Combining High-level Programming Languages and Spreadsheets An Alternative Route for Teaching Process Synthesis and Design

Mauricio A. Colombo*, María Rosa Hernández* and Jorge E. Gatica
Chemical Engineering Department, Cleveland State University, Cleveland, OH
*** Instituto de Ingeniería Química, Universidad Nacional de Tucumán, ARGENTINA**

Abstract

The present paper focuses on the combination of programming languages as an effective approach towards teaching process design concepts. The cornerstone for the success of this approach is the combination of widespread techniques, such as spreadsheets, with high-level languages as C or FORTRAN and object-oriented languages such as JAVA or Visual BASIC. This paper specifically demonstrates the combination of Microsoft Excel with FORTRAN programs, through a Visual BASIC interface. The popularity of spreadsheets makes them the ideal tools to illustrate nonlinear interrelations among different design variables in complex processes. These variables, however, are typically connected through a complex set of algebraic and differential equations, whose solution demands a robust numerical approach. Students are frequently distracted or frustrated in trying to deal with these difficulties. When these calculations are not essential to the process design principles being demonstrated, instructor-developed “black boxes” can be used to carry out cumbersome calculations in the background. This synergetic effect enables to highlight the most relevant process synthesis principles, while the student is kept away from the mathematical and numerical complexities involved in the solution of the problem. The case study presented serves to illustrate the effectiveness of a proper combination of programming techniques with conceptual design ideas.

I. Introduction

The use of calculation packages in Chemical Engineering (FLOWTRAN, PROCESS, and TK Solver) has become increasingly popular with technical advances in hardware. The use of these packages, however, is not an integral part of Chemical Engineering curriculum. Furthermore, there is not a consensus on which program or programs are preferable (Harb et al., 1997). This decision seems to depend in general of two factors: the instructor, and the course main topic.

Despite the technical advances in operating systems, and the development of graphical user interfaces that simplify the use of complex simulation packages, the controversy is still unresolved. The technological advances in new or alternative processes to deal with new financial, technical, environmental, and/or social issues impose new demands on instructors and course curricula. Indeed, while the total credits to obtain a Chemical Engineering degree has remained practically constant, a steadily increasing demand for supplementary material has been observed in courses covering fundamental Chemical Engineering principles (cf. Hernández and Gatica, 1997). Thus, much of the support material, such as computer programming and numerical analysis, has been relegated to support courses.

Senior-level courses, such as Process Synthesis and Design, are meant to demonstrate the combination of the fundamentals learnt in Sophomore and Junior years in the analysis of complex processes (cf. Douglas, 1988). Attention, however, has been primarily paid on the development of new analysis and simulation methods, while the process synthesis has remained as heavily dependent upon the expertise and experience of the engineer.

Experience in the classroom has shown that requiring students to develop their own solvers, when learning fundamental engineering principles, is typically an overwhelming and frustrating practice. Indeed, besides a solid understanding of the principles being demonstrated, developing a solver requires mastering programming and numerical analysis skills. The most common route to overcome these roadblocks, is to train students to use pre-packaged commercial, instructor-developed solvers, or some combination of both. The shortcoming to this approach is that students use these programs as "black boxes" that implement some "esoteric" algorithm or numerical method. Moreover, the lack of motivation to understand the software, typically results in students losing interest in understanding what happens "behinds the curtains" or how the interrelation between engineering principles influences the calculations.

This paradox is very apparent in sophomore and junior-level courses. Seniors, on the other hand, concentrate their attention on process synthesis and design. Most teaching philosophies dictate to steer senior students to integrate principles learnt during their junior and sophomore years to well defined process synthesis and design problems. The goal of the instructor is to now instill in the students an engineering approach towards the problem. In other words, students should develop the skills to clearly pinpoint the critical units in the process and identify the design variables which have a major impact on the quality of the final design and/or problem solution.

This paper proposes an alternative based on a combination among different programming languages to streamline tutorial-like exercises for users with minimal programming skills. Particular attention is given to new programming techniques and the potential of combining novel object-oriented languages such as JAVA or Visual BASIC (cf. Bruzas and Gatica, 1999) with interfaces which are common to most engineering students. On this end, spreadsheets such as Microsoft Excel, Lotus 1-2-3, and Corel Quattro Pro are the natural choice (Mitchell, 1997). The objective of this work is to show through a case of study, the usefulness of efficiently merge a systematic procedure of processes synthesis with suitable software.

The case study selected to illustrate this approach consists on an optimization problem that requires the solution of a set of coupled non-linear ordinary differential equations. This paper specifically demonstrates the combination of Microsoft Excel with FORTRAN programs, through a Visual BASIC interface. The spreadsheets illustrate the effect of nonlinear interrelations among the different design variables, while instructor-developed “black boxes” are used to carry out cumbersome calculations in the background. This synergetic effect enables to highlight the most relevant process synthesis principles, while the student is kept away from the mathematical and numerical complexities involved in the solution of the problem. The case study presented serves to illustrate the effectiveness of a proper combination of programming techniques with conceptual design ideas.

