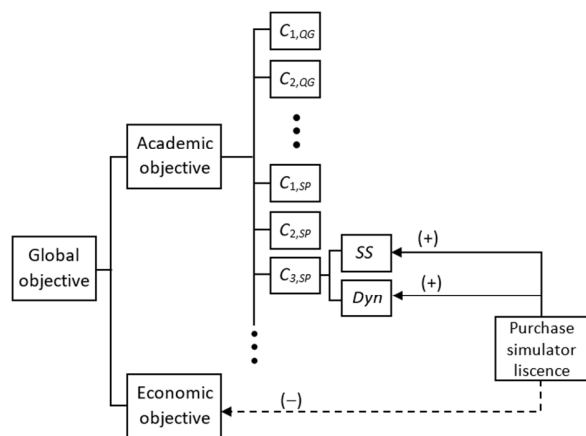
Fig. 3.   Details of academic objectives.

If a school hasn´t already bought a simulator license, then it views the purchase impact on $C_{3,SP}$ as not worth its cost. For the steady-state simulation part of $C_{3,SP}$, this is solved by downloading one of the trustable, free-ware stationary plant simulators that can be found on-line. However, as there is no availability of transient state simulators with these qualities, the part of $C_{3,SP}$ dealing with dynamic simulation goes uncovered. Nevertheless, if the institution was to build its own dynamic simulator, this deficiency would be corrected (Figure 4).
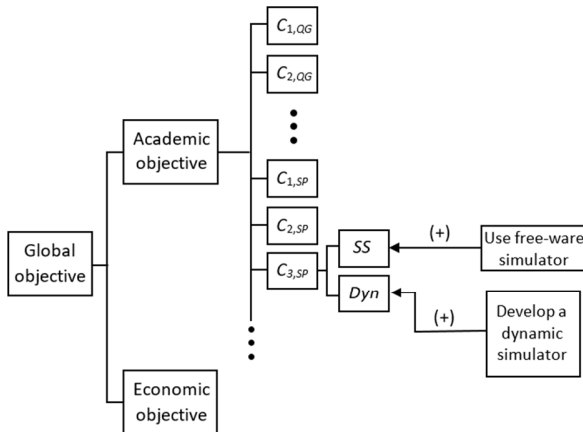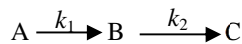

Fig. 4.   Institutional options.

B. Process Description

The reactor to be simulated is based on the case described in [32], which features a steam heating jacket at temperature T_S and a cooling coil. A pair of consecutive reactions occur in the reactor.

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

Reaction A→B has a second order kinetics while B→C is a first order reaction and component B is the desired product.

The assumptions used in the model are:

- The reactor is thermally insulated and perfectly mixed.

- The temperature of the water in the coil is described by its average T_C value.

- Condensed steam is disposed at its saturation temperature.

- Reaction volume V and specific heat capacity C_P are constant.

- The heat transfer coefficient U_j of the heating jacket is constant while the one of the cooling coil U_C depends on the F_C cooling water flow according to:

$$\frac{1}{U_C} = \frac{1}{4550 F_C^{0.8}} + \frac{1}{10.8} \quad (1)$$

Species and energy balances produce the following three differential equations, which describe reactor dynamics:

$$\frac{dC_A}{dt} = -k_1 C_A^2 \quad (2)$$

$$\frac{dC_B}{dt} = k_1 C_A^2 - k_2 C_B \quad (3)$$

$$\frac{dT}{dt} = \frac{(-\Delta H_1)}{\rho C_P} k_1 C_A^2 + \frac{(-\Delta H_2)}{\rho C_P} k_2 C_B + \frac{U_j A_j}{\rho C_P V} (T_S - T) - \frac{U_C A_C}{\rho C_P V} (T - T_C) \quad (4)$$

C_A and C_B are the concentrations of species A and B, $(-\Delta H_1)$ and $(-\Delta H_2)$ the enthalpies of reactions A→B and B→C respectively, ρ and C_P are the liquid density and specific heat capacity, and U_j and U_C are the global heat transfer coefficients of the heating jacket and cooling coil of heat transfer area respectively, A_j and A_C. Reaction constants k_1 and k_2 and the temperature T are related by:

$$k_1 = \alpha_{10} \exp\left(\frac{-E_1}{RT}\right) \quad (5)$$

$$k_2 = \alpha_{20} \exp\left(\frac{-E_2}{RT}\right) \quad (6)$$

where E_1 and E_2 are the reaction activation energies and α_{10} and α_{20} their respective frequency factors. When the reactor is in closed-loop operation, the water flow to the cooling coil (F_C) and the steam temperature in the heating jacket (T_S) are manipulated to control the reactor temperature (T). In order to improve the yield of product B, the temperature set-point (T_d) may follow a time (t) dependent trajectory [32].

