REFERENCES

Asante, N.D.K., and Zhu, X.X. (1996). An automated approach for heat exchanger network retrofit featuring minimal topology modifications. Computers and Chemical Engineering, 20, S7-S12.

Asante, N.D.K., and Zhu, X.X. (1999). Diagnosis and optimization approach for heat exchanger network retrofit. American Institute of Chemical Engineers, 45(7), 1488-1503.

Athier, G., Floquet, P., Pibouleau, L., and Domenech S. (1998). A Mixed Method For Retrofitting Heat-Exchanger Networks. Computers and Chemical Engineering, 22, S505-S511.

Bagajewicz, M.J., Valtinson, G., and Nguyen, D.Q. (2013). Retrofit of crude units preheating trains: mathematical programming versus pinch technology. Industrial and Engineering Chemistry Research, 52, 14913−14926.

Barbaro, A., and Bagajewicz, M.J. (2005). New rigorous one-step MILP formulation for heat exchanger network synthesis. Computers and Chemical Engineering, 29, 1945–1976.

Björk, K.M, and Nordman, R. (2005). Solving large-scale retrofit heat exchanger network synthesis problems with mathematical optimization methods. Chemical Engineering and Processing, 44, 869–876.

Björk, K.M., and Westerlund, T. (2002). Global optimization of heat exchanger network synthesis problems with and without the isothermal mixing assumption. Computers and Chemical Engineering, 26, 1581-1593.

Bogataj, M., and Bagajewicz, M.J. (2008). Synthesis of non-isothermal heat integrated water networks in chemical processes. Computers and Chemical Engineering, 32, 3130–3142.

Brandt, C., Fieg, G., and Engle, O. (2012). New method for large-scale heat exchanger network synthesis. Proceedings of the 11th International Symposium on Process Systems Engineering, 695-699.

Bulatov, I., Smith, R., and Pan, M. (2013). A novel optimization approach of improving energy recovery in retrofitting heat exchanger network with exchanger details. Energy, 57, 188-200.

Ciric, A.R., and Floudas, C.A. (1990). A Mixed Integer Nonlinear Programming Model for Retrofitting Heat-Exchanger Networks. Industrial and Engineering Chemistry Research, 29, 239-251.

Grossmann, I.E., and Carnegie Mellon University. (1992). Mathematical Methods for Heat Exchanger Network Synthesis.Engineering Design Research Center,Department of Chemical Engineering.Paper 187.

Hasan, M.M. F., Jayaraman, G., and Karimi, I.A. (2010). Synthesis of heat exchanger networks with non-isothermal phase changes. American Institute of Chemical Engineers, 56(4), 930-945.

Hostrup, M., Gani, R., Kravanja, Z., Sorsak, A., and Grossmann, I. (2001). Integration of thermodynamic insights and MINLP optimization for the synthesis, design and analysis of process flowsheets. Computers and Chemical Engineering, 25, 73-83.

Huang, K.F., Al-mutairi, E.M., and Karimi, I.A. (2012). Heat exchanger network synthesis using a stagewise superstructure with non-isothermal mixing. Chemical Engineering Science, 73, 30-43.

Isafiade, A.J., and Fraser, D.M. (2008). Interval-based MINLP superstructure synthesis of heat exchanger networks. Chemical Engineering Research and Design, 86, 245-257.

Jiang, N., Shelley, J.D., Doyle, S., and Smith, R. (2014). Heat exchanger network retrofit with a fixed network structure. Applied Energy, 127, 25-33.

Jongsuwat, P., Suriyapraphadilok, U., and Bagajewicz, M. (2014). New heat exchanger network design model. Chemical Engineering Transactions, 39, 121-126.

Karimi, I.A., and Huang, K.F. (2012). Heat exchanger network synthesis using a hyperstructure of stagewise stream superstructures. Proceedings of the 11th International Symposium on Process Systems Engineering, 1552-1556.

Karimi, I.A., and Huang, K.F. (2013). Simultaneous synthesis approaches for cost-effective heat exchanger networks. Chemical Engineering Science, 98, 231–245.

Karimi, I.A., and Huang, K.F. (2014). Efficient algorithm for simultaneous synthesis of heat exchanger networks. Chemical Engineering Science, 105. 53-68.

Kralj, A.K. (2010). Optimization of an industrial retrofitted heat exchanger network, using a stage-wise model. Energy, 35, 4748-4753.

Liang, Z., Chao, Y.H., and Zhaoyi, H. (2013). Simultaneous synthesis of heat exchanger network with the non-isothermal mixing. International Journal of Low-Carbon Technologies Advance Access, 1-8.

Liu, X., Luo, X., and Ma, H. (2014). Studies on the retrofit of heat exchanger network based on the hybrid genetic algorithm. Applied Thermal Engineering, 62, 785-790.

Ma, K.L., Hui, C.W., and Yee, T.F. (2000). Constant approach temperature model for HEN retrofit. Applied Thermal Engineering, 20, 1505-1533.

Nguyen, D.Q., Barbaro, A., Vipanurat, N., and Bagajewicz, M.J. (2010). All-at-once and step-wise detailed retrofit of heat exchanger networks using an MILP model. Industrial and Engineering Chemistry Research, 49, 6080–6103.

Ponce-Ortega, J.M., Jiménez-Gutiérrez, A. and Grossmann, I.E. (2008). Simultaneous retrofit and heat integration of chemical process. Industrial and Engineering Chemistry Research, 47, 5512–5528.

Siemanond, K., and Kosol, S. (2012). Heat Exchanger Network Retrofit by Pinch Design Method using Stage-Model Mathematical Programming. Chemical Engineering Transactions, 29, 367-372.

Sieniutycz, S., and Jez owski, J. (2009). Energy Optimization in Process Systems, Elsevier: Amsterdam, 2009.

Smith, R., Jobson, M. and Chen, L. (2010). Recent development in retrofit of heat exchanger networks. Applied Thermal Engineering, 30, 2281-2289.

Sreepathi, B.K., and Rangaiah, G.P. (2014). Improved heat exchanger network retrofitting using exchanger reassignment strategies and multi-objective optimization. Energy, 67, 584-594.

Wan Alwi, S.R, Manan, Z.A., and Nam, S.K. (2012). A New Method To Determine The Optimum Heat Exchanger Network Approach Temperature. 11th International Symposium on Process Systems Engineering, 190-194.

Yee, T.F., and Grossmann, I.E. (1990). Simultaneous optimization models for heat integration. II. Heat exchanger network synthesis. Computers and Chemical Engineering, 14(10), 1165-1184.

Yee, T.F., Grossmann, I.E. and Carnegie Mellon University.Engineering Design Research Center., (1988). A screening and optimization approach for the retrofit of heat exchanger networks. Department of Chemical Engineering, 133.

Yee, T.F., Grossmann, I.E., and Kravanja, Z. (1990). Simultaneous optimization models for heat integration. I. Area and energy targeting and modeling of multi-stream exchangers. Computers and Chemical Engineering, 14(10), 1151-1164.

Yee, T.F., Grossmann, I.E., and Kravanja, Z. (1990). Simultaneous optimization models for heat integration. III. Process and heat exchanger network optimization. Computers and Chemical Engineering, 14(10), 1185-1200.

Yerramsetty, K.M., and Murty, C.V.S. (2008). Synthesis of cost-optimal heat exchanger networks using differential evolution. Computers and Chemical Engineering, 32, 1861–1876.

APPENDICES

Appendix A The intermediate results of each step for case study 2

1st step

- thi(I,K)

	KFIRST	K2	K3	KLAST
I1	155.000	105.320	105.320	30.000
I2	80.000	80.000	80.000	40.000
I3	200.000	176.415	40.000	40.000

- thpi(I,J,K)

	KFIRST	K2	K3
I1.J1	105.324	105.320	30.003
I1.J2	20.000	20.000	105.320
I2.J1	80.000	80.000	36.568
I2.J2	26.844	80.000	43.370
I3.J1	20.000	46.404	40.000
I3.J2	176.416	20.000	20.000

- tc(J,K)

	KFIRST	K2	K3	KLAST
J1	160.000	140.128	66.254	20.000
J2	100.000	76.415	38.498	20.000

- tcpi(I,J,K)

	KFIRST	K2	K3
I1.J1	160.000	66.254	68.753
I1.J2	76.415	38.498	20.000
I2.J1	140.128	66.254	62.212
I2.J2	76.415	38.498	38.498
I3.J1	140.128	140.128	20.000
I3.J2	100.000	76.415	20.000

- ai(I,J,K)

	KFIRST	K2	K3
I1.J1	4.973	0	5.801
I2.J1	0	0	3.549
I2.J2	0	0	3.467
I3.J1	0	12.758	0
I3.J2	1.459	3.592	0

- acui(I)

(ALL 0.000)

- ahui(J)

(ALL 0.000)

- qi(I,J,K)

	KFIRST	K2	K3
I1.J1	397.443	0	602.557
I2.J1	0	0	322.526
I2.J2	0	0	277.474
I3.J1	0	1477.474	0
I3.J2	353.779	568.747	0

- qcu.(I)

(ALL 0.000)

- qhu(J)

(ALL 0.000)

- zi(I,J,K)

	KFIRST	K2	K3
I1.J1	0.397	0	0.603
I2.J1	0	0	0.538
I2.J2	0	0	0.462
I3.J1	0	0.616	0
I3.J2	0.295	0.474	0

- zcui(I)

(ALL 0.000)

- zhui(J)

(ALL 0.000)

2nd step

- th(I,K)

	KFIRST	K2	K3	KLAST
I1	155.000	44.425	44.425	30.000
I2	80.000	80.000	80.000	40.000
I3	200.000	120.000	40.000	40.000

- thp(I,K,BH,SK)

INDEX 1 = I1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	155.000	55.639	44.425	44.425
KFIRST.BH2	155.000	62.694	62.694	62.694
K2 .BH1	44.425	44.425	44.425	44.425
K2 .BH2	44.425	20.000	20.000	20.000
K3 .BH1	44.425	20.000	20.000	20.000
K3 .BH2	44.425	30.000	30.000	30.000

INDEX 1 = I2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	80.000	68.164	45.261	45.261
KFIRST.BH2	80.000	47.500	47.500	47.500
K2 .BH1	80.000	23.468	23.468	23.468
K2 .BH2	80.000	23.468	23.468	23.468
K3 .BH1	80.000	52.680	40.000	40.000

K3 .BH2	80.000	20.000	20.000	20.000

INDEX 1 = I3

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	200.000	194.431	21.791	21.791
KFIRST.BH2	200.000	147.570	130.949	130.949
K2 .BH1	120.000	20.000	20.000	20.000
K2 .BH2	120.000	46.667	46.667	46.667
K3 .BH1	40.000	20.000	20.000	20.000
K3 .BH2	40.000	20.000	20.000	20.000

- tc(J,K)

	KFIRST	K2	K3	KLAST
J1	160.000	115.770	55.770	20.000
J2	100.000	20.000	20.000	20.000

- tcp(J,K,BC,SK)

INDEX 1 = J1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	160.000	115.770	115.770	115.770
KFIRST.BC2	160.000	115.770	115.770	115.770
KFIRST.BC3	160.000	123.915	115.770	115.770
K2 .BC1	115.770	55.770	55.770	55.770
K2 .BC2	115.770	33.020	33.020	33.020
K2 .BC3	115.770	32.856	32.856	32.856
K3 .BC1	55.770	20.000	20.000	20.000

K3 .BC2	55.770	29.937	20.000	20.000
K3 .BC3	55.770	55.770	20.000	20.000

INDEX 1 = J2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	100.000	97.500	20.000	20.000
KFIRST.BC2	100.000	51.042	20.000	20.000
KFIRST.BC3	100.000	20.000	20.000	20.000
K2 .BC1	20.000	20.000	20.000	20.000
K2 .BC2	20.000	20.000	20.000	20.000
K2 .BC3	20.000	20.000	20.000	20.000
K3 .BC1	20.000	20.000	20.000	20.000
K3 .BC2	20.000	20.000	20.000	20.000
K3 .BC3	20.000	20.000	20.000	20.000

- a(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = KFIRST

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	4.973	4.973	4.973	4.973
BH1.BC2	4.973	4.973	4.973	4.973
BH1.BC3	4.973	4.973	4.973	4.973
BH2.BC1	4.973	4.973	4.973	4.973
BH2.BC2	4.973	4.973	4.973	4.973
BH2.BC3	4.973	4.973	4.973	4.973

