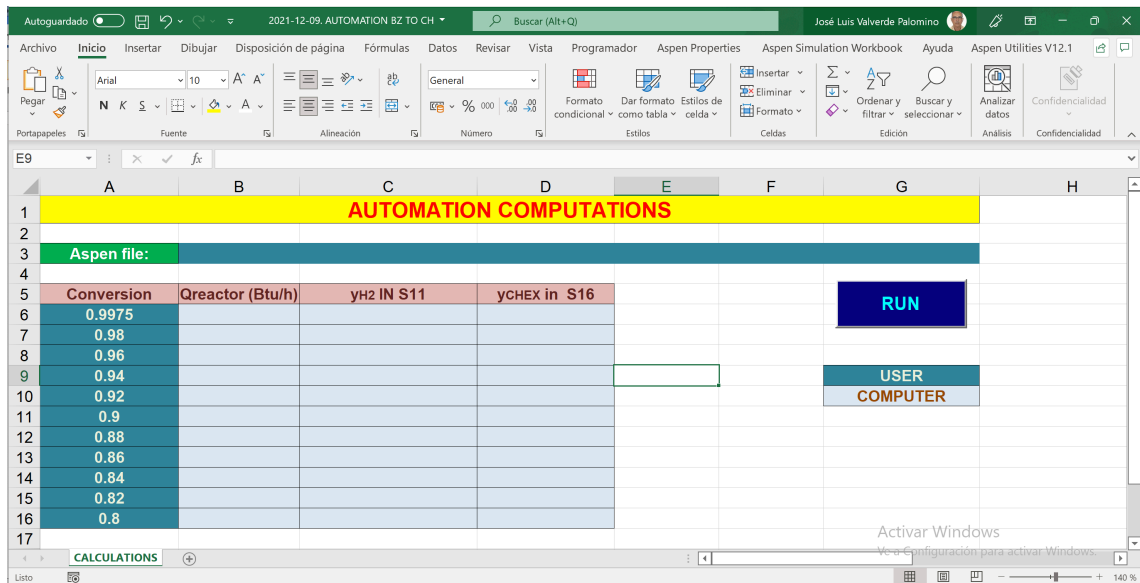
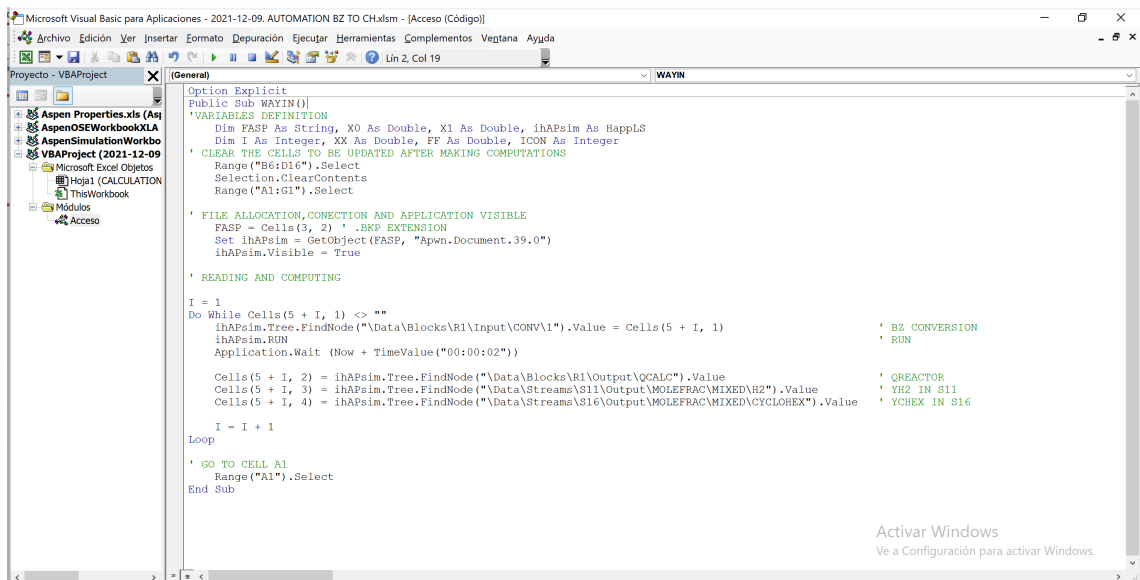