$$T_d(t) = 54 + 71 \times \exp(-2.5 \times 10^{-3} t) \quad (7)$$

As there are two manipulated variables, T_S and F_C, and one controlled variable T, the controller adjusts, through a proportional-integral control logic, the value of a parameter u, that linearly sets the value of the manipulated variables between their maximum and minimum values, according to:

$$T_S = (T_{S,max} - T_{S,min})u + T_{S,min} \quad (8)$$

$$U_C = (U_{C,min} - U_{C,max})u + U_{C,max} \quad (9)$$

The implicit Euler method is used to solve the differential equations and a Newton-Raphson procedure is used to solve the nonlinear algebraic equations in each integration step.

C. Simulator Description

The simulator was implemented as an Excel Macro. It is delivered to the user as an Excel file, the opening of which produces Figure 5. Clicking on the "Batch reactor" button displays the user interface (Figure 6). At the interface top (Figure 7) the simulation time and initial conditions are set: concentrations of A and B and temperature. The final values of these variables will appear in the "Final CA", "Final CB" and "Final T" dialog boxes. The user can propose a step size for the integration (10s is the default value). If the implicit Euler method does not converge for this step size, the program reduces it until convergence can be achieved. The step size used is shown the "Final step size (h)" box.
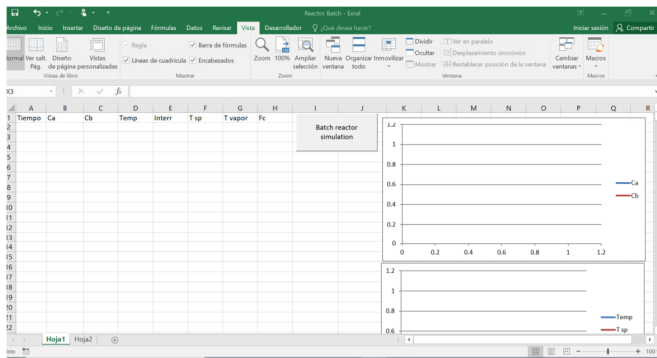
Fig. 5.   Initial screen.

Fig. 6.   User interface.

Fig. 7.   Run initial state and parameters.

Fig. 8.   Heating/cooling characteristics.

The characteristics of the heating/cooling are set to the right of the interface (Figure 8). If "Adiabatic" is selected, no heat transfer is considered and when "With heat transfer" is chosen,