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K3

	SKFIRST	SK2	SK3	SKLAST

BH1.BC1	5.801	5.801	5.801	5.801
BH1.BC2	5.801	5.801	5.801	5.801
BH1.BC3	5.801	5.801	5.801	5.801
BH2.BC1	5.801	5.801	5.801	5.801
BH2.BC2	5.801	5.801	5.801	5.801
BH2.BC3	5.801	5.801	5.801	5.801

INDEX 1 = I2 INDEX 2 = J1 INDEX 3 = K3

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	3.549	3.549	3.549	3.549
BH1.BC2	3.549	3.549	3.549	3.549
BH1.BC3	3.549	3.549	3.549	3.549
BH2.BC1	3.549	3.549	3.549	3.549
BH2.BC2	3.549	3.549	3.549	3.549
BH2.BC3	3.549	3.549	3.549	3.549

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	3.467	3.467	3.467	3.467
BH1.BC2	3.467	3.467	3.467	3.467
BH1.BC3	3.467	3.467	3.467	3.467
BH2.BC1	3.467	3.467	3.467	3.467
BH2.BC2	3.467	3.467	3.467	3.467
BH2.BC3	3.467	3.467	3.467	3.467

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	12.758	12.758	12.758	12.758

BH1.BC2	12.758	12.758	12.758	12.758
BH1.BC3	12.758	12.758	12.758	12.758
BH2.BC1	12.758	12.758	12.758	12.758
BH2.BC2	12.758	12.758	12.758	12.758
BH2.BC3	12.758	12.758	12.758	12.758

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = KFIRST

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	1.459	1.459	1.459	1.459
BH1.BC2	1.459	1.459	1.459	1.459
BH1.BC3	1.459	1.459	1.459	1.459
BH2.BC1	1.459	1.459	1.459	1.459
BH2.BC2	1.459	1.459	1.459	1.459
BH2.BC3	1.459	1.459	1.459	1.459

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

	SKFIRST	SK2	SK3	SKLAST
BH1.BC1	3.592	3.592	3.592	3.592
BH1.BC2	3.592	3.592	3.592	3.592
BH1.BC3	3.592	3.592	3.592	3.592
BH2.BC1	3.592	3.592	3.592	3.592
BH2.BC2	3.592	3.592	3.592	3.592
BH2.BC3	3.592	3.592	3.592	3.592

- acu(I)

(ALL 0.000)

- ahu(J)

(ALL 0.000)

- q(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = KFIRST

	SKFIRST	SK2
BH1.BC1	397.443	0
BH1.BC3	397.443	89.715

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K3

	SKFIRST
BH2.BC1	115.398

INDEX 1 = I2 INDEX 2 = J1 INDEX 3 = K3

	SKFIRST	SK2
BH1.BC1	101.427	0
BH1.BC2	308.372	118.617
BH1.BC3	0	71.584

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST
BH1.BC1	375.000
BH2.BC1	825.000

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = KFIRST

	SKFIRST	SK2
BH1.BC1	8.379	259.753
BH2.BC2	353.779	224.310
BH2.BC3	353.779	0

- qcu(I)

(ALL 0.000)

- qhu(J)

(ALL 0.000)

- fhp(I,K,BH)

	BH1	BH2
I1.KFIRST	8.000	0
I1.K2	8.000	0
I1.K3	0	8.000
I2.K3	15.000	0
I3.KFIRST	1.505	13.495
I3.K2	3.750	11.250

- fcp(J,K,BC)

	BC1	BC2	BC3
J1.KFIRST	8.986	0	11.014
J1.K2	20.000	0	0
J1.K3	6.062	11.937	2.001
J2.KFIRST	3.352	7.226	4.422
J2.K2	6.133	0	8.867
J2.K3	5.690	4.655	4.655

3rd step

- th(I,K)

	KFIRST	K2	K3	KLAST
I1	155.000	155.000	30.000	30.000
I2	80.000	80.000	80.000	40.000
I3	200.000	200.000	40.000	40.000

- thp(I,K,BH,SK)

INDEX 1 = I1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	155.000	155.000	155.000	155.000
KFIRST.BH2	155.000	155.000	155.000	155.000
K2 .BH1	155.000	155.000	30.000	30.000
K2 .BH2	155.000	155.000	155.000	155.000
K3 .BH1	30.000	30.000	30.000	30.000
K3 .BH2	30.000	30.000	30.000	30.000

INDEX 1 = I2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	80.000	80.000	80.000	80.000
KFIRST.BH2	80.000	80.000	80.000	80.000
K2 .BH1	80.000	80.000	80.000	80.000
K2 .BH2	80.000	80.000	80.000	80.000
K3 .BH1	80.000	80.000	80.000	40.000
K3 .BH2	80.000	80.000	80.000	80.000

INDEX 1 = I3

	SKFIRST	SK2	SK3	SKLAST

KFIRST.BH1	200.000	200.000	200.000	200.000
KFIRST.BH2	200.000	200.000	200.000	200.000
K2 .BH1	200.000	154.482	40.000	40.000
K2 .BH2	200.000	131.429	40.000	40.000
K3 .BH1	40.000	40.000	40.000	40.000
K3 .BH2	40.000	40.000	40.000	40.000

- tc(J,K)

	KFIRST	K2	K3	KLAST
J1	160.000	160.000	20.000	20.000
J2	100.000	100.000	60.000	20.000

- tcp(J,K,BC,SK)

INDEX 1 = J1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	160.000	160.000	160.000	160.000
KFIRST.BC2	160.000	20.000	20.000	20.000
KFIRST.BC3	160.000	160.000	160.000	160.000
K2 .BC1	160.000	121.429	20.000	20.000
K2 .BC2	160.000	20.000	20.000	20.000
K2 .BC3	160.000	20.000	20.000	20.000
K3 .BC1	20.000	20.000	20.000	20.000
K3 .BC2	20.000	20.000	20.000	20.000
K3 .BC3	20.000	20.000	20.000	20.000

INDEX 1 = J2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	100.000	100.000	100.000	100.000
KFIRST.BC2	100.000	100.000	100.000	100.000
KFIRST.BC3	100.000	100.000	100.000	100.000
K2 .BC1	100.000	100.000	30.000	30.000
K2 .BC2	100.000	20.000	20.000	20.000
K2 .BC3	100.000	80.750	80.750	80.750
K3 .BC1	60.000	60.000	60.000	20.000
K3 .BC2	60.000	60.000	60.000	20.000
K3 .BC3	60.000	60.000	60.000	20.000

- q(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

	SK2
BH1.BC1	1000.000

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

	SK3
BH1.BC1	227.597
BH1.BC2	186.202
BH1.BC3	186.202

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST	SK2
BH2.BC1	771.429	1028.571

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

	SKFIRST	SK2

BH1.BC1	0	429.306
BH1.BC3	170.694	0

- qcu(I)

(ALL 0.000)

- qhu(J)

(ALL 0.000)

- z(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

	SK2
BH1.BC1	1.000

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

	SK3
BH1.BC1	1.000
BH1.BC2	1.000
BH1.BC3	1.000

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST	SK2
BH2.BC1	1.000	1.000

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

	SKFIRST	SK2
BH1.BC1	0	1.000
BH1.BC3	1.000	0

- zcu(I)

(ALL 0.000)

- zhu(J)

(ALL 0.000)

- a(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

SK2

BH1.BC1 99.993

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

SK3

BH1.BC1 22.758

BH1.BC2 18.619

BH1.BC3 18.619

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

SKFIRST SK2

BH2.BC1 77.138 102.850

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

SKFIRST SK2

BH1.BC1 0 42.928

BH1.BC3 17.068 0

- acu(I)

(ALL 0.000)

- ahu(J)

(ALL 0.000)

4th step

- th(I,K)

	KFIRST	K2	K3	KLAST
I1	155.000	155.000	30.000	30.000
I2	80.000	80.000	80.000	40.000
I3	200.000	200.000	40.000	40.000

- thp(I,K,BH,SK)

INDEX 1 = I1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	155.000	155.000	155.000	155.000
KFIRST.BH2	155.000	155.000	155.000	155.000
K2 .BH1	155.000	155.000	30.000	30.000
K2 .BH2	155.000	155.000	155.000	155.000
K3 .BH1	30.000	30.000	30.000	30.000
K3 .BH2	30.000	30.000	30.000	30.000

INDEX 1 = I2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	80.000	80.000	80.000	80.000
KFIRST.BH2	80.000	80.000	80.000	80.000
K2 .BH1	80.000	80.000	80.000	80.000
K2 .BH2	80.000	80.000	80.000	80.000

K3 .BH1	80.000	80.000	80.000	40.000
K3 .BH2	80.000	80.000	80.000	80.000

INDEX 1 = I3

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	200.000	200.000	200.000	200.000
KFIRST.BH2	200.000	200.000	200.000	200.000
K2 .BH1	200.000	200.000	30.037	30.037
K2 .BH2	200.000	149.036	43.066	43.066
K3 .BH1	40.000	40.000	40.000	40.000
K3 .BH2	40.000	40.000	40.000	40.000

- tc(J,K)

	KFIRST	K2	K3	KLAST
J1	160.000	160.000	20.000	20.000
J2	100.000	100.000	60.000	20.000

- tcp(J,K,BC,SK)

INDEX 1 = J1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	160.000	160.000	160.000	160.000
KFIRST.BC2	160.000	20.000	20.000	20.000
KFIRST.BC3	160.000	160.000	160.000	160.000
K2 .BC1	160.000	130.773	20.000	20.000
K2 .BC2	160.000	20.000	20.000	20.000
K2 .BC3	160.000	20.000	20.000	20.000

K3 .BC1	20.000	20.000	20.000	20.000
K3 .BC2	20.000	20.000	20.000	20.000
K3 .BC3	20.000	20.000	20.000	20.000

INDEX 1 = J2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	100.000	100.000	100.000	100.000
KFIRST.BC2	100.000	100.000	100.000	100.000
KFIRST.BC3	100.000	100.000	100.000	100.000
K2 .BC1	100.000	100.000	20.000	20.000
K2 .BC2	100.000	20.000	20.000	20.000
K2 .BC3	100.000	00.000	100.000	100.000
K3 .BC1	60.000	0.000	60.000	20.000
K3 .BC2	60.000	0.000	60.000	20.000
K3 .BC3	60.000	0.000	60.000	20.000

- q(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

	SK2
BH1.BC1	1000.000

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

	SK3
BH1.BC3	600.000

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST	SK2
BH2.BC1	584.547	1215.453

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

	SK2
BH1.BC1	600.000

- qcu(I)

(ALL 0.000)

- qhu(J)

(ALL 0.000)

- fhp(I,K,BH)

	BH1	BH2
I1.KFIRST	8.000	0
I1.K2	8.000	0
I1.K3	0	8.000
I2.K3	15.000	0
I3.KFIRST	1.505	13.495
I3.K2	3.530	11.470

- fcp(J,K,BC)

	BC1	BC2	BC3
J1.KFIRST	8.986	0	11.014
J1.K2	20.000	0	0
J1.K3	6.062	11.937	2.001
J2.KFIRST	3.352	7.226	4.422

J2.K2	7.500	0	7.500
J2.K3	0	0	15.000

- a(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

SK2

BH1.BC1 33.466

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

SK3

BH1.BC3 18.011

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

SKFIRST SK2

BH2.BC1 13.344 32.043

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

SK2

BH1.BC1 10.108

- acu(I)

(ALL 0.000)

- ahu(J)

(ALL 0.000)

5th step

- th(I,K)

	KFIRST	K2	K3	KLAST
I1	155.000	155.000	30.000	30.000
I2	80.000	80.000	80.000	40.000
I3	200.000	200.000	40.000	40.000

- thp(I,K,BH,SK)

INDEX I = I1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	155.000	155.000	155.000	155.000
KFIRST.BH2	155.000	155.000	155.000	155.000
K2 .BH1	155.000	155.000	30.000	30.000
K2 .BH2	155.000	155.000	155.000	155.000
K3 .BH1	30.000	30.000	30.000	30.000
K3 .BH2	30.000	30.000	30.000	30.000

INDEX I = I2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BH1	80.000	80.000	80.000	80.000
KFIRST.BH2	80.000	80.000	80.000	80.000
K2 .BH1	80.000	80.000	80.000	80.000
K2 .BH2	80.000	80.000	80.000	80.000
K3 .BH1	80.000	80.000	80.000	40.000
K3 .BH2	80.000	80.000	80.000	80.000