## SUPPLEMENTARY FIGURES



**Figure S1.** Sheet COMPUTATIONS in the MS EXCEL-VBA template used in Case of Study 1.



**Figure S2.** Definition of module Acceso used in Case of Study 1.

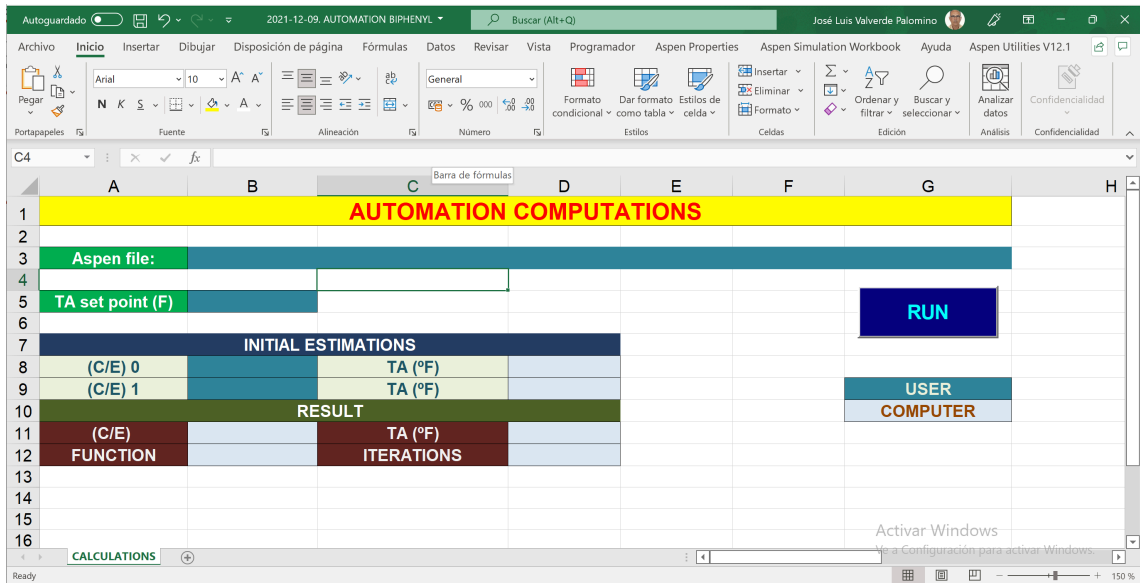


Figure S3. Sheet COMPUTATIONS in the MS EXCEL-VBA template used in Case of Study 2.