II. Statement of the Problem

Process Design problems typically present a challenge for Chemical Engineering students. The assignment can rapidly become a frustrating experience unless a systematic analysis approach is followed. The underlying philosophy is to formulate several alternate designs and evaluate their appropriateness to reach a given (usually economic) goal.

Although one can resort to heuristics to combine creativity and theoretical background to simplify the problem, a large number of design variables to define is typically present in these problems. Thus, a computational tool that would enable the student to rapidly explore different design alternatives and the impact of design variables on the objective function, becomes of primary importance. A number of user-friendly of process simulation packages (e.g. Aspen Plus) are readily available to students. To take advantage of these tools, nevertheless, the student needs to have a solid understanding of the flowsheeting process and the design variables.

Due to the open-ended nature characteristic of processes synthesis problems, their solution typically reduces to breaking down the overall problem into smaller subsystems. A precedence order for their resolution, with hierarchical levels of growing complexity is established next (cf. Douglas 1988). This strategy possesses the following advantages:

- It yields a larger number of alternative flow sheets.
- It highlights key design variables
- It enables to identify the variables that cause process units interaction.

A general procedure for the synthesis of chemical processes ("onion diagram") is outlined in Fig. 1.This procedure clearly highlights the hierarchical levels in the definition of design variables

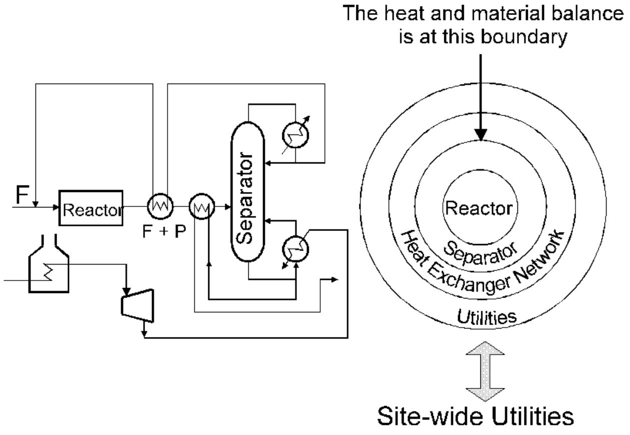


Figure 1: Process design hierarchical "Onion diagram"
(adapted from Douglas, 1988)

Thus, it is apparent that the reactor performance is at the core of the process, the reactor performance will dictate the necessary conditions for the separator, which in turn will define the heat exchanger network, with the latter determining the utilities consumption.

Even with use of heuristics, the reactor presents numerous variables to specify whose determination is not a simple task! How to select the value of important design variables as temperature, pressure, molar ratio of reactants, etc? Unfortunately, in agreement with Murphy´s laws, the heart of the proposed system, of which the value of other proposed subsystems depends, is the most difficult to solve,, even if we had *all* the needful information.

The solutions of the equations that model reactor behavior are, for their nature, complicated and time consuming . As a consequence of that, the solutions are always limited, based on simplified assumptions and they depend on a limited set of selected variables in order to their numerical computation. With the use of tools like the proposed in this work, the students can extend their calculations to more rigorous models, obtaining a deeper knowledge of the requirements of the synthesis problem.

Even more, the combination of powerful computational tools with graphic interfaces allows to explore trade offs among different variables to find an optimal solution. It is highly advisable to encourage students to evaluate and to recommend alternatives according to the desired targets, using the deeper knowledge acquired of the synthesis process, being able to identify the most important variables and their influence. This and many other opportunities can be achieved through the use of the proposed tool.

.

III. Case Study

The case study selected consists of catalytic dehydrogenation process for the production of butadiene (cf. Fig. 2).The objective is to optimize the operating conditions to yield annual production of 25,000 ton of butadiene. A thermodynamic analysis (which is also integrated in the working file) indicates that a catalytic process may be the only feasible route to inhibit complete combustion reactions.

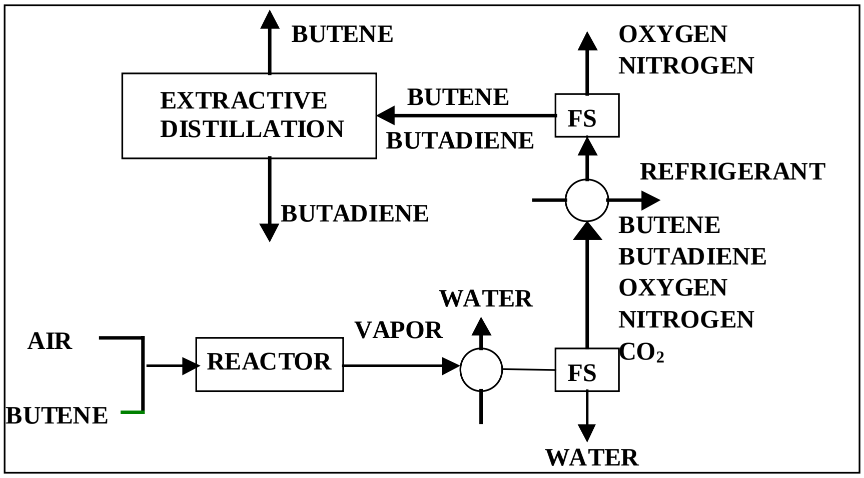


Figure 2: Schematic of the catalytic butene dehydrogenation process

The catalytic butene dehydrogenation occurs through a complex mechanism of combined reactions as shown in Figure 3.