INDEX I = I3

	SKFIRST	SK2	SK3	SKLAST

KFIRST.BH1	200.000	200.000	200.000	200.000
KFIRST.BH2	200.000	200.000	200.000	200.000
K2 .BH1	200.000	200.000	30.037	30.037
K2 .BH2	200.000	149.036	43.066	43.066
K3 .BH1	40.000	40.000	40.000	40.000
K3 .BH2	40.000	40.000	40.000	40.000

- tc(J,K)

	KFIRST	K2	K3	KLAST
J1	160.000	160.000	20.000	20.000
J2	100.000	100.000	60.000	20.000

- tcp(J,K,BC,SK)

INDEX 1 = J1

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	160.000	160.000	160.000	160.000
KFIRST.BC2	160.000	20.000	20.000	20.000
KFIRST.BC3	160.000	20.000	20.000	20.000
K2 .BC1	160.000	130.773	20.000	20.000
K2 .BC2	160.000	20.000	20.000	20.000
K2 .BC3	160.000	20.000	20.000	20.000
K3 .BC1	20.000	20.000	20.000	20.000
K3 .BC2	20.000	20.000	20.000	20.000
K3 .BC3	20.000	20.000	20.000	20.000

INDEX 1 = J2

	SKFIRST	SK2	SK3	SKLAST
KFIRST.BC1	100.000	100.000	100.000	100.000
KFIRST.BC2	100.000	20.000	20.000	20.000
KFIRST.BC3	100.000	20.000	20.000	20.000
K2 .BC1	100.000	100.000	20.000	20.000
K2 .BC2	100.000	20.000	20.000	20.000
K2 .BC3	100.000	100.000	100.000	100.000
K3 .BC1	60.000	20.000	20.000	20.000
K3 .BC2	60.000	20.000	20.000	20.000
K3 .BC3	60.000	60.000	60.000	20.000

- q(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

	SK2
BH1.BC1	1000.000

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

	SK3
BH1.BC3	600.000

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

	SKFIRST	SK2
BH2.BC1	84.547	1215.453

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

	SK2
BH1.BC1	600.000

- qcu(I)

(ALL 0.000)

- qhu(J)

(ALL 0.000)

- fhp(I,K,BH)

	BH1	BH2
I1.K2	8.000	0
I1.K3	0	8.000
I2.K2	0	15.000
I2.K3	15.000	0
I3.KFIRST	15.000	0
I3.K2	3.530	11.470
I3.K3	15.000	0

- fcp(J,K,BC)

	BC1	BC3
J1.KFIRST	20.000	0
J1.K2	20.000	0
J1.K3	20.000	0
J2.KFIRST	15.000	0
J2.K2	7.500	7.500
J2.K3	0	15.000

- z(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

SK2

BH1.BC1 1.000

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

SK3

BH1.BC3 1.000

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

SKFIRST SK2

BH2.BC1 1.000 1.000

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

SK2

BH1.BC1 1.000

- zcu(I)

(ALL 0.000)

- zhu(J)

(ALL 0.000)

- a(I,J,K,BH,BC,SK)

INDEX 1 = I1 INDEX 2 = J1 INDEX 3 = K2

SK2

BH1.BC1 33.466

INDEX 1 = I2 INDEX 2 = J2 INDEX 3 = K3

SK3

BH1.BC3 18.011

INDEX 1 = I3 INDEX 2 = J1 INDEX 3 = K2

SKFIRST SK2

BH2.BC1 13.344 32.043

INDEX 1 = I3 INDEX 2 = J2 INDEX 3 = K2

SK2

BH1.BC1 10.108

- acu(I)

(ALL 0.000)

- ahu(J)

(ALL 0.000)

Appendix B The Comparison of the HEN with Isothermal and Non-isothermal for Case Study 2

The purpose of this comparison is to illustrate the benefits of using non-isothermal mixing model instead of isothermal mixing one. The optimal structure for the non-isothermal mixing model is different from for the isothermal mixing model.

The TAC for using non-isothermal mixing model allowing stream splitting is \$100,720 per year with 210.9 m^2 exchange area; whereas, the non-isothermal mixing model obtains the TAC of \$94,183 and total exchange area of 187.55 m^2. No external utilities are used in any solutions. This difference is because an isothermal model cannot even admit the HEN in Figure B2, as it would see a temperature cross in heat exchanger number 2.

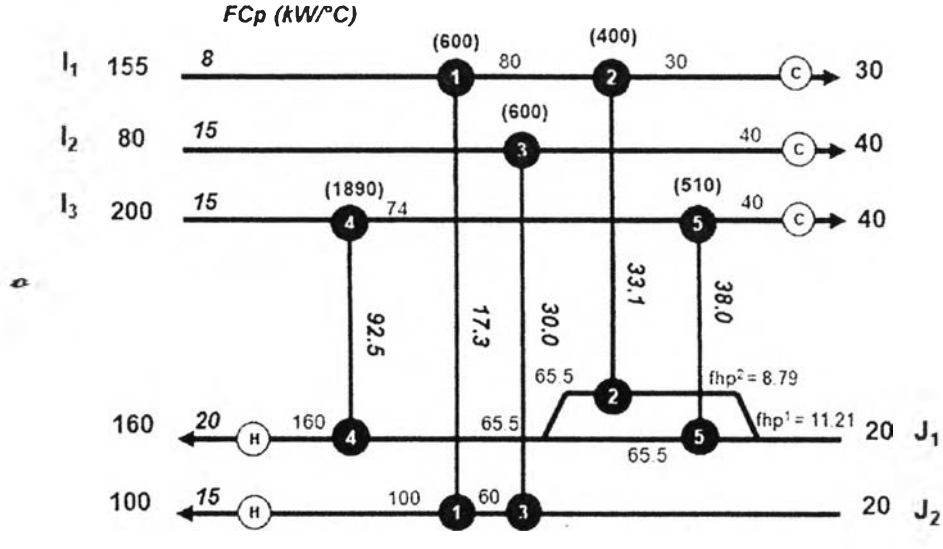

Figure B1 The HEN under isothermal mixing for case study 2 with the TAC of \$100,720 and total exchange area of 210.9 m^2.

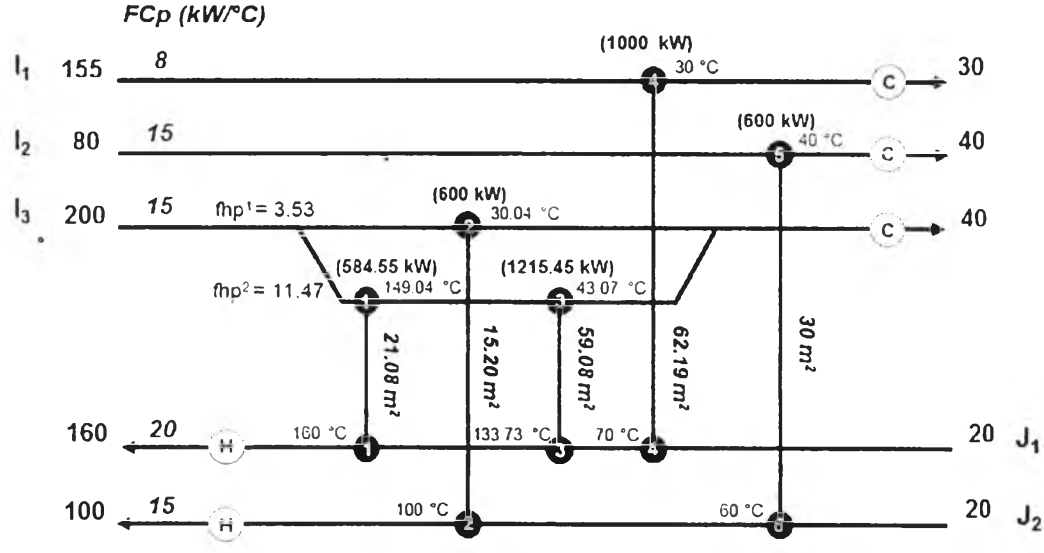

Figure B2 The HEN from this work considering non-isothermal mixing for case study 2 with the TAC of \$94,183 and total exchange area of 187.55 m^2.

Appendix C HEN Synthesis

SETS

I Hot stream /I1,I2,I3/

J Cold stream /J1,J2/

K Major stage /KFIRST,K2*K3,KLAST/

SK Sub-stage /SKFIRST,SK2*SK3,SKLAST/

BH Branch of hot splitting stream /BH1*BH2/

BC Branch of hot splitting stream /BC1*BC3/

;

SCALARS

NoBH Number of branch of hot splitting stream /2/

NoBC Number of branch of hot splitting stream /3/

ACHX Area cost coefficient of process heat exchanger /600/

ACCU Area cost coefficient of cold utility /600/

ACHU Area cost coefficient of hot utility /600/

CFHX Fixed charges of exchanger /6000/

CFCU Fixed charges of cold utility /6000/

CFHU Fixed charges of hot utility /6000/

CCU Per unit cost of cold utility /20/

CHU Per unit cost of hot utility /120/

CUMIN Minimum cold utility require /0/

CUMAX Maximum cold utility require /4000/

HUMIN Minimum hot utility require /0/

HUMAX Maximum hot utility require /4000/

EMAT Exchange minimum approach temperature /10/

TMIN Minimum temperature In HEN /20/

TMAX Maximum temperature In HEN /200/

PARAMETERS

*TEMPERATURE OF STREAM.....

THIN(I) Inlet temperature of hot stream

/ I1 155

I2 80

I3 200 /

THOUT(I) Outlet temperature of hot stream

/ I1 30

I2 40

I3 40 /

TCIN(J) Inlet temperature of cold stream

/ J1 20

J2 20 /

TCOUT(J) Outlet temperature of cold stream

/ J1 160

J2 100 /

TCUIN /20/

TCUOUT /30/

THUIN /220/

THUOUT /220/

*HEAT CAPACITY FLOWRATE OF PROCESS STREAM.....

FH(I) Heat capacity flowrate of hot stream

/ I1 8

I2 15

I3 15 /

FC(J) Heat capacity flowrate of cold stream

/ J1 20

J2 15 /

*BRANCH FLOW __ [PARAMETER].....

FHP_P(I,K,BH) Branch flow parameter of hot stream

FCP_P(J,K,BC) Branch flow parameter of cold stream

FBHPT_P(I,K)

FBCPT_P(J,K)

*BINARY PARAMETER __ [PARAMETER].....

Z_P(I,J,K,BH,BC,SK) Binary parameter of exchanger existence

ZCU_P(I) Binary parameter of cold utility existence

ZHU_P(J) Binary parameter of hot utility existence

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....

QUP_P(I,J,K,BH,BC,SK) Upper bound of heat exchange

QLO_P(I,J,K,BH,BC,SK) Lower bound of heat exchange

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....

OMEGA(I,J) Upper bound for heat exchange

HCT_P(I) Heat content of hot stream

CCT_P(J) Heat content of cold stream

GAMMA(I,J) Upper bound for temperature difference

GAMMAH(I) Upper bound for temperature difference of hot stream

GAMMAC(J) Upper bound for temperature difference of cold stream

*UPPER BOUND FOR HEAT UTILITY __ [PARAMETER].....