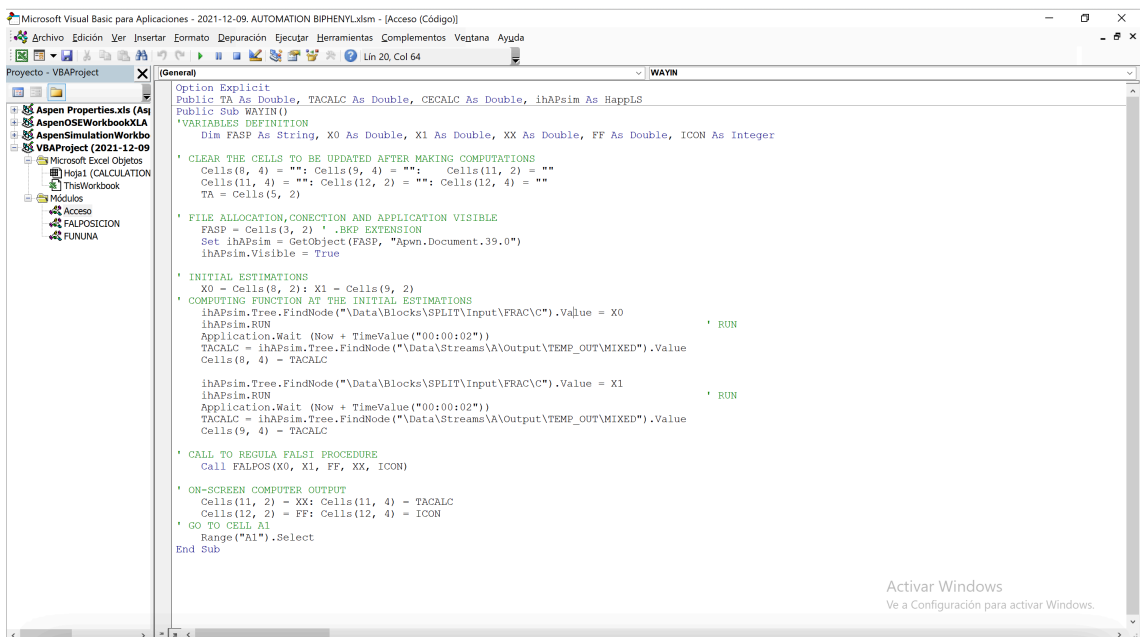


Figure S4. Definition of module Acceso used in Case of Study 2.

```

Microsoft Visual Basic para Aplicaciones - 2021-12-09. AUTOMATION BIPHENYLxism - [FALPOSICION (Código)]
Archivo Edición Ver Insertar Formato Depuración Ejecutar Herramientas Complementos Ventana Ayuda
Proyecto - VBAProject (General) (Declaraciones)
Option Explicit
Option Base 1
Public Sub FALPOS(X0 As Double, X1 As Double, F As Double, X As Double, N As Integer)
' SUBROUTINE FOR EVALUATING THE REGULA FALSI PROCEDURE
' N: NUMBER OF ITERATIONS; X0: INITIAL GUESS 0; X1: INITIAL GUESS 1
' X: ROOT TO BE COMPUTED; F: FUNCTION TO MAKE 0
' *****
' VARIABLES DEFINITION
' *****
Dim TEST As Boolean, F0 As Double, F1 As Double, F2 As Double, X2 As Double
' *****
' PROCESS
' *****
TEST = True
N = 0
Do While (TEST)
N = N + 1
Call FUNCION1(X0, F0)
Call FUNCION1(X1, F1)
X2 = X1 - (X1 - X0) * F1 / (F1 - F0)
If (X2 = 0) Then
X2 = 0.0000000000000001
End If
Call FUNCION1(X2, F2)
If ((Abs((X2 - X0) / X2) < 0.000001) Or _
(Abs((X2 - X1) / X2) < 0.000001)) Then
X = X2
F = F2
TEST = False
Else
If (F2 * F0 < 0) Then
X1 = X2
Else
X0 = X2
End If
End If
If (N > 200) Then
X = X2
F = F2
TEST = False
MsgBox "NO CONVERGE EN 200 ITERACIONES"
End If
Loop
End Sub

```

Figure S5. Definition of module FALPOSICION used in Case of Study 2.

```

Microsoft Visual Basic para Aplicaciones - 2021-12-09. AUTOMATION BIPHENYLxism - [FUNUNA (Código)]
Archivo Edición Ver Insertar Formato Depuración Ejecutar Herramientas Complementos Ventana Ayuda
Proyecto - VBAProject (General) FUNCION1
Option Explicit
Option Base 1
Public Sub FUNCION1(X As Double, F As Double)
ihaPsim.Tree.FindNode("\Data\Blocks\ASPLIT\input\FRAC\C").Value = X
ihaPsim.RUN
Application.Wait (Now + TimeValue("00:00:02"))
TACALC = ihaPsim.Tree.FindNode("\Data\Streams\A\Output\TEMP_OUT\MIXED").Value
F = TA - TACALC
End Sub

```

Figure S6. Definition of module FUNUNA used in Case of Study 2.