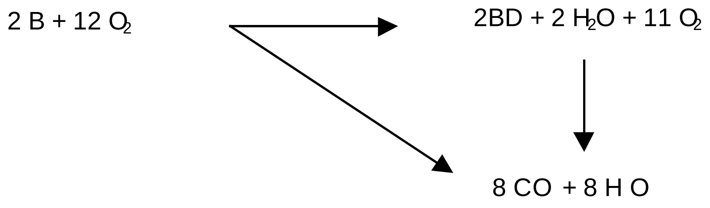


Figure 3: Catalytic dehydrogenation of butene to butadiene mechanism

IV. Formulation of the Design Problem

The design problem can be formulated in two stages:

☐ Define the reactor operating conditions, and
☐ Optimize the process financial profit for these operating conditions.

The solution to the first design question reduces to a parametric analysis of the operating conditions space. In other words, a set of non-linear ordinary differential equations must be solved repeatedly for different operating conditions. The complexity of the governing equations is compounded by the intricacy of the reaction mechanism for this process (cf. Fig. 3).

The first question posed requires optimizing conversion and selectivity. The pressure, temperature, and composition of the inlet stream of the reactor will define the operating conditions. Although the volumetric flow rate and the reactor volume are also design variables, the problem can be easily optimized by its optimum residence time by resorting to dimensionless variables. Once the optimum values for the conversion per pass and selectivity are found, the flow rate (to treat) will define the reactor volume needed.

V. Model Formulation

For the sake of simplicity, this reactive process has been assumed isothermal. Thus, only five ordinary differential equations are needed. For a packed bed reactor, these equations take on the form:

$$\frac{dF_i}{dV} = r_i$$

where, “F” is species “i” molar flux , “V” is the reactor volume, and “r” stands for the rate of reaction expression. For instance, for butene ("B"), the governing equation would be:

$$\frac{dF_B}{dV} = -2r_1 - 2r_3$$

or, in dimensionless form:

$$\frac{dx_B}{d\theta} = \mathrm{Da}_1\ \mathrm{f}_1(x_B, x_O, Y_{BD}) + \mathrm{Da}_2\ \mathrm{f}_2(x_B, x_O, Y_{BD})$$

where “x” represents the conversion, “θ” is the dimensionless residence time, “Y” is the yield, “f” is a concentration non-linear function, while “Da” is the Damköhler number. The Damköhler number is defined as:

$$\mathrm{Da}_i = \frac{k_i \tau}{C_B^o}$$

where “k_i” is the reaction rate constant (with a Arrhenius exponential dependence on temperature, “τ” is the characteristic time, and C_B^o is the inlet butene concentration. All kinetic parameters and rate of reaction expressions have been adapted from the literature (Sterret and McIlvried , 1974).

VI. Solution Approach

The set of equations can be easily solved using an explicit method. Fixed-step routines might prove inefficient, and an adaptive integration step algorithm is recommended. The method selected for the case study has been a fourth-order Runge-Kutta method with adaptive integration step. To minimize round-off errors, the algorithm uses the formulas developed by Gill (method frequently referred to as Runge-Kutta-Gill). This is a well-known method, described in most numerical analysis textbooks (cf. Chapra and Canale, 1998). For this example the authors have written their own version, but robust FORTRAN and ANSI C codes can be readily found in several public domain libraries.

The solution to the set of non-linear ordinary differential equations is attained via a FORTRAN program (“reactor.for”). This program can be run as an interactive program or on a batch mode via an input file. The approach chosen here is the use of an input file, "reactor.ini," which defines the reactor operating conditions (cf. Fig. 4), i.e.,

☐ Inlet temperature
☐ Inlet volumetric flow rate
☐ Total Pressure
☐ Inlet butene molar fraction
☐ Dilution ratio (oxygen/butene molar ratio in the feed).

```
open (1, file='reactor.ini')
c    To   : inlet temperature, K
c    Qo   : inlet volumetric flowrate, m^3/hr
c    PT   : total pressure, atm
c    yBo  : inlet molar butene molar fraction
c    beta : dilution ratio, beta = FO2o/FBo
read (1, *, end = 10) To, Qo, PT, yBo, beta
```

Figure 4: Input to FORTRAN program

The student defines these conditions in the worksheet "Reactor" of the spreadsheet. A macro can be recorded to write this data into the input file and execute the FORTRAN program. The program will integrate the reactor governing equations (e.g., $0 \leq \theta \leq 4$) and will write (cf. Fig. 5) the solution (butene, x_B, and oxygen x_{O2}, conversion and the yield for butadiene, Y_{BD}, carbon dioxide, Y_{CO2}, and water, Y_{H2O}).

```
     open (2, file='reactor.out')
c    1: butene conversion
c    2: oxygen conversion
c    3: butadiene yield
c    4: carbon dioxide yield
c    5: water (steam) yield
     volume = time * Tau * Qo
```

```
write (2,*) time, (x(k), k = 1, neq, 1)
```

Figure 5: Output from FORTRAN program

The reactant conversion is defined as:

$$x_i = \frac{F_i}{F_i^o}$$

while yield is defined as:

$$Y_i = \frac{F_i}{F_B^o}$$

The results are written to an output file, "reactor.out," and integrated into the spreadsheet for graphical and economical analysis.