TOTALUTIL_P Total heat exchange value

BETA(I,J) exponent for area costs of HX I-J

BETACU(I) exponent for area costs of cooler

BETAHU(J) exponent far area costs of heater

*OVERALL HEAT TRANSFER COEFFICIENT

U(I,J) overall heat transfer coeff. of heat exchanger of I-J [KW*(m2*K)-1]

UCU(I) overall heat transfer coeff. of cooler

UHU(J) overall heat transfer coeff. of heater

*U(I,J) = [H(I)*H(J)]/[H(I)+H(J)]

DTCUP(I)

DTHUP(J)

;

DTHUP(J) = THUIN-TCOUT(J);

DTCUP(I) = THOUT(I)-TCUIN;

BETA(I,J) = 0.85;

BETACU(I) = 0.85;

BETAHU(J) = 0.85;

U(I,J) = 1;

UCU(I) = 1;

UHU(J) = 1;

VARIABLES

*TEMPERATURE OF PROCESS STREAM.....

th(I,K) Temperature of hot stream at stage K

thp(I,K,BH,SK) Temperature of hot stream at sub-stage SK In stage K

tc(J,K) Temperature of cold stream at stage K

tcp(J,K,BC,SK) Temperature of cold stream at sub-stage SK In stage K

thpi(I,J,K)

tcpi(I,J,K)

*HEAT CAPACITY FLOWRATE OF PROCESS STREAM.....

f(I,J,K)

g(I,J,K)

fhp(I,K,BH) Branch flow parameter of hot stream

fcp(J,K,BC) Branch flow parameter of cold stream

*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE (T*FCP).....

fhpt(I,K,BH,SK) MultIple of temperature and heat flow of hot stream

fcpt(J,K,BC,SK) MultIple of temperature and heat flow of cold stream

fbhpt(I,K)

fbcpt(J,K)

*HEAT EXCHANGE.....

qi(I,J,K)

qh(K,I,BH,SK) Heat exchange of hot splIttIng BH In sub-stage SK

qhKb(K,I,BH) Heat exchange of hot splIttIng BH at stage K

qhK(K,I) Heat exchange of hot (I-J) at stage K

qhc(K) Heat exchange of hot stream I In stage K

qc(K,J,BC,SK) Heat exchange of hot and cold (I-J) In sub-stage SK

qcKb(K,J,BC) Heat exchange of cold splIttIng BC at stage K

qcK(K,J) Heat exchange of cold (I-J) at stage K

qch(K) Heat exchange of cold stream I In stage K

TOTALqex Total heat exchange

TOTAL_HU Total hot utIlIty

TOTAL_CU Total cold utIlIty

*COST.....

costi

UC UtIlIty cost

NEX Number of exchanger

AC Area cost

BRANCH
TAC1
TAC2
TAC3
TAC4
TAC5
COST

;

BINARY VARIABLES
*EXISTENCE OF EXCHANGER.....
zi(I,J,K)
zcui(I)
zhui(J)
z(I,J,K,BH,BC,SK) ExIstence of exchanger I-J In each sK
zcu(I) ExIstence of cold utIIIty
zhu(J) ExIstence of hot utIIIty

;

POSITIVE VARIABLES
*TEMPERATURE APPROACH.....
dthi(I,J,K)
dtci(I,J,K)
dtcui(I)
dthui(J)
dth(I,J,K,BH,BC,SK) Temperature dIfference at "hot end" of exchanger
dtc(I,J,K,BH,BC,SK) Temperature dIfference at "cold end" of exchanger
dtcu(I) Temperature dIfference of cold utIIIty
dthu(J) Temperature dIfference of hot utIIIty

*LOG MEAN TEMPERATURE DIFFERENCE.....
LMTDHX(I,J,K,BH,BC,SK) Log mean emperature dIfference of

exchanger I-J

LMTCU(I) Log mean emperature dIfference of cold utIIIty

LMTHU(J) Log mean emperature dIfference of hot utIIIty

*AREA.....

ai(I,J,K)

acui(I)

ahui(J)

a(I,J,K,BH,BC,SK) Heat exchange area of process exchanger

acu(I) Heat exchange area of cold utIIIty

ahu(J) Heat exchange area of hot utIIIty

*HEAT EXCHANGE.....

qi(I,J,K)

qcui(I)

qhui(J)

q(I,J,K,BH,BC,SK) Heat exchange between process stream I-J

qcu(I) Heat exchange of cold utIIIty

qhu(J) Heat exchange of hot utIIIty

*_____

_____

*..... AUTOMATICCALCULATION OF PARAMETER / VARIABLE

.....

*_____

_____

SCALARS HI,CJ;

HI=1;

```
CJ=1;
*LOOP OF HOT PROCESS STREAM.....
FOR(HI=1 to CARD(I),
    HCT_P(I)$[ORD(I) = HI] = FH(I)*(THIN(I)-THOUT(I));
    GAMMAH(I)$[ORD(I) = HI] = THIN(I)-THOUT(I);
*TEMPERATURE
    th.lo(I,K)$[ORD(I) = HI] = THOUT(I);
    th.up(I,K)$[ORD(I) = HI] = THIN(I);
    thp.lo(I,K,BH,SK)$[ORD(I) = HI] = TMIN;
    thp.up(I,K,BH,SK)$[ORD(I) = HI] = TMAX;
*HEAT CAPACITY FLOWRATE
    fhp.lo(I,K,BH)$[ORD(I) = HI] = 0;
    fhp.up(I,K,BH)$[ORD(I) = HI] = FH(I);
*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE
(T*FCP)
    fhpt.lo(I,K,BH,SK)$[ORD(I) = HI] = 0;
    fhpt.up(I,K,BH,SK)$[ORD(I) = HI] = FH(I)*TMAX;
    fbhpt.lo(I,K)$[ORD(I) = HI] = 0;
    fbhpt.up(I,K)$[ORD(I) = HI] =FH(I)*THIN(I);
*HEAT EXCHANGE
    qh.lo(K,I,BH,SK)$[ORD(I) = HI] = 0;
    qh.up(K,I,BH,SK)$[ORD(I) = HI] = HCT_P(I);
    qhKb.lo(K,I,BH)$[ORD(I) = HI] = 0;
    qhKb.up(K,I,BH)$[ORD(I) = HI] = HCT_P(I);
    qhK.lo(K,I)$[ORD(I) = HI] = 0;
    qhK.up(K,I)$[ORD(I) = HI] = HCT_P(I);
    qhc.lo(K) = 0;
    qhc.up(K) = CUMAX;
    qcu.lo(I)$[ORD(I) = HI] = 0;
    qcu.up(I)$[ORD(I) = HI] = HCT_P(I);

    qcui.lo(I)$[ORD(I) = HI] = 0;
```

```
        qcui.up(I)$[ORD(I) = HI] = HCT_P(I);
);
*LOOP OF COLD PROCESS STREAM.....
For(CJ=1 to CARD(J),
        CCT_P(J)$[ORD(J) = CJ] = FC(J)*(TCOUT(J)-TCIN(J));
        GAMMAC(J)$[ORD(J) = CJ] = TCOUT(J)-TCIN(J);
*TEMPERATURE
        tc.lo(J,K)$[ORD(J) = CJ] = TCIN(J);
        tc.up(J,K)$[ORD(J) = CJ] = TCOUT(J);
        tcp.lo(J,K,BC,SK)$[ORD(J) = CJ] = TMIN;
        tcp.up(J,K,BC,SK)$[ORD(J) = CJ] = TMAX;
*HEAT CAPACITY FLOWRATE
        fcp.lo(J,K,BC)$[ORD(J) = CJ] = 0;
        fcp.up(J,K,BC)$[ORD(J) = CJ] = FC(J);
*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE
(T*FCP)
        fcpt.lo(J,K,BC,SK)$[ORD(J) = CJ] = 0;
        fcpt.up(J,K,BC,SK)$[ORD(J) = CJ] = FC(J)*TMAX;
        fbcpt.lo(J,K)$[ORD(J) = CJ] = 0;
        fbcpt.up(J,K)$[ORD(J) = CJ] = FC(J)*TCOUT(J);
*HEAT EXCHANGE
        qc.lo(K,J,BC,SK)$[ORD(J) = CJ] = 0;
        qc.up(K,J,BC,SK)$[ORD(J) = CJ] = CCT_P(J);
        qcKb.lo(K,J,BC)$[ORD(J) = CJ] = 0;
        qcKb.up(K,J,BC)$[ORD(J) = CJ] = CCT_P(J);
        qcK.lo(K,J)$[ORD(J) = CJ] = 0;
        qcK.up(K,J)$[ORD(J) = CJ] = CCT_P(J);
        qch.lo(K) = 0;
        qch.up(K) = HUMAX;
        qhu.lo(J)$[ORD(J) = CJ] = 0;
        qhu.up(J)$[ORD(J) = CJ] = CCT_P(J);
```

```
        qhui.lo(J)$[ORD(J) = CJ] = 0;
        qhui.up(J)$[ORD(J) = CJ] = CCT_P(J);
);

*_____
_____

*..... BOUNDS .....

*_____
_____

qi.lo(I,J,K) = 0;
qi.up(I,J,K) = MIN(HCT_P(I),CCT_P(J));

f.lo(I,J,K) = 0;
f.up(I,J,K) = FH(I);
g.lo(I,J,K) = 0;
g.up(I,J,K) = FC(J);

thpi.lo(I,J,K) = TMIN;
thpi.up(I,J,K) = TMAX;
tcpi.lo(I,J,K) = TMIN;
tcpi.up(I,J,K) = TMAX;

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....
        QUP_P(I,J,K,BH,BC,SK) = MIN(HCT_P(I),CCT_P(J),MAX(0,(THIN(I)-
TCIN(J)-EMAT)*MIN(FH(I),FC(J))));
        QLO_P(I,J,K,BH,BC,SK) = 1*U(I,J)*EMAT;

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....
        OMEGA(I,J) = MIN(HCT_P(I),CCT_P(J));
```

GAMMA(I,J) = MAX[0,(THIN(I)-TCIN(J)),(THIN(I)-TCOUT(J)),(THOUT(I)-TCIN(J)),
(THOUT(I)-TCOUT(J)),(TCIN(J)-THIN(I)),(TCIN(J)-THOUT(I)),
(TCOUT(J)-THIN(I)),(TCOUT(J)-THOUT(I))];

*_____

*..... BQUATION .....

*_____

EQUATIONS

thinassign(I) assignment of inlet hot temperatures
tcinassign(J) assignment of inlet cold temperature

hotk(K,I) heat balance for hot stream at each stage
coldk(K,J) heat balance for cold stream at each stage

qhnoniso(I,J,K)
qcnoniso(I,J,K)

thoutst(I,K)
tcoutst(J,K)

qcooler(I) cold utility load
qheater(J) hot utility load

hoteqi(I) overall heat balance for hot stream
coldeqi(J) overall heat balance for cold stream

logif(I,J,K)

logig(I,J,K)

sumf(I,K)

sumg(J,K)

monoth(I,K) monotonic decrease in temperature for hot stream at each stage

monothout(I) monotonic decrease in temperature for hot stream at outlet of superstructure

monotc(J,K) monotonic decrease in temperature for cold stream at each stage

monotcout(J) monotonic decrease in temperature for cold stream at outlet of superstructure

logiqi(I,J,K) logical constraint for qi(ijk)

logiqcu(I) logical constraint for qcu(i)

logiqhu(J) logical constraint for qhu(j)

dthimin(I,J,K) minimum temperature approach for hot end of HX i-j at stage k

dtcimin(I,J,K) minimum temperature approach for cold end of HX i-j at stage k

dtcuimin(I) minimum temperature approach for cooler

dthuimin(J) minimum temperature approach for heater

logidthi(I,J,K) logical constraint for dt(ijk) at inlet of stage k

logidtci(I,J,K) logical constraint for dt(ijk) at outlet of stage k

logidtcu(I) logical constraint for dtcu(i)

logidthu(J) logical constraint for dthu(j)

areai(I,J,K) area equation of heat exchanger i-j in stage k

areacui(I) area equation of cold utility

areahui(J) area equation of hot utility

obji objective function to be minimized

****
****
*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BH,BC)

* 1. OVERALL HEAT BALANCE FOR EACH STREAM**********
HOTEQ(I)
COLDEQ(J)

* 2. HEAT BALANCE AT EACH STAGE K**********
HOTCOLD_K(K)

QHC_K(K)
QHK_K(K,I)
QCH_K(K)
QCK_K(K,J)

* 3. HEAT BALANCE AT EACH SUB-STAGE SK**********
*[VARIABLE]
QHK_SK(K,I)
SUMSK_QH(K,I,BH)
SUMJ_BC_Q(I,K,BH,SK)
QH_SK(I,K,BH,SK)
HOT_SK(I,K,BH,SK)
FHPT_SK(I,K,BH,SK)

QCK_SK(K,J)
SUMSK_QC(K,J,BC)
SUMI_BH_Q(J,K,BC,SK)
QC_SK(J,K,BC,SK)
COLD_SK(J,K,BC,SK)
FCPT_SK(J,K,BC,SK)
*[FLOW PARAMETER]
HOT_SK_P(I,K,BH,SK)
FHPT_SK_P(I,K,BH,SK)
COLD_SK_P(J,K,BC,SK)
FCPT_SK_P(J,K,BC,SK)

* 4. COLD/HOT
UTILITY**********
***
QCULOAD(I)
QHULOAD(J)