dialog boxes become active so the heat transfer areas per unit volume for the cooling coil and steam jacket are entered. If the "With heat transfer" option is selected, fields become active for the user to specify how the cooling water flow and steam temperature are to be managed. When selecting the "Open Loop" option, the user must input a value for the water flow and steam temperature, which remains constant throughout the simulation. If the cooling does not begin at time zero, a cooling commencement time can be specified (Figure 9). If the "Closed Loop" option is checked, the user should decide the temperature set point value management scheme: if the option "Automatic" is active, the temperature set point follows the time-dependent path of (7). If "Manual" is selected, the user inputs the temperature set point. Additionally, the proportional gain (K_C) and integral reset time (tau) of the temperature controller must be specified.
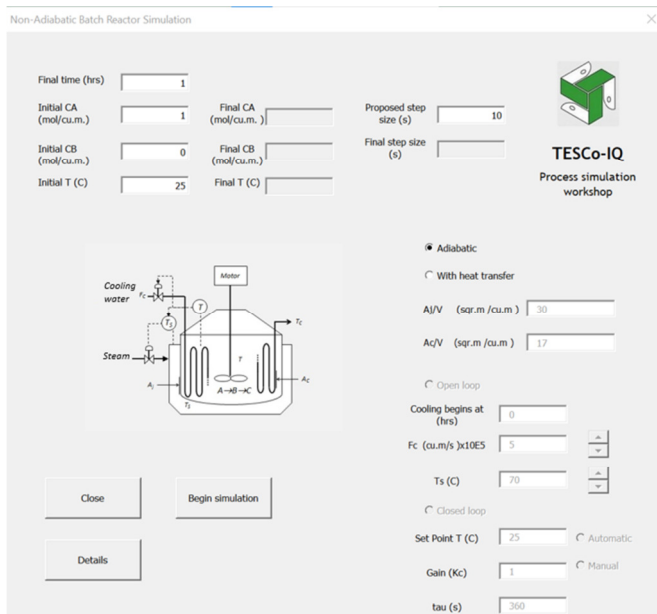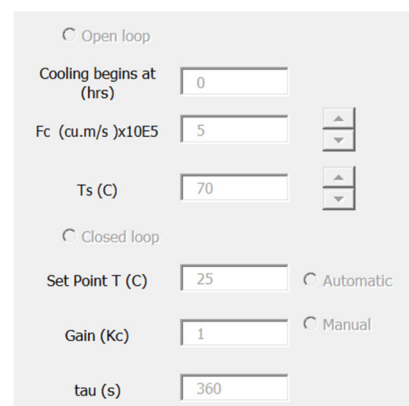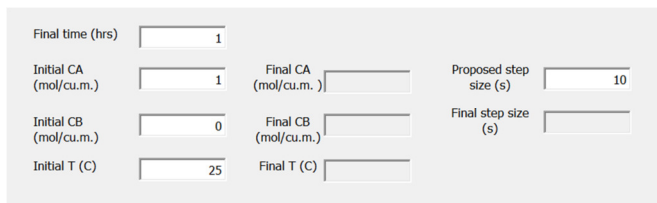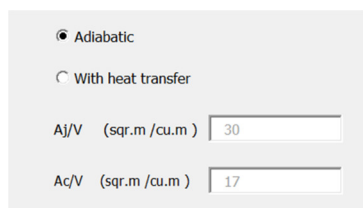

Fig. 9.   Control loop specification.

The functionality of the interface (Figure 6) is completed by the buttons "Close", which closes it, "Start Simulation", which begins integrating the differential equations, and "Details", which displays the problem equations. Once "Start Simulation" is clicked, the interface hides and a progress bar appears (Figure 10). The columns A-F of the Excel sheet are cleared as the new results will appear there.
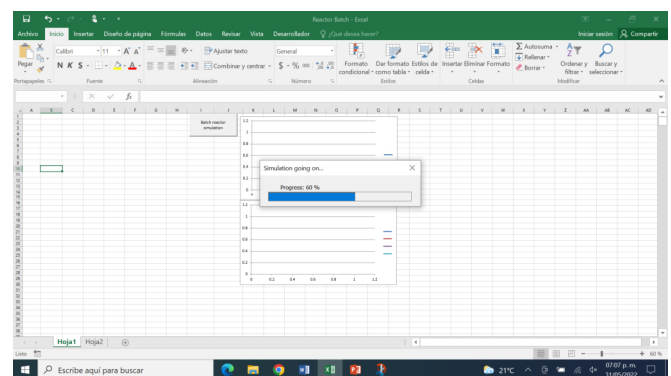

Fig. 10.   Progress bar.

Once the simulation is completed, the simulation interface reappears and a dialog box asks if the results are to be graphed

(Figure 11). If "Yes" is answered, the results are placed in the spreadsheet, and the graphs are updated with the most recent results (Figure 12). If "No" is clicked, the dialog box disappears, and the user can read the final concentrations and temperature in the corresponding dialog boxes.
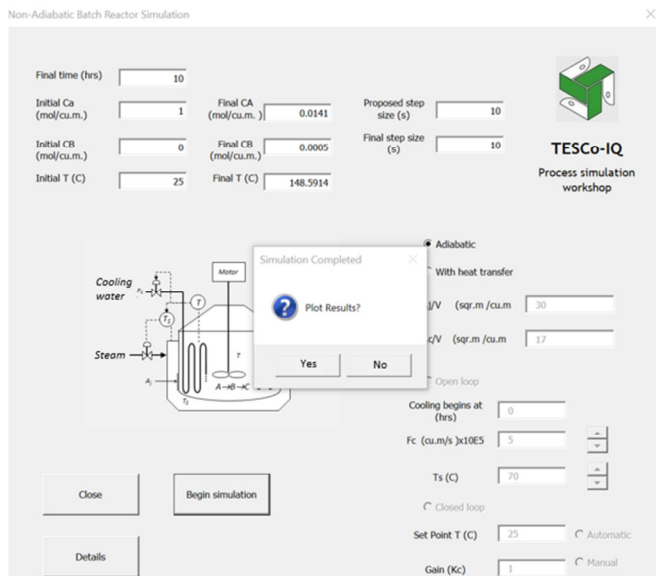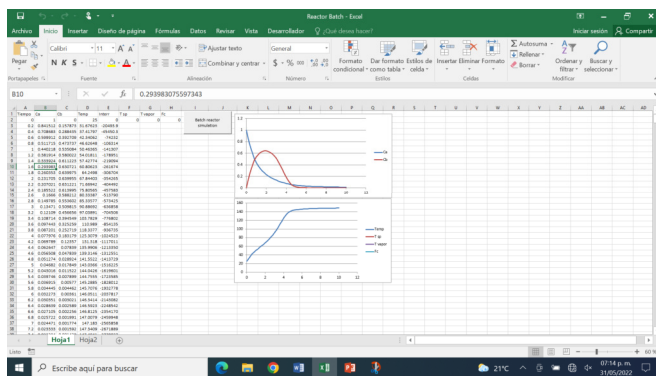