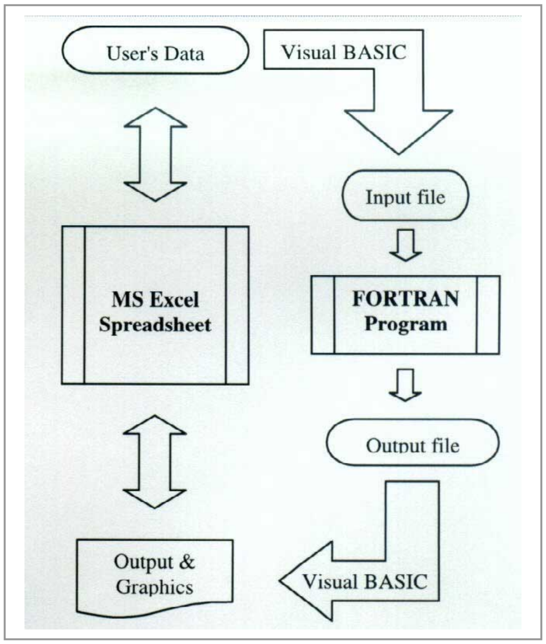


Figure 6: Flow of information between FORTRAN program and Spreadsheet

The flow of information (cf. Fig. 6) between the spreadsheet (in the foreground) and the computer program (in the background) is controlled by a key component: a Visual BASIC macro integrated in the spreadsheet, "SolveReactor" (cf. Fig. 7).

```
Sub SolveReactor()
Reset

’define input file
FileIn = "a:\reactor.ini"

’ read operating conditions from the spreadsheet
Call WriteDataToInputFile(FileIn)
```

```
’ run program
RetVal = Shell("a:\reactor.exe", 1)

’ define output file
FileOut = "a:\reactor.out"

’ process results and transport them to
spreadsheet
Call ReadBackFromOutputFile(FileOut)
End Sub
```

Figure 7: "SolveReactor" Visual BASIC Macro

The two major subroutines in this macro are (cf. Figs. 8 and 9):

WriteDataToInputFile
Reads the values for the operating variables and creates the input for the FORTRAN program

ReadBackFromOutputFile
Reads the output file created by the FORTRAN program and transcribes them to the spreadsheet.

```
Sub WriteDataToInputFile(FileIn)
’ subroutine to coordinate input
’ file with a given worksheet

’ indexes
Dim I As Integer
Dim J As Integer
’ operating  variables
Dim Flow As Single ’Flow rate, m^3/hr
Dim Temp As Single ’Temperature, K
Dim Pres As Single
’Total Pressure, atm
Dim yBo As Single
’butene molar fraction
Dim beta As Single
’dilution ratio FO2o/FBo

’ I/O cells
I = 5
J = 5 + 2 * Cells(11, 3)

’ read data from spreadsheet
Flow = Cells(I, J)
beta = Cells(I + 1, J)
yBo = Cells(I + 2, J)
Temp = Cells(I + 3, J)
Pres = Cells(I + 4, J)

’ write information to input fie
Open FileIn For Output As #1
Write #1, Temp, Flow, Pres, yBo, beta
Close #1
```

Figure 8: Suroutine in "SolveReactor" macro Coordinates Spreadsheet to Program flow

```
Sub ReadBackFromOutputFile(FileOut)
’ subroutine to coordinate output file
’ with a given set of cells in a
’ spreadsheet

’ indexes
Dim I As Integer
Dim J As Integer
Dim K As Integer
’ output variables
Dim time As Single ’time
Dim xB As Single ’butene conversion
Dim xO2 As Single ’oxygen conversion
Dim yB As Single ’yield of butadiene
Dim yCO2 As Single ’ yield of carbon
dioxide
Dim yH2O As Single ’ yield of water

Dim Conversion As Single
Dim Selectivity As Single

’ figure out cells in spreadsheet
J = 5 + 2 * Cells(11, 3)
I = 10

’ read output file
Open FileOut For Input As #1

RowIndex = I
Do Until EOF(1)
    Input #1 time,xB,xO2,yBD,yCO2,yH2O
      RowIndex = RowIndex + 1
      Conversion = xB
  Selectivity = 1
  If xB > 0 Then
      Selectivity = yBD / xB
      Cells(RowIndex,J) = Conversion
      Cells(RowIndex,J+1)=Selectivity
  End If

  If J < 6 Then
    ' optimum conditions … then
    ' store complete solution
    K = 17
    Cells(RowIndex, K) = time
    Cells(RowIndex, K + 1) = xB
    Cells(RowIndex, K + 2) = xO2
    Cells(RowIndex, K + 3) = yBD
    Cells(RowIndex, K + 4) = yCO2
    Cells(RowIndex, K + 5) = yH2O
  End If
Loop
Close #1
End Sub
```