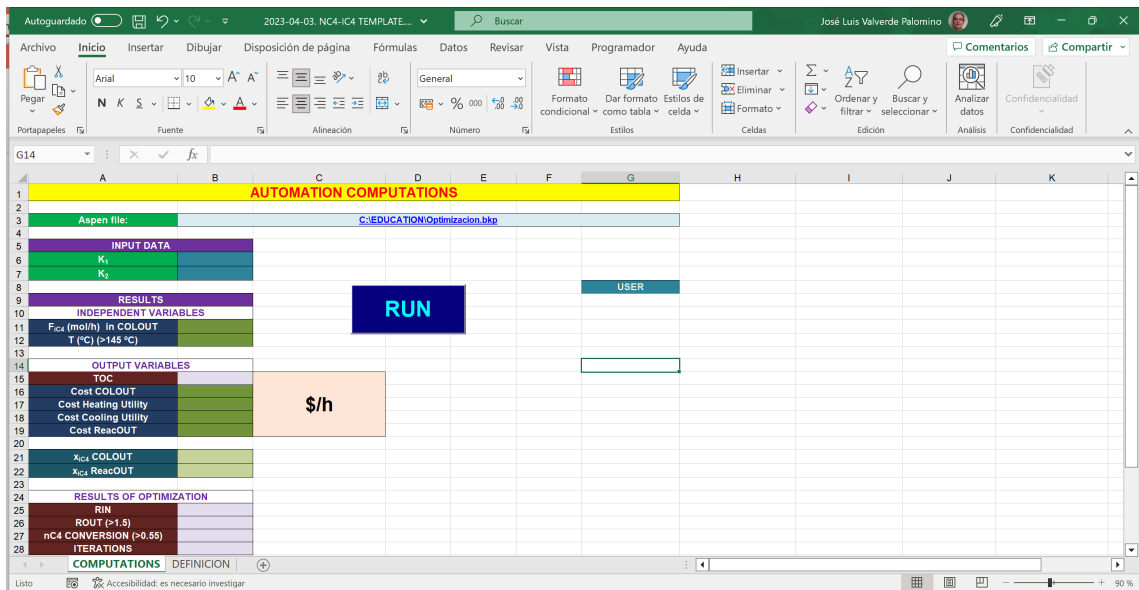


Figure S7. Sheet COMPUTATIONS in the MS EXCEL-VBA template used in Case of Study 3.