* 5. TEMPERATURE
ASSIGNMENT**********
ASSIGNTH_K(I)
ASSIGNTH_SK(I,K)
ASSIGNTH_BYPASSH(I,K)
HMIX(I,K)
ASSIGNTH_TSK(I,K,BH)
ASSIGNTC_K(J)
ASSIGNTC_SK(J,K)
ASSIGNTH_BYPASSC(J,K)
CMIX(J,K)
ASSIGNTC_TSK(J,K,BC)
*[FLOW PARAMETER]
ASSIGNTH_SK_P(I,K)

ASSIGNTH_BYPASSH_P(I,K)

HMIX_P(I,K)

ASSIGNTC_SK_P(J,K)

ASSIGNTH_BYPASSC_P(J,K)

CMIX_P(J,K)

* 7. TEMPERATURE

FEASIBILITY**********

THFEAS_K(I,K)

THPFEAS_SK(I,K,BH,SK)

THFEAS_KLAST(I)

THPFEAS_SK_MIN(I,K,BH,SK)

THPFEAS_SK_MAX(I,K,BH,SK)

TCFEAS_K(J,K)

TCPFEAS_SK(J,K,BC,SK)

TCFEAS_KFIRST(J)

TCPFEAS_SK_MIN(J,K,BC,SK)

TCPFEAS_SK_MAX(J,K,BC,SK)

* 8. FLOW

FEASIBILITY**********

****

*[VARIABLE]

SUMFHP(I,K)

SUMFCP(J,K)

*[FLOW PARAMETER]

SUMFHP_P(I,K)

SUMFCP_P(J,K)

* 9.LOGICAL CONSTRAINTS [HEAT EXCHANGE

,BRANCH+BYPASS]**********

**********HEAT EXCHANGE**********

*[Z-VARIABLE]
LOGq(I,J,K,BH,BC,SK)
LOGQCU(I)
LOGQHU(J)
*[Z-PARAMETER]
LOGQ_P(I,J,K,BH,BC,SK)
LOGQCU_P(I)
LOGQHU_P(J)

*10.OTHER
CONSTRAINTS**********
****
**********MAXIMUM MATCHING**********
*[Z- VARIABLE]
CONZ1(I,K,BH,SK)
CONZ2(J,K,BC,SK)
CONZ3(I,J,K)
CONZH(I,K)
CONZC(J,K)
*[Z-PARAMETER]
CONZ1_P(I,K,BH,SK)
CONZ2_P(J,K,BC,SK)
CONZ3_P(I,J,K)
CONZH_P(I,K)
CONZC_P(J,K)
**********FLOW CONSTRAINT**********
*[VARIABLES]
CON_FLOWH1(I,K,BH)
CON_FLOWH2(I,K,BH)
CON_FLOWC1(J,K,BC)
CON_FLOWC2(J,K,BC)

*[FLOW PARAMETER]
CON_FLOWH1_F_P(I,K,BH)
CON_FLOWH2_F_P(I,K,BH)
CON_FLOWC1_F_P(J,K,BC)
CON_FLOWC2_F_P(J,K,BC)
**********MASS BALANCE AT EACH STAGE*****
MASSK_H1(I,K)
MASSK_C1(J,K)
**********CONSTRAINT**********
CON_CU1
CON_CU2
CON_CU3(I)
CON_CU4
CON_HU1
CON_HU2
CON_HU3(J)
CON_HU4
CONQ_UP(I,J,K,BH,BC,SK)
CONQ_LO(I,J,K,BH,BC,SK)
CON_EX

* 11. CALCULATION OF APPROACH
TEMPERATURE**********
*[Z- VARIABLE]
DTHMIN(I,J,K,BH,BC,SK)
DTHMAX(I,J,K,BH,BC,SK)
DTCMIN(I,J,K,BH,BC,SK)
DTCMAX(I,J,K,BH,BC,SK)

DTCUMIN(I)
DTCUMAX(I)
DTHUMIN(J)

DTHUMAX(J)

*[Z- PARAMETER]

DTHMIN_P(I,J,K,BH,BC,SK)

DTHMAX_P(I,J,K,BH,BC,SK)

DTCMIN_P(I,J,K,BH,BC,SK)

DTCMAX_P(I,J,K,BH,BC,SK)

DTCUMIN_P(I)

DTCUMAX_P(I)

DTHUMIN_P(J)

DTHUMAX_P(J)

* 12. AREA EQUATION**********

LMTD(I,J,K,BH,BC,SK)

LMTHc(I,J,K,BH,BC,SK)

LMTDCU(I)

LMTCUc(I)

LMTDHU(J)

LMTHUc(J)

AREA(I,J,K,BH,BC,SK)

AREACU(I)

AREAHU(J)