Figure 9: Suroutine in "SolveReactor" macro Coordinates Program to Spreadsheet flow

VII. Results and Discussion

The "tutorial" begins in the worksheet "Reactor," where the student can test the performance, based for instance on the selectivity, of different sets of operating conditions. The trial-and-error process will continue until an optimum set is identified. The results can be easily analyzed in graphical format, the navigation between specific sections of the worksheets can be easily implemented through hyperlinks. (cf. Fig. 10).

Once the optimum reactor operating conditions, the values of the design variables are inserted in the corresponding block ("set #0"), and a macro is used to generate the reactor solution. The "SolveReactor" output generates automatically the columns required for the economical analysis. The student can then proceed to the Economical Analysis (cf. Fig. 11).

VIII. Conclusions

- Students can readily develop their own calculation sheets, leaving the most complicated calculations for the instructor-supplied package.
- Students need to understand the problem that they are trying to solve, basic knowledge of some spreadsheet (in contrast to deep knowledge of programming), and a basic knowledge of numerical methods. The emphasis can then be put on understanding the foundations of the outlined problem, instead of having to pay attention to the solution procedure.
- The pedagogic importance of the use of new available tools has been demonstrated through a concrete example of process synthesis.

IX. Acknowledgements

The authors gratefully acknowledge support from the National University of Tucumán (UNT). One of the authors (JEG) gratefully acknowledges support from the Established Full-time Faculty Research Development (EFFRD) program and the Graduate Council at Cleveland State University (CSU). Financial support from the University Center for Teaching and Learning (UCTL) at CSU and the National University of Tucumán Research Council (CIUNT) are also gratefully acknowledged.

Bibliography

1. Bruzas, A. E. and J. E. Gatica "JAVA-based Interactive Computer Modules for Computer-aided Instruction," *Ohio Space Grant Consortium*, April 1999 (Ohio Aerospace Institute, Cleveland, OH).
2. Chapra, S.C. and R.P. Canale *''Numerical Methods for Engineers,"* 3rd. Ed., McGraw-Hill, New York, NY (1997).
3. Douglas, J.M. *"Conceptual Design of Chemical Processes,"* McGraw-Hill, New York, NY (1988).
4. Harb, J.N., A. Jobes, R.L. Rowley and W.V. Wilding "Use of Computational Tools in Engineering Education," *Chemical Engineering Education*, pp. 180-187 (Summer 1997).
5. Hernández de Colombo, M.R. and J. E. Gatica "Nuevas metas de la Computación Aplicada a la Ingeniería," *Revista de la Facultad de Ciencias Exactas y Tecnología (UNT)*, 6(11), pp. 48-52 (1997).
6. Mitchell, B.S. "Use of Spreadsheets in Introductory Statistics and Probability, *" Chemical Engineering Education*, pp. 194-200 (Summer 1997).
7. Sterret, J.S. and H.G. McIlvried "Kinetics of Oxidative Dehydrogenation of Butene to Butadiene over a Ferrite Catalyst," *Ind. Chem. Process Des. & Develop*., 13(1), pp. 54-58 (1974)
8. FLOWTRAN. Monsanto Company, Engineering Teac. - F4EE, 800 N. Lindbergh Blvd. , St. Louis, Missouri 63166.
9. PROCESS . Simulation Sciences, Inc. , 1440 North Harbour Blvd., Fullerton, California 92635.