Fig. 11.   "Simulation completed" message.

Fig. 12.   Final charts.

IV.   CONCLUSIONS

As public education institutions receive taxpayer's money, they must use their resources efficiently, minimizing expenses that are perceived as unnecessary. Due to this, some institutions have opted not to purchase simulator licenses, thus limiting their students' acquisition of the skills related to the usage of this type of software. As far as the steady-state simulation of chemical processes is concerned, this can be solved by using a simulator downloadable for free, as some of them have been extensively validated. This is not the case, however, for the simulation of processes in a dynamic or transient state, where there are no free simulators that have been sufficiently tested. However, higher education institutions in most countries have the necessary human capacities to produce their own process simulators. This paper argues, based on a set of typical

institutional objectives, that this is a convenient alternative for schools, and illustrates how a dynamic simulator developed using the Microsoft Excel software, an everywhere present tool, looks like. It is hoped that this will encourage more institutions to develop their own dynamic process simulators.

The main advantage of the shown simulator is its running on Microsoft Excel that is already widely used by most engineering faculties in developing countries, and so it can be used without purchasing another license. One of its shortcomings, compared to the most powerful commercial dynamic process simulators available (eg. HYSYS dynamics), is that it doesn´t show an animation of the real-time behavior of the variables, but rather calculates a response and then plots it. This, however, suffices for tuning of controllers and studying the dynamics of the system when faced with several disturbances, which are some of the main applications of simulation.

Finally, the presented simulator requires that the version of MS Office its run in codifies Macros in the Visual Basic programming language. This does not appear to be a serious limitation, as MS Excel has maintained this condition in versions as old as Office 95, and new desktop versions of MS Office are projected to continue to use Visual Basic as their Macro-Programming language for the foreseeable future. However, cloud or web-based versions of Excel (eg. in Office 365) as well as open versions of Office are transitioning to a java-script kind of language for macro programming, and the simulator in its current form cannot be expected to run in these programs. The simulator would need to be re-programmed in said languages, of made to run directly in a Visual Basic stand-alone programming suite. The basic simulation functionalities shown here, however, would be maintained in these changed simulator versions.

REFERENCES

[1]   "Licenciaturas," *Tecnológico Nacional de México*. https://www.tecnm. mx/?vista=Licenciaturas.

[2]   "COCO - the CAPE-OPEN to CAPE-OPEN simulator." https://www. cocosimulator.org/.

[3]   B. O. Odedairo and N. Nwabuokei, "Framework for Operational Performance Measurements in Small and Medium Scale Industries Using Discrete Event Simulation Approach," *Engineering, Technology & Applied Science Research*, vol. 8, no. 4, pp. 3103–3107, Aug. 2018, https://doi.org/10.48084/etasr.2106.

[4]   M. L. C. Hernandez, L. V. Rosas, R. F. R. Mantilla, G. J. E. Martínez, and V. V. Romero, "Supply Chain Cooperation by Agreed Reduction of Behavior Variability: A Simulation-based Study," *Engineering, Technology & Applied Science Research*, vol. 7, no. 2, pp. 1546–1551, Apr. 2017, https://doi.org/10.48084/etasr.1039.