* 13. OBJECTIVE FUNCTION**********

OBJ1_NoHX

OBJ2_UTIL

OBJ3_AREA

```
OBJFN1
OBJFN2
OBJFN3
OBJFN4
OBJFN5
TOTALCOST
;

hoteqi(I)..      FH(I)*(THIN(I)-THOUT(I))   =e=    sum((J,K),qi(I,J,K)) + qcui(I);
coldeqi(J)..     FC(J)*(TCOUT(J)-TCIN(J))   =e=    sum((I,K),qi(I,J,K)) + qhui(J);

hotK(K,I)$(ORD(K) NE CARD(K))..     FH(I)*(th(I,K) -
th(I,K+1))      =e=    sum(J,qi(I,J,K)) ;
coldK(K,J)$(ORD(K) NE CARD(K))..    FC(J)*(tc(J,K) -
tc(J,K+1))      =e=    sum(I,qi(I,J,K));

qcooler(I)..      FH(I)*(th(I,'KLAST')-THOUT(I))       =e=      qcui(I) ;
qheater(J)..      FC(J)*(TCOUT(J)-tc(J,'KFIRST'))      =e=      qhui(J) ;

qhnoniso(I,J,K)$(ORD(K) NE
CARD(K))..       qi(I,J,K)      =l=    (f(I,J,K)+0.001)*(th(I,K)-thpi(I,J,K));
qcnoniso(I,J,K)$(ORD(K) NE
CARD(K))..       qi(I,J,K)      =l=    (g(I,J,K)+0.001)*(tcpi(I,J,K)-tc(J,K+1));

thoutst(I,K)$(ORD(K) NE
CARD(K))..         th(I,K+1)*FH(I)      =e=    sum(J,f(I,J,K)*thpi(I,J,K));
tcoutst(J,K)$(ORD(K) NE
CARD(K))..         tc(J,K)*FC(J)        =e=    sum(I,g(I,J,K)*tcpi(I,J,K));

logif(I,J,K)$(ORD(K) NE CARD(K))..          f(I,J,K)       =l=    FH(I);
logig(I,J,K)$(ORD(K) NE CARD(K))..          g(I,J,K)       =l=    FC(J);
```

```
sumf(I,K)$(ORD(K) NE CARD(K)).. sum(J,f(I,J,K)) =l= FH(I);
sumg(J,K)$(ORD(K) NE CARD(K)).. sum(I,g(I,J,K)) =l= FC(J);

monoth(I,K)$(ORD(K) NE CARD(K)).. th(I,K) =g= th(I,K+1);

monothout(I).. th(I,'KLAST') =g= THOUT(I);
monotc(J,K)$(ORD(K) NE CARD(K)).. tc(J,K) =g= tc(J,K+1);

monotcout(J).. tc(J,'KFIRST') =l= TCOUT(J);

thinassign(I). THIN(I) =e= th(I,'KFIRST');
tcinassign(J).. TCIN(J) =e= tc(J,'KLAST');

logiqi(I,J,K)$(ORD(K) NE
CARD(K)).. qi(I,J,K) =l= OMEGA(I,J)*zi(I,J,K);
logiqcu(I).. qcui(I) =l= HCT_P(I)*zcui(I);
logiqhu(J).. qhui(J) =l= CCT_P(J)*zhui(J);

dthimin(I,J,K).. dthi(I,J,K) =g= EMAT;
dtcimin(I,J,K).. dtci(I,J,K) =g= EMAT;
dtcuimin(I).. dtcui(I) =g= EMAT;
dthuimin(J).. dthui(J) =g= EMAT;

logidthi(I,J,K)$(ORD(K) NE CARD(K)).. dthi(I,J,K) =l= th(I,K)-
tcpi(I,J,K)+GAMMA(I,J)*(1-zi(I,J,K));
logidtci(I,J,K)$(ORD(K) NE
CARD(K)).. dtci(I,J,K) =l= thpi(I,J,K)-
tc(J,K+1)+GAMMA(I,J)*(1-zi(I,J,K));
logidtcu(I).. dtcui(I) =l= th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-zcui(I));
logidthu(J).. dthui(J) =l= THUOUT-
tc(J,'KFIRST')+GAMMAC(J)*(1-zhui(J));
```

```
areai(I,J,K)..   qi(I,J,K) -
(2/3)*U(I,J)*(dthi(I,J,K)**0.5)*(dtci(I,J,K)**0.5)*(ai(I,J,K)**(1/BETA(I,J)))-
          (1/6)*U(I,J)*dthi(I,J,K)*(ai(I,J,K)**(1/BETA(I,J))) -
(1/6)*U(I,J)*dtci(I,J,K)*(ai(I,J,K)**(1/BETA(I,J)))     =l=    0;
areacui(I)..     qcui(I) -
(2/3)*UCU(I)*(dtcui(I)**0.5)*(dtcup(I)**0.5)*(acui(I)**(1/BETACU(I)))-
          (1/6)*UCU(I)*dtcui(I)*(acui(I)**(1/BETACU(I))) -
(1/6)*UCU(I)*dtcup(I)*(acui(I)**(1/BETACU(I)))          =l=    0;
areahui(J)..     qhui(J) -
(2/3)*UHU(J)*(dthui(J)**0.5)*(dthup(J)**0.5)*(ahui(J)**(1/BETAHU(J)))-
          (1/6)*UHU(J)*dthui(J)*(ahui(J)**(1/BETAHU(J))) -
(1/6)*UHU(J)*dthup(J)*(ahui(J)**(1/BETAHU(J)))          =l=    0;

obji..          costi   =e=    sum((I,J,K),CFHX*zi(I,J,K))+sum(I,CFCU*zcui(I))+
          sum(J,CFHU*zhui(J))+sum(I,CCU*qcui(I))+sum(J,CHU*qhui(J))+
          sum((I,J,K),ACHX*ai(I,J,K))+sum(I,ACCU*acui(I))+sum(J,ACHU*ahui(
J));

**************************************************************************
**************************************************************************
**************************

*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM
ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BH,BC)..  q(I,J,K,BH,BC,'SKLAST') =E=    0;

* 1. OVERALL HEAT BALANCE FOR EACH
STREAM**************************************
HOTEQ(I)..      FH(I)*[THIN(I)-
THOUT(I)]       =E=    SUM((K,BH,J,BC,SK)$[(ORD(K) NE
```

```
CARD(K))AND(ORD(SK) NE CARD(SK))],q(I,J,K,BH,BC,SK)) + qcu(I);
COLDEQ(J)..     FC(J)*[TCOUT(J)-
TCIN(J)]        =E=     SUM((K,BC,I,BH,SK)$[(ORD(K) NE
CARD(K))AND(ORD(SK) NE CARD(SK))],q(I,J,K,BH,BC,SK)) + qhu(J);

* 2. HEAT BALANCE AT EACH STAGE
K************************************************
QHC_K(K)$[ORD(K) NE CARD(K)]..          qhc(K)  =E=     SUM(I,qhK(K,I));
QHK_K(K,I)$[ORD(K) NE CARD(K)]..        FH(I)*[th(I,K) -
th(I,K+1)]      =E=     qhK(K,I);
QHK_SK(K,I)$[ORD(K) NE
CARD(K)]..              qhK(K,I)        =E=     SUM(BH,qhKb(K,I,BH));
SUMJ_BC_Q(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qh(K,I,BH,SK)  =E= SUM((J,BC),q(I,J,K,BH,BC,SK));
SUMSK_QH(K,I,BH)$[ORD(K) NE
CARD(K)]..          qhKb(K,I,BH)    =E=     SUM(SK$(ORD(SK) NE
CARD(SK)),qh(K,I,BH,SK));
QCH_K(K)$[ORD(K) NE CARD(K)]..          qch(K)  =E=     SUM(J,qcK(K,J));
QCK_K(K,J)$[ORD(K) NE CARD(K)]..        FC(J)*[tc(J,K) -
tc(J,K+1)]      =E=     qcK(K,J);
QCK_SK(K,J)$[ORD(K) NE
CARD(K)]..              qcK(K,J)        =E=     SUM(BC,qcKb(K,J,BC));
SUMI_BH_Q(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qc(K,J,BC,SK)          =E=     SUM((I,BH),q(I,J,K,BH,BC,SK));
SUMSK_QC(K,J,BC)$[ORD(K) NE
CARD(K)]..          qcKb(K,J,BC)    =E=     SUM(SK$(ORD(SK) NE
CARD(SK)),qc(K,J,BC,SK));

HOTCOLD_K(K)$[ORD(K) NE CARD(K)]..      qhc(K)  =E=     qch(K);

* 3. HEAT BALANCE AT EACH SUB-STAGE
SK*******************************************
```

```
*[VARIABLE]
QH_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qh(K,I,BH,SK) =E= fhpt(I,K,BH,SK) - fhpt(I,K,BH,SK+1);
HOT_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fhpt(I,K,BH,SK) -
fhpt(I,K,BH,SK+1) =E= fhp(I,K,BH)*[thp(I,K,BH,SK)-thp(I,K,BH,SK+1)];
FHPT_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fhp(I,K,BH)*thp(I,K,BH,SK) =E= fhpt(I,K,BH,SK);
QC_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qc(K,J,BC,SK) =E= fcpt(J,K,BC,SK) -
fcpt(J,K,BC,SK+1);
COLD_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fcpt(J,K,BC,SK) -
fcpt(J,K,BC,SK+1) =E= fcp(J,K,BC)*[tcp(J,K,BC,SK)-tcp(J,K,BC,SK+1)];
FCPT_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fcp(J,K,BC)*tcp(J,K,BC,SK) =E= fcpt(J,K,BC,SK);
*[FLOW PARAMETER]
HOT_SK_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FHP_P(I,K,BH)*[thp(I,K,BH,SK)-
thp(I,K,BH,SK+1)] =E= fhpt(I,K,BH,SK) - fhpt(I,K,BH,SK+1);
FHPT_SK_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FHP_P(I,K,BH)*thp(I,K,BH,SK) =E= fhpt(I,K,BH,SK);
COLD_SK_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FCP_P(J,K,BC)*[tcp(J,K,BC,SK)-
tcp(J,K,BC,SK+1)] =E= fcpt(J,K,BC,SK) - fcpt(J,K,BC,SK+1);
FCPT_SK_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FCP_P(J,K,BC)*tcp(J,K,BC,SK) =E= fcpt(J,K,BC,SK);

* 4. COLD/HOT
UTILITY**********************************************************
***
QCULOAD(I).. qcu(I) =E= [th(I,'KLAST')-THOUT(I)]*FH(I) ;
```

```
QHULOAD(J).. qhu(J) =E= [TCOUT(J)-tc(J,'KFIRST')]*FC(J) ;

* 5. TEMPERATURE
ASSIGNMENT**********************************************************
ASSIGNTH_K(I).. THIN(I) =E= th(I,'KFIRST');
ASSIGNTH_SK(I,K)$[ORD(K) NE
CARD(K)].. FH(I)*th(I,K) =E= SUM(BH,fhpt(I,K,BH,'SKFIRST'))+
fbhpt(I,K);
ASSIGNTH_BYPASSH(I,K)$[ORD(K) NE
CARD(K)].. fbhpt(I,K) =E= (FH(I)-SUM(BH,fhp(I,K,BH)))*th(I,K);
HMIX(I,K)$[ORD(K) NE
CARD(K)].. FH(I)*th(I,K+1) =E= SUM(BH,fhpt(I,K,BH,'SKLAS
T'))+fbhpt(I,K);
ASSIGNTH_TSK(I,K,BH)$[ORD(K) NE CARD(K)]
.. TH(I,K) =E= thp(I,K,BH,'SKFIRST');
ASSIGNTC_K(J).. TCIN(J) =E= tc(J,'KLAST');
ASSIGNTC_SK(J,K)$[ORD(K) NE
CARD(K)].. FC(J)*tc(J,K+1) =E= SUM(BC,fcpt(J,K,BC,'SKLAST'))
+fbcpt(J,K+1);
ASSIGNTH_BYPASSC(J,K)$[ORD(K) NE
CARD(K)].. fbcpt(J,K+1) =E= (FC(J)-
SUM(BC,fcp(J,K,BC)))*tc(J,K+1);
CMIX(J,K)$[ORD(K) NE
CARD(K)].. FC(J)*tc(J,K) =E= SUM(BC,fcpt(J,K,BC,'SKFIRS
T'))+fbcpt(J,K+1);
ASSIGNTC_TSK(J,K,BC)$[ORD(K) NE CARD(K)]
.. TC(J,K+1) =E= tcp(J,K,BC,'SKLAST');

*[FLOW PARAMETER]
ASSIGNTH_SK_P(I,K)$[ORD(K) NE
CARD(K)].. FH(I)*th(I,K) =E= SUM(BH,FHP_P(I,K,BH)*thp(I,K,BH,'S
KFIRST'))+FBHPT_P(I,K);
```

```
ASSIGNTH_BYPASSH_P(I,K)$[ORD(K) NE
CARD(K)]..      FBHPT_P(I,K)   =E=   (FH(I)-
SUM(BH,FHP_P(I,K,BH)))*th(I,K);
HMIX_P(I,K)$[ORD(K) NE
CARD(K)]..              FH(I)*th(I,K+1) =E=   SUM(BH,FHP_P(I,K,BH)*thp(I
,K,BH,'SKLAST'))+(FH(I)-SUM(BH,FHP_P(I,K,BH)))*th(I,K);
ASSIGNTC_SK_P(J,K)$[ORD(K) NE
CARD(K)]..      FC(J)*tc(J,K+1) =E=   SUM(BC,FCP_P(J,K,BC)*tcp(J,K,BC,'
SKLAST'))+FBCPT_P(J,K+1);
ASSIGNTH_BYPASSC_P(J,K)$[ORD(K) NE
CARD(K)]..      FBCPT_P(J,K+1)  =E=   (FC(J)-
SUM(BC,FCP_P(J,K,BC)))*tc(J,K+1);
CMIX_P(J,K)$[ORD(K) NE