10. TK SOLVER . Universal Technical Systems Inc. 1220 Rock Street , Rockford, Illinois 61101.

MAURICIO A. COLOMBO

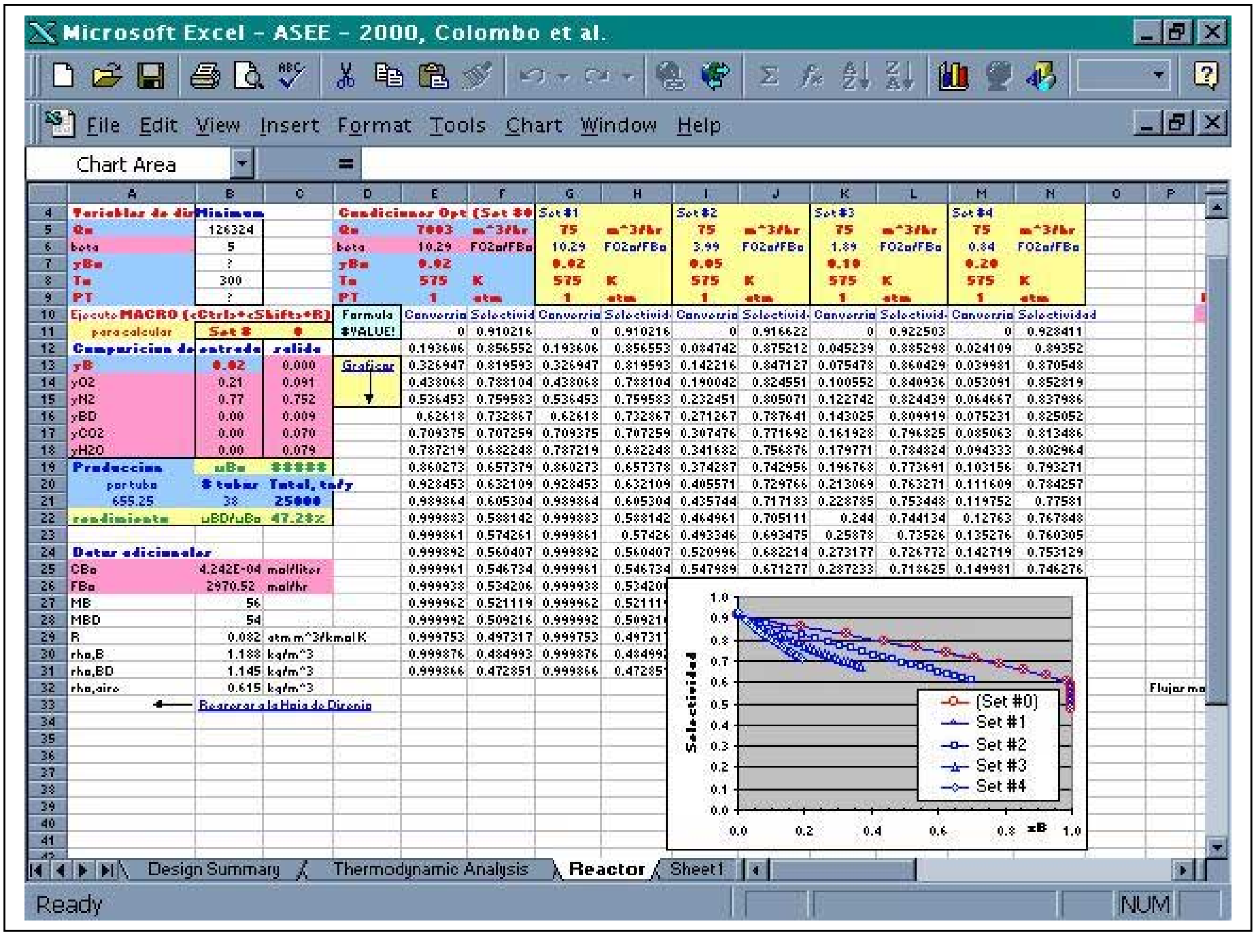
Mauricio Colombo is currently a doctoral student at the National University of Tucumán (UNT, Argentina). He received his B.Sc. in Chemical Engineering from the UNT where he serves as a Lecturer for Process Design and Optimization and for Chemical Thermodynamics in the Chemical Engineering department. Mauricio's interests include new applications of educational software tools to Process Design problems, area in which he has been active for several years. He has numerous presentations and publications on the subject.