[5]   M. L. C. Hernandez, E. K. V. Hernandez, and S. L. Dominguez, "A Decision-Analytic Feasibility Study of Upgrading Machinery at a Tools Workshop," *Engineering, Technology & Applied Science Research*, vol. 2, no. 2, pp. 182–189, Apr. 2012, https://doi.org/10.48084/etasr.139.

[6]   C. Renotte, A. Vande Wouwer, Ph. Bogaerts, and M. Remy, "Neural Network Applications in Non-Linear Modelling of (Bio)Chemical Processes," *Measurement and Control*, vol. 34, no. 7, pp. 197–201, Sep. 2001, https://doi.org/10.1177/002029400103400702.

[7]   Adams II T. A., Ed., *Modeling and Simulation of Energy Systems - Processes, Special Issue*. MDPI, Basel, 2019, https://doi.org/10.3390/ books978-3-03921-519-5.

[8]   P. E. Bauer and R. Maciel Filho, "Incorporation of environmental impact criteria in the design and operation of chemical processes," *Brazilian*

Journal of Chemical Engineering, vol. 21, pp. 405–414, Sep. 2004, https://doi.org/10.1590/S0104-66322004000300005.

[9]   D. Huang, H. Zhang, S. Weng, and M. Su, "Modeling and Simulation of IGCC Considering Pressure and Flow Distribution of Gasifier," *Applied Sciences*, vol. 6, no. 10, Oct. 2016, Art. no. 292, https://doi.org/10.3390/app6100292.

[10]   I. M Ariff and M. Bakir, "Dynamic Simulation of Petrochemical Wastewater Treatment Using Wastewater Plant Simulation Software," *MATEC Web of Conferences*, vol. 203, no. 1, Jan. 2018, Art. no. 03005, https://doi.org/10.1051/matecconf/201820303005.

[11]   Y.-H. Yu and D. Jenne, "Numerical Modeling and Dynamic Analysis of a Wave-Powered Reverse-Osmosis System," *Journal of Marine Science and Engineering*, vol. 6, no. 4, Dec. 2018, Art. no. 132, https://doi.org/10.3390/jmse6040132.

[12]   V. Meshalkin, V. Bobkov, M. Dli, and V. Dovì, "Optimization of Energy and Resource Efficiency in a Multistage Drying Process of Phosphate Pellets," *Energies*, vol. 12, no. 17, Jan. 2019, Art. no. 3376, https://doi.org/10.3390/en12173376.

[13]   S. Silviana, F. Dalanta, D. Q. A'yuni, L. Khoiriyah, P. R. Nabila, and M. F. Alfaris, "Design simulation and economic optimization of a benzene-toluene-xylene system distillation process upon the energy cost," *E3S Web of Conferences*, vol. 202, 2020, Art. no. 10003, https://doi.org/10.1051/e3sconf/202020210003.

[14]   A. Lemita, S. Boulahbel, and S. Kahla, "Gradient Descent Optimization Control of an Activated Sludge Process based on Radial Basis Function Neural Network," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, pp. 6080–6086, Aug. 2020, https://doi.org/10.48084/etasr.3714.

[15]   M. Khodadoost and J. Sadeghi, "Dynamic Simulation of Distillation Sequences in Dew Pointing Unit of South Pars Gas Refinery," *Journal of Chemical and Petroleum Engineering*, vol. 45, no. 2, pp. 109–116, Dec. 2011, https://doi.org/10.22059/jchpe.2011.1512.

[16]   I. Šoljić Jerbić, S. Kuzmić, and A. Jukić, "Dynamic Simulation of Batch Polymerization Reactor and Sensitivity Analysis of Styrene Homopolymerization," *Kemija u industriji : Časopis kemičara i kemijskih inženjera Hrvatske*, vol. 64, no. 3–4, pp. 151–167, Mar. 2015, https://doi.org/10.15255/KUI.2013.039.

[17]   T. P. Adhi and M. I. Prasetyo, "Process Stability Identification Through Dynamic Study of Single-bed Ammonia Reactor with Feed-Effluent Heat Exchanger (FEHE)," *MATEC Web of Conferences*, vol. 156, 2018, Art. no. 03003, https://doi.org/10.1051/matecconf/201815603003.

[18]   A. K. Patan, M. Mekala, and S. K. Thamida, "Dynamic Simulation of Heterogeneous Catalysis at Particle Scale to Estimate the Kinetic Parameters for the Pore Diffusion Model," *Bulletin of Chemical Reaction Engineering & Catalysis*, vol. 13, no. 3, pp. 420–428, Dec. 2018, https://doi.org/10.9767/bcrec.13.3.2098.420-428.