CARD(K)]..              FC(J)*tc(J,K)   =E=   SUM(BC,FCP_P(J,K,BC)*tcp(J,
K,BC,'SKFIRST'))+(FC(J)-SUM(BC,FCP_P(J,K,BC)))*tc(J,K+1);

**********FLOW CONSTRAINT**********
*[VARIABLES]
CON_FLOWH1(I,K,BH)$[ORD(K) NE
CARD(K)]..      fhp(I,K,BH)   =L=   FH(I);
CON_FLOWH2(I,K,BH)$[ORD(K) NE CARD(K)]..      fhp(I,K,BH)   =G=   0;
CON_FLOWC1(J,K,BC)$[ORD(K) NE
CARD(K)]..      fcp(J,K,BC)   =L=   FC(J);
CON_FLOWC2(J,K,BC)$[ORD(K) NE CARD(K)]..      fcp(J,K,BC)   =G=   0;
*[FLOW PARAMETER]
CON_FLOWH1_F_P(I,K,BH)$[ORD(K) NE
CARD(K)]..    FHP_P(I,K,BH)   =L=   FH(I);
CON_FLOWH2_F_P(I,K,BH)$[ORD(K) NE
CARD(K)]..    FHP_P(I,K,BH)   =G=   0;
CON_FLOWC1_F_P(J,K,BC)$[ORD(K) NE
CARD(K)]..    FCP_P(J,K,BC)   =L=   FC(J);
CON_FLOWC2_F_P(J,K,BC)$[ORD(K) NE
```

```
CARD(K)]..      FCP_P(J,K,BC)   =G=     0;

* 7. TEMPERATURE
FEASIBILITY**********************************************************
THFEAS_K(I,K)$[ORD(K) NE CARD(K)]..                th(I,K) =G=     th(I,K+1);
THPFEAS_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..        thp(I,K,BH,SK)    =G=     thp(I,K,BH,SK+1);
THFEAS_KLAST(I)..                          th(I,'KLAST')    =G=    THOUT(I);
TCFEAS_K(J,K)$[ORD(K) NE CARD(K)]..                tc(J,K) =G=     tc(J,K+1);
TCPFEAS_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..        tcp(J,K,BC,SK)    =G=     tcp(J,K,BC,SK+1);
TCFEAS_KFIRST(J)..                         tc(J,'KFIRST')   =L=    TCOUT(J);

* 8. FLOW
FEASIBILITY**********************************************************
********
*[VARIABLE]
SUMFHP(I,K)$[ORD(K) NE
CARD(K)]..      SUM(BH,fhp(I,K,BH))          =L=     FH(I);
SUMFCP(J,K)$[ORD(K) NE
CARD(K)]..      SUM(BC,fcp(J,K,BC))          =L=     FC(J);
*[FLOW PARAMETER]
SUMFHP_P(I,K)$[ORD(K) NE
CARD(K)]..      SUM(BH,FHP_P(I,K,BH))        =L=     FH(I);
SUMFCP_P(J,K)$[ORD(K) NE
CARD(K)]..      SUM(BC,FCP_P(J,K,BC))        =L=     FC(J);


* 9.LOGICAL CONSTRAINTS [HEAT EXCHANGE
,BRANCH+BYPASS]*****************************
***********HEAT EXCHANGE***********
*[ Z-VARIABLE]
```

```
LOGq(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. q(I,J,K,BH,BC,SK) - OMEGA(I,J)*z(I,J,K,BH,BC,SK)  =L=  0;
LOGQCU(I).. qcu(I) - HCT_P(I)*zcu(I)  =L=  0;
LOGQHU(J).. qhu(J) - CCT_P(J)*zhu(J)  =L=  0;
*[Z-PARAMETER]
LOGQ_P(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. q(I,J,K,BH,BC,SK) -
OMEGA(I,J)*Z_P(I,J,K,BH,BC,SK)  =L=  0;
LOGQCU_P(I).. qcu(I) - HCT_P(I)*ZCU_P(I)  =L=  0;
LOGQHU_P(J).. qhu(J) - CCT_P(J)*ZHU_P(J)  =L=  0;

*10.OTHER
CONSTRAINTS**************************************************
*******
************MAXIMUM MATCHING************
*[ Z- VARIABLE]
CONZ1(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((J,BC),z(I,J,K,BH,BC,SK))  =L=  1;
CONZ2(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((I,BH),z(I,J,K,BH,BC,SK))  =L=  1;
CONZ3(I,J,K)$[ORD(K) NE CARD(K)].. nexh(I,K)  =E=  nexc(J,K);
CONZH(I,K)$[ORD(K) NE CARD(K)]..
nexh(I,K)  =E=  SUM((J,SK,BH,BC),z(I,J,K,BH,BC,SK));
CONZC(J,K)$[ORD(K) NE CARD(K)]..
nexc(J,K)  =E=  SUM((I,SK,BH,BC),z(I,J,K,BH,BC,SK));
*[ Z-PARAMETER]
CONZ1_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((J,BC),Z_P(I,J,K,BH,BC,SK))  =L=  1;
CONZ2_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((I,BH),Z_P(I,J,K,BH,BC,SK))  =L=  1;
CONZ3_P(I,J,K)$[ORD(K) NE CARD(K)].. ZH_P(I,K)  =E=  ZC_P(J,K);
CONZH_P(I,K)$[ORD(K) NE
```

```
CARD(K)]..      ZH_P(I,K)  =E=    SUM((J,SK,BH,BC),Z_P(I,J,K,BH,BC,SK));
CONZC_P(J,K)$[ORD(K) NE
CARD(K)]..      ZC_P(J,K)  =E=    SUM((I,SK,BH,BC),Z_P(I,J,K,BH,BC,SK));

*MASS BALANCE AT EACH STAGE
MASSK_H1(I,K)$(ORD(K) NE CARD(K))..     SUM(BH,fhp(I,K,BH))+(FH(I)-
SUM(BH,fhp(I,K,BH)))      =E=     FH(I);
MASSK_C1(J,K)$(ORD(K) NE CARD(K))..     SUM(BC,fcp(J,K,BC))+(FC(J)-
SUM(BC,fcp(J,K,BC)))      =E=     FC(J);

* Heat Exchange and Heat & Cold utility constraints
* Cold utility....
CON_CU1 .. SUM(I,qcu(I))=G= CUMIN;
CON_CU2 .. SUM(I,qcu(I))=L= CUMAX;
CON_CU3(I) .. qcu(I) =L= zcu(I)*HCT_P(I);
CON_CU4 .. TOTAL_CU =E= SUM(I,qcu(I));
* Hot utility....
CON_HU1 .. SUM(J,qhu(J))=G= HUMIN;
CON_HU2 .. SUM(J,qhu(J))=L= HUMAX;
CON_HU3(J) .. qhu(J) =L= zhu(J)*CCT_P(J);
CON_HU4 .. TOTAL_HU =E= SUM(J,qhu(J));

**********HEAT EXCHANGE**********
*[ Z- VARIABLE]
CON_EX.. SUM((I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))],q(I,J,K,BH,BC,SK)) =G= HUMAX-TOTAL_HU;
CONQ_UP(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]
..   q(I,J,K,BH,BC,SK)  =L=    z(I,J,K,BH,BC,SK)*QUP_P(I,J,K,BH,BC,SK);
CONQ_LO(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]
..   q(I,J,K,BH,BC,SK)  =G=    z(I,J,K,BH,BC,SK)*QLO_P(I,J,K,BH,BC,SK);
```

```
* 11. CALCULATION OF APPROACH
TEMPERATURE*****************************************
*[ Z- VARIABLE]
DTHMAX(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dth(I,J,K,BH,BC,SK) =L= thp(I,K,BH,SK)-
tcp(J,K,BC,SK)+GAMMA(I,J)*(1-z(I,J,K,BH,BC,SK));
DTCMAX(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dtc(I,J,K,BH,BC,SK) =L= thp(I,K,BH,SK+1)-
tcp(J,K,BC,SK+1)+GAMMA(I,J)*(1-z(I,J,K,BH,BC,SK));
DTCUMAX(I).. dtcu(I) =L= th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-zcu(I));
DTHUMAX(J).. dthu(J) =L= THUOUT-
tc(J,'KFIRST')+GAMMAC(J)*(1-zhu(J));

DTHMIN(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dth(I,J,K,BH,BC,SK) =G= EMAT;
DTCMIN(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dtc(I,J,K,BH,BC,SK) =G= EMAT;
DTCUMIN(I).. dtcu(I) =G= EMAT;
DTHUMIN(J).. dthu(J) =G= EMAT;

*[ Z- PARAMETER]
DTHMAX_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))].. dth(I,J,K,BH,BC,SK) =L= thp(I,K,BH,SK)-
tcp(J,K,BC,SK)+GAMMA(I,J)*(1-Z_P(I,J,K,BH,BC,SK));
DTCMAX_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))].. dtc(I,J,K,BH,BC,SK) =L= thp(I,K,BH,SK+1)-
tcp(J,K,BC,SK+1)+GAMMA(I,J)*(1-Z_P(I,J,K,BH,BC,SK));
DTCUMAX_P(I).. dtcu(I) =L= th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-ZCU_P(I));
DTHUMAX_P(J).. dthu(J) =L= THUOUT-
```

```
tc(J,'KFIRST')+GAMMAC(J)*(1-ZHU_P(J));

DTHMIN_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))].. dth(I,J,K,BH,BC,SK) =G= EMAT;
DTCMIN_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))].. dtc(I,J,K,BH,BC,SK) =G= EMAT;
DTCUMIN_P(I).. dtcu(I) =G= EMAT;
DTHUMIN_P(J).. dthu(J) =G= EMAT;

* 12. AREA
EQUATION**********************************************************
******
LMTD(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. LMTDHX(I,J,K,BH,BC,SK) =E=
                (2/3)*((dth(I,J,K,BH,BC,SK)+0.001)**0.5)*((dtc(I,J,K,BH,BC,
SK)+0.001)**0.5)+(1/6)*dth(I,J,K,BH,BC,SK)+(1/6)*dtc(I,J,K,BH,BC,SK);
LMTDCU(I).. LMTCU(I) =E=
(2/3)*((dtcu(I)+0.001)**0.5)*((DTCUP(I)+0.001)**0.5)+(1/6)*dtcu(I)+(1/6)*DTCU
P(I);

LMTDHU(J).. LMTHU(J) =E=
(2/3)*((dthu(J)+0.001)**0.5)*((DTHUP(J)+0.001)**0.5)+(1/6)*dthu(J)+(1/6)*DTH
UP(J);

AREA(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..q(I,J,K,BH,BC,SK)-
(a(I,J,K,BH,BC,SK)**(1/BETA(I,J))*U(I,J)*LMTDHX(I,J,K,BH,BC,SK)) =L= 0;
AREACU(I).. qcu(I)-(acu(I)**(1/BETACU(I))*UCU(I)*LMTCU(I)) =l= 0;
AREAHU(J).. qhu(J)-(ahu(J)**(1/BETAHU(J))*UHU(J)*LMTHU(J)) =l= 0;

* 13. OBJECTIVE
FUNCTION**********************************************************
```

```
**
OBJ1_NoHX..   NEX  =E=  sum((I,J,K,BH,BC,SK),CFHX*z(I,J,K,BH,BC,SK
))+sum(I,CFCU*zcu(I))+sum(J,CFHU*zhu(J));
OBJ2_UTIL..   UC   =E=  sum(I,CCU*qcu(I))+sum(J,CHU*qhu(J));
OBJ3_AREA..   AC   =E=  sum((I,J,K,BH,BC,SK),ACHX*a(I,J,K,BH,BC,SK)
)+sum(I,ACCU*acu(I))+sum(J,ACHU*ahu(J));

OBJFN1 ..     TAC1  =E=  UC;
OBJFN2 ..     TAC2  =E=  NEX+UC;
OBJFN3 ..     TAC3  =E=  AC;
OBJFN4 ..     TAC4  =E=  (UC + AC);
OBJFN5 ..     TAC5  =E=  (UC + NEX + AC);
TOTALCOST..   COST  =E=  (UC+NEX+AC);
*************************************
OPTION sysout = on;
OPTION IterlIm = 1e+09;
OPTION reslIm = 5e+06;
*************************************
MODEL NONISO1 "NONISO1 MINLP"
/
      hoteqi,coldeqi,hotk,coldk,qhnoniso,qcnoniso,qcooler,qheater,thinassign,tcina
ssign,monoth,
monothout,monotc,monotcout,thoutst,tcoutst,logif,logig,
sumf,sumg,logiqi,logiqcu,logiqhu,dthimin,dtcimin,dtcuimin,dthuimin,logidthi,logidt
ci,logidtcu,
logidthu,areai,areacui,areahui,obji   /

MODEL FLOW "MIN TAC1 = UC _____NLP"
/
      QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,S
UMSK_QH,QCH_K,QCK_K,
QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,QH_SK,HOT_SK,FHPT_SK,
```