MARÍA ROSA HERNÁNDEZ

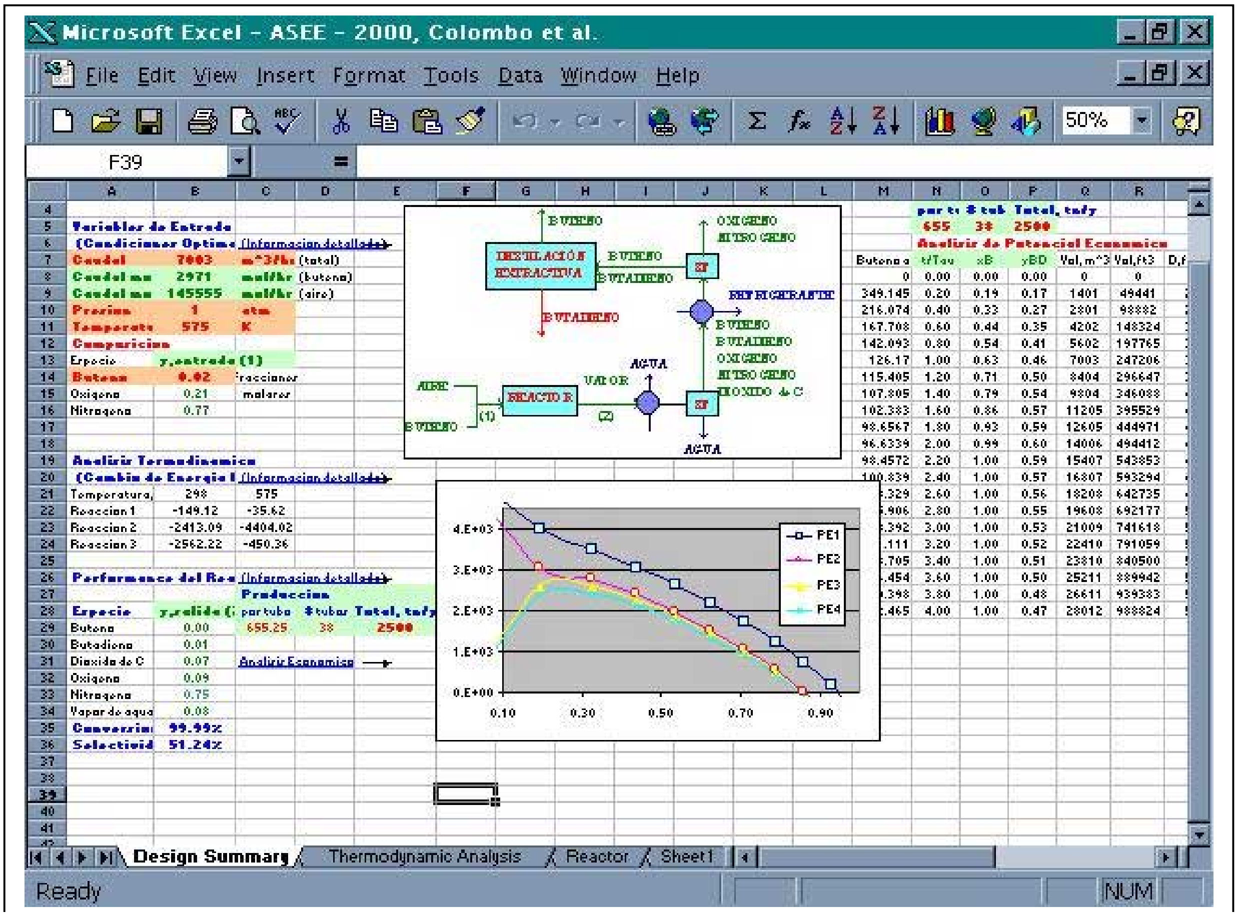
María Rosa Hernández is currently Associate Professor at the National University of Tucumán (UNT, Argentina). Professor Hernández teaches undergraduate core Chemical Engineering courses in Process Design and Optimization and Chemical Thermodynamics. She received her B.Sc. degree from the UNT and her M.Sc. degree from the Southern National University (UNS, Bahía Blanca, Argentina), both in Chemical Engineering. She is the Director of the "Industrial Process Engineering" research program funded by the National University of Tucumán Research Council (CIUNT). Professor Hernández is actively involved in joint research with industry.

JORGE E. GATICA

Jorge Gatica is currently an Associate Professor and Graduate Program coordinator in the Chemical Engineering Department at Cleveland State University (CSU). He has degrees in Chemistry, Applied Statistics, and Chemical Engineering. His current interests include the application of Reaction Engineering principles to materials manufacturing and processing, computers and programming in engineering curricula, and the use of computers in education. A long-term active member of the AIChE Catalysis and Reaction Engineering Division, Dr. Gatica is actively involved in basic and applied research. He has over 70 technical presentations at national and international meetings, and more than 50 publications in refereed journals. Dr. Gatica has taken the lead in several WWW-based developments at CSU, where he has received several Teaching Enhancement Awards in support of computer-aided instruction developments in Engineering. Dr. Gatica serves as a mentor for undergraduate minority and high-school students, as well as a summer advisor for Project SEED students.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	P
4	Variables de dis	Minimum		Condiciones Opt (Set #0			Set #1		Set #2		Set #3		Set #4		
5	Qo	126324		Qo	7003	m^3/hr	75	m^3/hr	75	m^3/hr	75	m^3/hr	75	m^3/hr	
6	beta	5		beta	10.29	FO2a/FBa	10.29	FO2a/FBa	3.99	FO2a/FBa	1.89	FO2a/FBa	0.84	FO2a/FBa	
7	yBo	?		yBo	0.02		0.02		0.05		0.10		0.20		
8	To	300		To	575	K	575	K	575	K	575	K	575	K	
9	PT	?		PT	1	atm	1	atm	1	atm	1	atm	1	atm	
10	Ejecute MACRO (<Ctrl>+<Shift>+R)			Formula	Conversio	Selectivid	Conversio	Selectivid	Conversio	Selectivid	Conversio	Selectivid	Conversio	Selectividad	
11	para calcular	Set #	0	#VALUE!	0	0.910216	0	0.910216	0	0.916622	0	0.922503	0	0.928411	
12	Composicion de	entrada	salida		0.193606	0.856552	0.193606	0.856553	0.084742	0.875212	0.045239	0.885298	0.024109	0.89352	
13	yB	0.02	0.000	Graficar	0.326947	0.819593	0.326947	0.819593	0.142216	0.847127	0.075478	0.860429	0.039981	0.870548	
14	yO2	0.21	0.091		0.438068	0.788104	0.438068	0.788104	0.190042	0.824551	0.100552	0.840936	0.053091	0.852819	
15	yN2	0.77	0.752		0.536453	0.759583	0.536453	0.759583	0.232451	0.805071	0.122742	0.824439	0.064667	0.837986	
16	yBD	0.00	0.009		0.62618	0.732867	0.62618	0.732867	0.271267	0.787641	0.143025	0.809919	0.075231	0.825052	
17	yCO2	0.00	0.070		0.709375	0.707259	0.709375	0.707259	0.307476	0.771692	0.161928	0.796825	0.085063	0.813486	
18	yH2O	0.00	0.079		0.787219	0.682248	0.787219	0.682248	0.341682	0.756876	0.179771	0.784824	0.094333	0.802964	
19	Produccion	uBo	#####		0.860273	0.657379	0.860273	0.657378	0.374287	0.742956	0.196768	0.773691	0.103156	0.793271	
20	por tubo	# tubos	Total, ta/y		0.928453	0.632109	0.928453	0.632109	0.405571	0.729766	0.213069	0.763271	0.111609	0.784257	
21	655.25	38	25000		0.989864	0.605304	0.989864	0.605304	0.435744	0.717183	0.228785	0.753448	0.119752	0.77581	
22	rendimiento	uBD/uBo	47.28%		0.999883	0.588142	0.999883	0.588142	0.464961	0.705111	0.244	0.744134	0.12763	0.767848	
23					0.999861	0.574261	0.999861	0.57426	0.493346	0.693475	0.25878	0.73526	0.135276	0.760305	
24	Datos adicionales				0.999892	0.560407	0.999892	0.560407	0.520996	0.682214	0.273177	0.726772	0.142719	0.753129	
25	CBo	4.242E-04	mol/liter		0.999961	0.546734	0.999961	0.546734	0.547989	0.671277	0.287233	0.718625	0.149981	0.746276	
26	FBo	2970.52	mol/hr		0.999938	0.534206	0.999938	0.53420							
27	MB	56			0.999962	0.521119	0.999962	0.52111							
28	MBD	54			0.999992	0.509216	0.999992	0.50921							
29	R	0.082	atm m^3/kmol K		0.999753	0.497317	0.999753	0.49731							
30	rho,B	1.188	kg/m^3		0.999876	0.484993	0.999876	0.48499							
31	rho,BD	1.145	kg/m^3		0.999866	0.472851	0.999866	0.47285							
32	rho,aire	0.615	kg/m^3												Flujar ma
33	←	Regresar a la Hoja de Diseño													