[19]   T. Wanotayaroj, B. Chalermsinsuwan, and P. Piumsomboon, "Dynamic simulation and control system for chemical looping combustion," *Energy Reports*, vol. 6, pp. 32–39, Feb. 2020, https://doi.org/10.1016/j.egyr.2019.11.038.

[20]   F. Calise, U. Eicker, J. Schumacher, and M. Vicidomini, "Wastewater Treatment Plant: Modelling and Validation of an Activated Sludge Process," *Energies*, vol. 13, no. 15, Jan. 2020, Art. no. 3925, https://doi.org/10.3390/en13153925.

[21]   Q. Li, W. Zhang, Y. Qin, and A. An, "Model Predictive Control for the Process of MEA Absorption of CO2 Based on the Data Identification Model," *Processes*, vol. 9, no. 1, Jan. 2021, Art. no. 183, https://doi.org/10.3390/pr9010183.

[22]   I. M. Joao and J. M. Silva, "Designing Solutions by a Student Centred Approach: Integration of Chemical Process Simulation with Statistical Tools to Improve Distillation Systems," *International Journal of Engineering Pedagogy (iJEP)*, vol. 7, no. 3, pp. 4–18, Sep. 2017, https://doi.org/10.3991/ijep.v7i3.6795.

[23]   J. Puskás, A. Egedy, and S. Németh, "Development of operator training simulator for isopropyl alcohol producing plant," *Education for Chemical Engineers*, vol. 22, pp. 35–43, Jan. 2018, https://doi.org/10.1016/j.ece.2017.11.003.

[24]   S. M. Riachi, M. Duarte, and J. A. Scortechini, "Diseño De Un Simulador De Procesos Químicos Para Uso Colaborativo Y Didáctico," *REFCalE: Revista Electrónica Formación y Calidad Educativa*, vol. 2, no. 1, pp. 71–82, Apr. 2014.

[25]   L. M. F. Lona, F. A. N. Fernandes, M. C. Roque, and S. Rodrigues, "Developing an educational software for heat exchangers and heat exchanger networks projects," *Computers & Chemical Engineering*, vol. 24, no. 2, pp. 1247–1251, Jul. 2000, https://doi.org/10.1016/S0098-1354(00)00324-0.

[26]   S. J. M. Cartaxo, P. F. G. Silvino, and F. A. N. Fernandes, "Transient analysis of shell-and-tube heat exchangers using an educational software," *Education for Chemical Engineers*, vol. 9, no. 3, pp. e77–e84, Jul. 2014, https://doi.org/10.1016/j.ece.2014.05.001.

[27]   Y. Lee, C. Ko, H. Lee, K. Jeon, S. Shin, and C. Han, "Interactive plant simulation modeling for developing an operator training system in a natural gas pressure-regulating station," *Petroleum Science*, vol. 14, no. 3, pp. 529–538, Aug. 2017, https://doi.org/10.1007/s12182-017-0170-5.

[28]   J. F. O. Granjo and M. G. Rasteiro, "Enhancing the autonomy of students in chemical engineering education with LABVIRTUAL platform," *Education for Chemical Engineers*, vol. 31, pp. 21–28, Apr. 2020, https://doi.org/10.1016/j.ece.2020.03.002.

[29]   R. Molina, G. Orcajo, Y. Segura, J. Moreno, and F. Martínez, "KMS platform: A complete tool for modeling chemical and biochemical reactors," *Education for Chemical Engineers*, vol. 34, pp. 127–137, Jan. 2021, https://doi.org/10.1016/j.ece.2020.09.003.

[30]   M. L. C. Hernández, T. R. G. Gómez, G. B. Brugada, and D. E. R. Vargas, "Desarrollo Del Simulador Dinámico De Un Reactor Batch Con Fines Didácticos (building a Dynamic Simulator of a Batch Reactor for Educational Purposes)," *Pistas Educativas*, vol. 43, no. 139, pp. 34–49, Dec. 2021.

[31]   R. L. Keeney, *Value-Focused Thinking: A Path to Creative Decisionmaking*. Cambridge, MA, USA: Harvard University Press, 1996.

[32]   A. K. Jana, *Chemical Process Modelling and Computer Simulation*, 2nd ed. New Delhi, India: Phi Learning pvt Ltd., 2011.