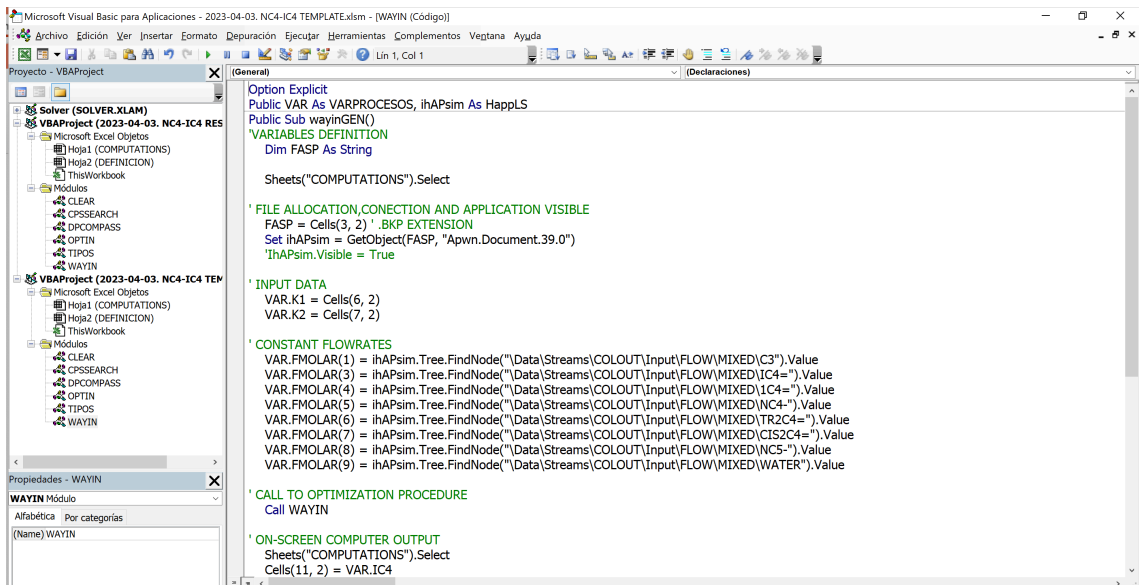


Figure S8. Definition of module WAYIN used in Case of Study 3.

```

VAR FMOLAR(6) = ihAPsim.Tree.FindNode("Data\Streams\COLOUT\input\FLOW\MIXED\TR2C4=").Value
VAR FMOLAR(7) = ihAPsim.Tree.FindNode("Data\Streams\COLOUT\input\FLOW\MIXED\CIS2C4=").Value
VAR FMOLAR(8) = ihAPsim.Tree.FindNode("Data\Streams\COLOUT\input\FLOW\MIXED\NCS-").Value
VAR FMOLAR(9) = ihAPsim.Tree.FindNode("Data\Streams\COLOUT\input\FLOW\MIXED\WATER").Value

' CALL TO OPTIMIZATION PROCEDURE
Call WAYIN

' ON-SCREEN COMPUTER OUTPUT
Sheets("COMPUTATIONS").Select
Cells(11, 2) = VAR.IC4
Cells(12, 2) = VAR.T

Cells(15, 2) = VAR.TOC
Cells(16, 2) = VAR.COLOUT
Cells(17, 2) = VAR.CHEATUTIL
Cells(18, 2) = VAR.CCOOLUTIL
Cells(19, 2) = VAR.CREACOUT
Cells(21, 2) = VAR.XIC4COLOUT
Cells(22, 2) = VAR.XIC4REACOUT

Cells(25, 2) = VAR.R
Cells(26, 2) = VAR.ROUT
Cells(27, 2) = VAR.NC4CONV
Cells(28, 2) = VAR.ITERACIONES

Range("A1").Select
End Sub

```

Figure S8. Definition of module WAYIN used in Case of Study 2 (continuation).

```

Option Explicit

' *****
' MODULO USADO PARA EVLUAR PROGRAMACIÓN LINEAL Y NO LINEAL
' *****
' TPRB. REGISTRO PARA INTRODUCIR DATOS DEL PROBLEMA
' APROX(), APROXIMACIONES INICIALES
' ITER, NÚMERO DE ITERACIONES
' XoBEST(), MEJORES SOLUCIONES ENCONTRADAS POR MARQUARDT

Public Sub COMPASS(tprb As VALENT, delta0 As Double, delat0 As Double, aprox() As Double, ITER As Long, ...
    XOPT() As Double, FOPT As Double)

    Dim I As Double, X() As Double, G() As Double, F As Double, TESTINT As Boolean, FOPTANT As Double
    Dim J As Long, testiter As Boolean, TEST As Boolean, GIGUAL() As Double, FIGUAL As Double
    ReDim X(tprb.VARIND), G(tprb.DESIG), GIGUAL(tprb.IGUAL)

    For I = 1 To tprb.VARIND
        X(I) = aprox(I)
    Next I

    Call FDESIG(X(), G())

    TEST = True

    For I = 1 To tprb.DESIG
        If G(I) > 0 Then
            TEST = False
        End If
    Next I

    ITER = 0
End Sub

```

Figure S9. Definition of module OPTIN used in Case of Study 3.

```

ITER = 0
Call funobj(X(), FOPT)
If tprb.IGUAL > 0 Then Call FIGUALD(X(), GIGUAL())
FIGUAL = 0
For J = 1 To tprb.IGUAL
    FIGUAL = FIGUAL + GIGUAL(J)
Next J
If tprb.MAXMIN = 1 Then ' MÁXIMO
    If FOPT = 0 Then FOPT = 1E-50
    FOPT = 1 / FOPT + (FIGUAL) ^ 2
End If
If tprb.MAXMIN = 2 Then FOPT = FOPT + (FIGUAL) ^ 2 ' MÍNIMO

For J = 1 To tprb.VARIND
    XOPT(J) = X(J)
Next J

Do While TEST And ITER <= 100

    ITER = ITER + 1

    Cells(6, 10) = ITER

    TESTINT = False

For I = 1 To tprb.VARIND
    For J = 1 To tprb.VARIND
        X(J) = aprox(J)
    Next J
    X(I) = aprox(I) + delta0 * aprox(I)
    Call funobj(X(), F)