	A	B	C	D	E
5	Variables de Entrada				
6	(Condiciones Optima		(Informacion detallada) →		
7	Caudal	7003	m^3/hr	(total)	
8	Caudal mo	2971	mol/hr	(buteno)	
9	Caudal mo	145555	mol/hr	(aire)	
10	Presion	1	atm		
11	Temperatu	575	K		
12	Composicion				
13	Especie	y,entrada (1)			
14	Buteno	0.02	fracciones		
15	Oxigeno	0.21	molares		
16	Nitrogeno	0.77			
19	Analisis Termodinamico				
20	(Cambio de Energia l		(Informacion detallada) →		
21	Temperatura,	298	575		
22	Reaccion 1	-149.12	-35.62		
23	Reaccion 2	-2413.09	-4404.02		
24	Reaccion 3	-2562.22	-450.36		
26	Performance del Rea		(Informacion detallada) →		
27			Produccion		
28	Especie	y,salida (:	por tubo	# tubos	Total, ta/y
29	Buteno	0.00	655.25	38	2500
30	Butadieno	0.01			
31	Dioxido de C	0.07	Analisis Economico →		
32	Oxigeno	0.09			
33	Nitrogeno	0.75			
34	Vapor de agua	0.08			
35	Conversion	99.99%			
36	Selectivid	51.24%			

	M	N	O	P	Q	R
4		par tu	# tub	Total, ta/y		
5		655	38	2500		
6		Analisis de Potencial Economico				
7	Buteno a	t/Tau	xB	yBD	Vol, m^3	Vol,ft3
8	0	0.00	0.00	0.00	0	0
9	349.145	0.20	0.19	0.17	1401	49441
10	216.074	0.40	0.33	0.27	2801	98882
11	167.708	0.60	0.44	0.35	4202	148324
12	142.093	0.80	0.54	0.41	5602	197765
13	126.17	1.00	0.63	0.46	7003	247206
14	115.405	1.20	0.71	0.50	8404	296647
15	107.805	1.40	0.79	0.54	9804	346088
16	102.383	1.60	0.86	0.57	11205	395529
17	98.6567	1.80	0.93	0.59	12605	444971
18	96.6339	2.00	0.99	0.60	14006	494412
19	98.4572	2.20	1.00	0.59	15407	543853
20	100.839	2.40	1.00	0.57	16807	593294
21	.329	2.60	1.00	0.56	18208	642735
22	.906	2.80	1.00	0.55	19608	692177
23	.392	3.00	1.00	0.53	21009	741618
24	.111	3.20	1.00	0.52	22410	791059
25	.705	3.40	1.00	0.51	23810	840500
26	.454	3.60	1.00	0.50	25211	889942
27	.398	3.80	1.00	0.48	26611	939383
28	.465	4.00	1.00	0.47	28012	988824