```

Figure S9. Definition of module OPTIN used in Case of Study 3 (continuation).

```

Call funobj(X(), F)
If tprb.DESIG > 0 Then Call FDESIG(X(), G())
If tprb.IGUAL > 0 Then Call FIGUALD(X(), GIGUAL())
FIGUAL = 0
For J = 1 To tprb.IGUAL
    FIGUAL = FIGUAL + GIGUAL(J)
Next J
tesIter = True
For J = 1 To tprb.DESIG
    If G(J) > 0 Then
        tesIter = False
    End If
Next J
If tesIter Then
    If tprb.MAXMIN = 1 Then ' max
        If F = 0 Then F = 1E-50
        F = 1 / F + (FIGUAL) ^ 2
        If F < FOPT Then
            For J = 1 To tprb.VARIND
                XOPT(J) = X(J)
            Next J
            FOPT = F
            TESTINT = True
        End If
    End If
    If tprb.MAXMIN = 2 Then ' mIN
        F = F + (FIGUAL) ^ 2
        If F < FOPT Then
            For J = 1 To tprb.VARIND
                XOPT(J) = X(J)
            Next J
            FOPT = F
            TESTINT = True
        End If
    End If

```

Figure S9. Definition of module OPTIN used in Case of Study 3 (continuation).

```

TESTINT = True
End If
End If
End If

X(1) = aprox(I) - delta0 * aprox(I)
Call funobj(X(), F)
If tprb.DESIG > 0 Then Call FDESIG(X(), G())
If tprb.IGUAL > 0 Then Call FIGUALD(X(), GIGUAL())
FIGUAL = 0
For J = 1 To tprb.IGUAL
    FIGUAL = FIGUAL + GIGUAL(J)
Next J
testIter = True
For J = 1 To tprb.DESIG
    If G(J) > 0 Then
        testIter = False
    End If
Next J
If testIter Then
    If tprb.MAXMIN = 1 Then ' max
        If F = 0 Then F = 1E-50
        F = 1 / F + (FIGUAL) ^ 2
        If F < FOPT Then
            For J = 1 To tprb.VARIND
                XOPT(J) = X(J)
            Next J
            FOPT = F
            TESTINT = True
        End If
    End If
    If tprb.MAXMIN = 2 Then ' min
        F = F + (FIGUAL) ^ 2

```

Figure S9. Definition of module OPTIN used in Case of Study 3 (continuation).

```

F = F + (FIGUAL) ^ 2
If F < FOPT Then
    For J = 1 To tprb.VARIND
        XOPT(J) = X(J)
    Next J
    FOPT = F
    TESTINT = True
End If
End If
Next I

If TESTINT Then
    For J = 1 To tprb.VARIND
        aprox(J) = XOPT(J)
        Cells(19 + J, 8) = aprox(J)
    Next J
    Else
        delta0 = delta0 / 2
    End If
Cells(7, 10) = delta0

Cells(8, 10) = FOPT

Call funobj(aprox(), FOPT)

Cells(9, 10) = FOPT
FOPTANT = FOPT
If delta0 < deltatol Then
    TEST = False
End If

```

Figure S9. Definition of module OPTIN used in Case of Study 3 (continuation).

```

End If

Loop

For J = 1 To tprb.VARIND
    X(J) = aprox(J)
Next J
Call funobj(X(), FOPT)
VAR.ITERACIONES = ITER

End Sub

```

Figure S9. Definition of module OPTIN used in Case of Study 3 (continuation).

```

Public Sub funobj(X() As Double, F As Double)
' FUNCION OBJETIVO A MINIMIZAR O MAXIMIZAR
' ESCRITURA DE LA FUNCION: X(1), X(2), ..., VARIABLES INDEPENDIENTES Dim X1 As Double, X2 As Double, X3 As Double, X4 As Double, X5 As Double, Y1

Dim XNC4 As Double, T As Double, R As Double, XCONV As Double
Dim I As Integer, SUM As Double, XIC4 As Double, XIC4OUT As Double
Dim XNC4OUT As Double

T = X(2)

If R <= 0 Then R = 0.0000000001
VAR.FMOLAR(2) = X(1)
R = VAR.FMOLAR(2) / VAR.FMOLAR(5)
VAR.IC4 = VAR.FMOLAR(2)
VAR.T = T
VAR.R = R
'XCONV = X(3)
VAR.NC4CONV = XCONV

XIC4OUT = ihAPsim.Tree.FindNode("Data\Streams\REACTOUT\Output\MOLEFRAC\MIXED\IC4-").Value
XNC4OUT = ihAPsim.Tree.FindNode("Data\Streams\REACTOUT\Output\MOLEFRAC\MIXED\NC4-").Value
VAR.XIC4REACOUT = XIC4OUT
SUM = 0
For I = 1 To 9
    SUM = SUM + VAR.FMOLAR(I)
Next I
XNC4 = VAR.FMOLAR(5) / SUM
XIC4 = VAR.FMOLAR(2) / SUM
VAR.XIC4COLOUT = XIC4

' INDEPENDENT VARIABLES

```

Figure S10. Definition of module DPCOMPASS used in Case of Study 3.



```

(General)
' INDEPENDENT VARIABLES
ihAPsim.Tree.FindNode("Data\Streams\COLOUT\Input\FLOW\MIXED\IC4-").Value = VAR.FMOLAR(2)
ihAPsim.Tree.FindNode("Data\Blocks\REACTOR\Input\TEMP").Value = T
ihAPsim.Tree.FindNode("Data\Blocks\HEATER\Input\TEMP").Value = T
ihAPsim.Tree.FindNode("Data\Streams\COLOUT\Input\FLOW\MIXED\NC4-").Value = VAR.FMOLAR(5)

' CONVERSION
XCONV = 0.8455006 - 0.00109878 * T - 0.13784487 * R - 0.00118546 * T * R
If XCONV < 0 Then XCONV = 1E-40
ihAPsim.Tree.FindNode("Data\Blocks\REACTOR\Input\CONV\1").Value = XCONV
VAR_NC4CONV = XCONV
ihAPsim.RUN
Application.Wait (Now + TimeValue("00:00:01"))

' COSTS
If XIC4 <= 0 Then XIC4 = 1E-20
If XNC4OUT <= 0 Then XNC4OUT = 1E-20
VAR.CCOOLUT = VAR.K1 * 26.39014 * XIC4 ^ (-0.28834)
VAR.CREACOUT = VAR.K2 * (-44.44096 * XNC4OUT ^ 2 - 0.00099 * XNC4OUT + 41.06606)
VAR.CCOOLUTIL = ihAPsim.Tree.FindNode("Data\Results Summary\Utility-Sum\Output\TOT_COOL_COST").Value
VAR.CHEATUTIL = ihAPsim.Tree.FindNode("Data\Results Summary\Utility-Sum\Output\TOT_HEAT_COST").Value

F = VAR.COLOUT - VAR.CREACOUT + VAR.CCOOLUTIL + VAR.CHEATUTIL
VAR.TOC = F

ihAPsim.RUN
Application.Wait (Now + TimeValue("00:00:01"))

End Sub

```

Figure S10. Definition of module DPCOMPASS used in Case of Study 3 (continuation).

```

Microsoft Visual Basic para Aplicaciones - 2023-04-03. NC4-IC4 TEMPLATE.xlsm - [DPCOMPASS (Código)]
Public Sub FDESIG(X() As Double, G() As Double)

' DESIGUALDADES
' ESCRITURA DE LAS DESIGUALDADES. LA FUNCIÓN SIEMPRE HA DE QUEDAR <= 0

Dim XNC4 As Double, T As Double, R As Double, XCONV As Double
Dim I As Integer, SUM As Double, F As Double, XIC4 As Double, XIC4OUT As Double
Dim XNC4OUT As Double

T = X(2)

If R <= 0 Then R = 0.0000000001
VAR.FMOLAR(2) = X(1)
R = VAR.FMOLAR(2) / VAR.FMOLAR(5)
VAR.IC4 = VAR.FMOLAR(2)
'XCONV = X(3)
VAR_NC4CONV = XCONV

VAR.T = T
VAR.R = R

XIC4OUT = ihAPsim.Tree.FindNode("Data\Streams\REACTOUT\Output\MOLEFRAC\MIXED\IC4-").Value
XNC4OUT = ihAPsim.Tree.FindNode("Data\Streams\REACTOUT\Output\MOLEFRAC\MIXED\NC4-").Value
VAR.ROUT = XIC4OUT / XNC4OUT
VAR.XIC4REACOUT = XIC4OUT
SUM = 0
For I = 1 To 9
    SUM = SUM + VAR.FMOLAR(I)
Next I
XNC4 = VAR.FMOLAR(5) / SUM

```

Figure S10. Definition of module DPCOMPASS used in Case of Study 3 (continuation).

```

Microsoft Visual Basic para Aplicaciones - 2023-04-03. NC4-IC4 TEMPLATE.xlsm - [DPCOMPASS (Código)]
Archivo Edición Ver Insertar Formato Depuración Ejecutar Herramientas Complementos Vegetana Ayuda
Proyecto - VBAProject (General) FDESIG Lin 141, Col 51
XNC4 = VAR.FMOLAR(5) / SUM: XIC4 = VAR.FMOLAR(2) / SUM: VAR.XIC4COLOUT = XIC4

' INDEPENDENT VARIABLES
ihAPsim.Tree.FindNode("\Data\Streams\COLOUT\Input\FLOW\MIXED\IC4-").Value = VAR.FMOLAR(2)
ihAPsim.Tree.FindNode("\Data\Blocks\REACTOR\Input\TEMP").Value = T
ihAPsim.Tree.FindNode("\Data\Blocks\HEATER\Input\TEMP").Value = T
ihAPsim.Tree.FindNode("\Data\Streams\COLOUT\Input\FLOW\MIXED\NC4-").Value = VAR.FMOLAR(5)

' CONVERSION
XCONV = 0.8455006 - 0.00109878 * T - 0.13784487 * R - 0.00118546 * T * R
If XCONV < 0 Then XCONV = 1E-40
ihAPsim.Tree.FindNode("\Data\Blocks\REACTOR\Input\CONV\1").Value = XCONV
VAR.NC4CONV = XCONV
ihAPsim.RUN
Application.Wait (Now + TimeValue("00:00:01")) ' RUN

' COSTS
If XIC4 <= 0 Then XIC4 = 1E-20
If XNC4OUT <= 0 Then XNC4OUT = 1E-20
VAR.COLOUT = VAR.K1 * 26.39014 * XIC4 ^ (-0.28834)
VAR.CREACOUT = VAR.K2 * (-44.44096 * XNC4OUT ^ 2 - 0.00099 * XNC4OUT + 41.06606)
VAR.CCOOLUTIL = ihAPsim.Tree.FindNode("\Data\Results Summary\Utility-Sum\Output\TOT_COOL_COST").Value
VAR.CHEATUTIL = ihAPsim.Tree.FindNode("\Data\Results Summary\Utility-Sum\Output\TOT_HEAT_COST").Value

F = VAR.COLOUT - VAR.CREACOUT + VAR.CCOOLUTIL + VAR.CHEATUTIL

G(1) = -F
G(2) = 0.55 - XCONV
G(3) = 1.5 - VAR.ROUT
G(4) = 145 - T
ihAPsim.RUN
Application.Wait (Now + TimeValue("00:00:01")) ' RUN
End Sub

```

**Figure S10.** Definition of module DPCOMPASS used in Case of Study 3 (continuation).