QC_SK,COLD_SK,FCPT_SK,
ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPASSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,
ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,CON_FLOWH1,CON_FLOWH2
,SUMFHP,CON_FLOWC1,
CON_FLOWC2,SUMFCP,THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS
_K,TCPFEAS_SK,
TCFEAS_KFIRST,QCULOAD,QHULOAD,LOGQ_P,LOGQCU_P,LOGQHU_P,M
ASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,
DTHMAX_P,DTCMAX_P,DTCUMAX_P,DTHUMAX_P,DTHMIN_P,DTCMIN_
P,DTCUMIN_P,DTHUMIN_P,
OBJ2_UTIL,OBJFN1 /

MODEL STRUCTURE "MIN TAC2 = UC+NoHX _____MILP"
/
QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QCH_K,QCK_K,QHK_SK
,SUMJ_BC_Q,SUMSK_QH,
QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,QH_SK,HOT_SK_P,FHPT_S
K_P,QC_SK,COLD_SK_P,
FCPT_SK_P,ASSIGNTH_K,ASSIGNTH_SK_P,ASSIGNTH_BYPASSH_P,HMIX
_P,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK_P,ASSIGNTH_BYPASSC_P,CMIX_P,ASSIGNT
C_TSK,
CON_FLOWH1_F_P,CON_FLOWH2_F_P,CON_FLOWC1_F_P,CON_FLOWC2_
F_P,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEA
S_KFIRST,
QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU3,CON_CU4,CON_HU1,CON_HU2,CON_HU3,C
ON_HU4,CON_EX,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D

```
THUMIN,
OBJ1_NoHX,OBJ2_UTIL,OBJFN2          /

MODEL FLOWarea "MIN TAC3 = UC+AC _____NLP"
/
        QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,S
UMSK_QH,QCH_K,QCK_K,
QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,QH_SK,HOT_SK,FHPT_SK,
QC_SK,COLD_SK,FCPT_SK,
ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPASSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,
CON_FLOWH1,CON_FLOWH2,SUMFHP,CON_FLOWC1,CON_FLOWC2,SUM
FCP,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEA
S_KFIRST,QCULOAD,QHULOAD,
LOGQ_P,LOGQCU_P,LOGQHU_P,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,
DTHMAX_P,DTCMAX_P,DTCUMAX_P,DTHUMAX_P,DTHMIN_P,DTCMIN_
P,DTCUMIN_P,DTHUMIN_P,
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU,
OBJ2_UTIL,OBJ3_AREA,OBJFN1,OBJFN2,OBJFN3,OBJFN4          /


MODEL NONLINEAR  "FINAL"
/
        QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,S
UMSK_QH,QCH_K,QCK_K,
QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,QH_SK,HOT_SK,FHPT_SK,
QC_SK,COLD_SK,FCPT_SK,
ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPASSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,
CON_FLOWH1,CON_FLOWH2,SUMFHP,CON_FLOWC1,CON_FLOWC2,SUM
```

```
FCP,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEA
S_KFIRST,
QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,C
ONQ_UP,CONQ_LO,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D
THUMIN,
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU,OBJ1_NoHX,OBJ2_UTI
L,OBJ3_AREA,OBJFN5      /
;

**********************************************************************
**********
* DECLARATION
PARAMETER
      AHX_INT(I,J,K,BH,BC,SK)       Area of HX I-J (Area-lInear)
      ACU_INT(I)                    Area of HX I-CU (Area-lInear)
      AHU_INT(J)                    Area of HX J-HU (Area-lInear)

      TH_INT(I,K)
      THP_INT(I,K,BH,SK)
      TC_INT(J,K)
      TCP_INT(J,K,BC,SK)

      Q_INT(I,J,K,BH,BC,SK)
      QCU_INT(I)
      QHU_INT(J)
      QHK_INT(K,I)
      QCK_INT(K,J)

      FHP_INT(K,I,BH)
```

```
FCP_INT(J,K,BC)

Z_INT(I,J,K,BH,BC,SK)
ZCU_INT(I)
ZHU_INT(J)

FBHPT_INT(I,K)
FBCPT_INT(J,K)

DTH_INT(I,J,K,BH,BC,SK)
DTC_INT(I,J,K,BH,BC,SK)
DTCU_INT(I)
DTHU_INT(J)
;

*******************************ASSIGN INITIAL
PT.********************** ************
ai.l(I,J,K)=    0.001;
acui.l(I) =     0.001;
ahui.l(J) =     0.001;

f.l(I,J,K)      =       FH(I);
g.l(I,J,K)      =       FC(J);
qi.l(I,J,K)$(ORD(K) NE CARD(K)) =       MIN(HCT_P(I),CCT_P(J));
qcui.l(I)       =       0;
qhui.l(J)       =       0;
dtcui.l(I)      =       THIN(I) - TCUOUT;
dthui.l(J)      =       THUOUT - TCIN(J);
dthi.l(I,J,K)   =       THIN(I) - TCIN(J);
dtci.l(I,J,K)   =       THIN(I) - TCIN(J);
*************************************************************************
**************
```

```
NONISO1.optfile = 2;
option reslim = 10000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
************************
$onecho>dicopt.op2
STOP 0
MAXCYCLES 400
epsmip 50
$offecho
************************
SOLVE NONISO1 USING MINLP MINIMIZING costi;
DISPLAY th.l,thpi.l,tc.l,tcpi.l,ai.l,acui.l,ahui.l,qi.l,qcui.l,qhui.l,zi.l,zcui.l,zhui.l,f.l,g.l,
     costi.l;
*******************************END OF SOLVE
NONISO1*******************************
AHX_INT(I,J,K,BH,BC,SK) = ai.L(I,J,K);
ACU_INT(I) = acui.L(I);
AHU_INT(J) = ahui.L(J);

a.L(I,J,K,BH,BC,SK)=AHX_INT(I,J,K,BH,BC,SK);
acu.L(I) = ACU_INT(I);
ahu.L(J) = AHU_INT(J);

DTH_INT(I,J,K,BH,BC,SK) = dthi.L(I,J,K);
DTC_INT(I,J,K,BH,BC,SK) = dtci.L(I,J,K);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);

dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
```

```
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) =qcui.L(I);
QHU_INT(J) = qhui.L(J);
Q_INT(I,J,K,BH,BC,SK) = qi.L(I,J,K);
QHK_INT(K,I) = sum(J,qi.L(I,J,K));
QCK_INT(K,J) = sum(I,qi.L(I,J,K));

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);
qhK.L(K,I) = QHK_INT(K,I);
qcK.L(K,J) = QCK_INT(K,J);

Z_P(I,J,K,BH,BC,SK) = zi.L(I,J,K);
ZCU_P(I) = zcui.L(I);
ZHU_P(J) = zhui.L(J);
*************************
FLOW.optfile = 3;
option reslim = 10000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
*************************
$onecho>dicopt.op3
STOP 0
MAXCYCLES 200
mipoptfile cplex.opt 1
epsmip 20
$offecho
*************************
SOLVE FLOW USING NLP MINIMIZING TAC1;
DISPLAY
```

```
th.l,thp.l,tc.l,tcp.l,a.l,acu.l,ahu.l,q.l,qcu.l,qhu.l,Z_P,ZCU_P,ZHU_P,fhp.l,fcp.l,
     TAC1.l;
**********************************END OF SOLVE
FLOW*****************************
FHP_P(I,K,BH)=fhp.l(I,K,BH);
FCP_P(J,K,BC)=fcp.l(J,K,BC);
FBHPT_P(I,K) =(FH(I)-SUM(BH,FHP_P(I,K,BH)))*th.l(I,K);
FBCPT_P(J,K+1)= (FC(J)-SUM(BC,FCP_P(J,K,BC)))*tc.l(J,K+1);

AHX_INT(I,J,K,BH,BC,SK) = a.L(I,J,K,BH,BC,SK);
ACU_INT(I) = acu.L(I);
AHU_INT(J) = ahu.L(J);

a.L(I,J,K,BH,BC,SK)=AHX_INT(I,J,K,BH,BC,SK);
acu.L(I) = ACU_INT(I);
ahu.L(J) = AHU_INT(J);

TC_INT(J,K) = tc.l(J,K);
TCP_INT(J,K,BC,SK) = tcp.l(J,K,BC,SK);
TH_INT(I,K)  = th.l(I,K);
THP_INT(I,K,BH,SK) = thp.l(I,K,BH,SK);

tc.l(J,K) = TC_INT(J,K);
tcp.l(J,K,BC,SK) = TCP_INT(J,K,BC,SK);
th.l(I,K) = TH_INT(I,K);
thp.l(I,K,BH,SK) = THP_INT(I,K,BH,SK);

DTH_INT(I,J,K,BH,BC,SK) = dth.L(I,J,K,BH,BC,SK);
DTC_INT(I,J,K,BH,BC,SK) = dtc.L(I,J,K,BH,BC,SK);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);
```

```
dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K;BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcu.L(I);
QHU_INT(J) = qhu.L(J);
Q_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK);

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);

z.L(I,J,K,BH,BC,SK) = Z_P(I,J,K,BH,BC,SK);
zcu.L(I) = ZCU_P(I);
zhu.L(J) = ZHU_P(J);

************************
SOLVE STRUCTURE USING MIP MINIMIZING TAC2;
DISPLAY
th.l,thp.l,tc.l,tcp.l,a.l,acu.l,ahu.l,q.l,qcu.l,qhu.l,z.l,zcu.L,zhu.L,FHP_P,FCP_P,
     TAC2.l;
*******************************END OF SOLVE
STRUCTURE*****************************

fhp.l(I,K,BH) = FHP_P(I,K,BH);
fcp.l(J,K,BC) = FCP_P(J,K,BC);
fbhpt.l(I,K)  = FBHPT_P(I,K);
fbcpt.l(J,K+1) = FBCPT_P(J,K+1);

* Initial Area.....
AHX_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK)/((U(I,J)*(
```

```
(2/3)*((dth.L(I,J,K,BH,BC,SK)+0.001)**0.5)*((dtc.L(I,J,K,BH,BC,SK)+0.001)**0.
5)+(1/6)*dth.L(I,J,K,BH,BC,SK)+(1/6)*dtc.L(I,J,K,BH,BC,SK)))+1e-06);
ACU_INT(I) =
qcu.l(I)/((UCU(I)*(2/3)*((dtcu.L(I)+0.001)**0.5)*((DTCUP(I)+0.001)**0.5)+(1/6)*
dtcu.L(I)+(1/6)*DTCUP(I))+1e-06);
AHU_INT(J) =
qhu.l(J)/((UHU(J)*(2/3)*((dthu.L(J)+0.001)**0.5)*((DTHUP(J)+0.001)**0.5)+(1/6)
*dthu.L(J)+(1/6)*DTHUP(J))+1e-06);

a.l(I,J,K,BH,BC,SK) = AHX_INT(I,J,K,BH,BC,SK);
acu.l(I) = ACU_INT(I);
ahu.l(J) = AHU_INT(J);

Display a.l,acu.l,ahu.l;

TC_INT(J,K) = tc.l(J,K);
TCP_INT(J,K,BC,SK) = tcp.l(J,K,BC,SK);
TH_INT(I,K)  = th.l(I,K);
THP_INT(I,K,BH,SK) = thp.l(I,K,BH,SK);

tc.l(J,K) = TC_INT(J,K);
tcp.l(J,K,BC,SK) = TCP_INT(J,K,BC,SK);
th.l(I,K) = TH_INT(I,K);
thp.l(I,K,BH,SK) = THP_INT(I,K,BH,SK);

DTH_INT(I,J,K,BH,BC,SK) = dth.L(I,J,K,BH,BC,SK);
DTC_INT(I,J,K,BH,BC,SK) = dtc.L(I,J,K,BH,BC,SK);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);

dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
```

```
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcu.L(I);
QHU_INT(J) = qhu.L(J);
Q_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK);

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);

Z_INT(I,J,K,BH,BC,SK) = z.l(I,J,K,BH,BC,SK);
ZCU_INT(I) = zcu.l(I);
ZHU_INT(J) = zhu.l(J);

Z_P(I,J,K,BH,BC,SK) = Z_INT(I,J,K,BH,BC,SK);
ZCU_P(I) = ZCU_INT(I);
ZHU_P(J) = ZHU_INT(J);

*************************
FLOWarea.optfile = 4;
option reslim = 10000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
*************************
$onecho>dicopt.op4
STOP 0
MAXCYCLES 200
mipoptfile cplex.opt 1
epsmip 20
$offecho
*************************
```

```
SOLVE FLOWarea USING NLP MINIMIZING TAC4;
DISPLAY
th.l,thp.l,tc.l,tcp.l,q.l,qcu.l,qhu.l,fhp.l,fcp.l,Z_P,ZCU_P,ZHU_P,a.l,acu.l,ahu.l,
     TAC1.l,TAC2.l,TAC3.l,TAC4.l;
**********************************END OF SOLVE
FLOWarea**************************
FHP_INT(K,I,BH) = fhp.l(I,K,BH);
FCP_INT(J,K,BC) = fcp.l(J,K,BC);
FBHPT_INT(I,K)  = fbhpt.l(I,K);
FBCPT_INT(J,K+1) = fbcpt.l(J,K+1);

fhp.l(I,K,BH) = FHP_INT(K,I,BH);
fcp.l(J,K,BC) = FCP_INT(J,K,BC);
fbhpt.l(I,K)  = FBHPT_INT(I,K);
fbcpt.l(J,K+1)= FBCPT_INT(J,K+1);

AHX_INT(I,J,K,BH,BC,SK) = a.l(I,J,K,BH,BC,SK);
ACU_INT(I) = acu.l(I);
AHU_INT(J) = ahu.l(J);

a.l(I,J,K,BH,BC,SK) = AHX_INT(I,J,K,BH,BC,SK);
acu.l(I) = ACU_INT(I);
ahu.l(J) = AHU_INT(J);

TC_INT(J,K) = tc.l(J,K);
TCP_INT(J,K,BC,SK) = tcp.l(J,K,BC,SK);
TH_INT(I,K)  = th.l(I,K);
THP_INT(I,K,BH,SK) = thp.l(I,K,BH,SK);

tc.l(J,K) = TC_INT(J,K);
tcp.l(J,K,BC,SK) = TCP_INT(J,K,BC,SK);
th.l(I,K) = TH_INT(I,K);
```

```
thp.l(I,K,BH,SK) = THP_INT(I,K,BH,SK);

DTH_INT(I,J,K,BH,BC,SK) = dth.L(I,J,K,BH,BC,SK);
DTC_INT(I,J,K,BH,BC,SK) = dtc.L(I,J,K,BH,BC,SK);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);

dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcu.L(I);
QHU_INT(J) = qhu.L(J);
Q_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK);

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);
z.L(I,J,K,BH,BC,SK) = Z_P(I,J,K,BH,BC,SK);
zcu.L(I) = ZCU_P(I);
zhu.L(J) = ZHU_P(J);

*************************
NONLINEAR.optfile = 5;
option reslim = 100000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
*************************
$onecho>dicopt.op5
STOP 0
MAXCYCLES 300
```

```
mipoptfile cplex.opt 1
epsmip 500
$offecho
************************
SOLVE NONLINEAR USING MINLP MINIMIZING TAC5;
DISPLAY th.l,thp.l,tc.l,tcp.l,q.l,qcu.l,qhu.l,fhp.l,fcp.l,z.L,zcu.L,zhu.L,a.l,acu.l,ahu.l,
    TAC1.l,TAC2.l,TAC3.l,TAC4.l,TAC5.l,NEX.L,UC.L,AC.L;
```

Appendix D HEN Retrofit

```
SETS
      I     Hot stream /I1,I2/
      J     Cold stream /J1,J2/
      K     Major stage /KFIRST,K2*K3,KLAST/
      SK    Sub-stage /SKFIRST,SK2*SK3,SKLAST/
      BH    Branch of hot splitting stream /BH1*BH2/
      BC    Branch of hot splitting stream /BC1*BC3/
;

SCALARS
****************************************
      NoBH          Number of branch of hot splIttIng stream /2/
      NoBC          Number of branch of hot splIttIng stream /3/
      CUMIN         Minimum cold utility require  /200/
      CUMAX         Maximum cold utility require  /5100/
      HUMIN         Minimum hot utility require    /0/
      HUMAX         Maximum hot utility require    /4700/
****************************************
      EMAT          Exchange minimum approach temperature /10/
      TMIN          Minimum temperature in HEN /20/
      TMAX          Maximum temperature in HEN /170/
      AREA_MAX      Miximum area for retrofit HEN /375/
****** RETROFIT ******
      NEW_ACHX      Area cost coefficient of "NEW" process heat exchanger
/1300/
      NEW_ACCU      Area cost coefficient of "NEW" cold utility /1300/
      NEW_ACHU      Area cost coefficient of "NEW" hot utility /1300/
      NEW_CFHX      Fixed charges of "NEW" exchanger /3000/
      NEW_CFCU      Fixed charges of "NEW" cold utility /3000/
      NEW_CFHU      Fixed charges of "NEW" hot utility /3000/
```

```
      NEW_CCU       Per unit cost of cold utility /20/
      NEW_CHU       Per unit cost of hot utility /80/
;

PARAMETERS
*TEMPERATURE OF STREAM.....
      THIN(I)       Inlet temperature of hot stream
      /     I1      170
            I2      150   /
      THOUT(I)      Outlet temperature of hot stream
      /     I1      60
            I2      30    /
      TCIN(J)       Inlet temperature of cold stream
      /     J1      20
            J2      80    /
      TCOUT(J)      Outlet temperature of cold stream
      /     J1      135
            J2      140   /
      TCUIN   /20/
      TCUOUT  /40/
      THUIN   /177/
      THUOUT  /177/
*HEAT CAPACITY FLOWRATE OF PROCESS
STREAM.....
      FH(I)         Heat capacity flowrate of hot stream
      /     I1      30
            I2      15    /
      FC(J)         Heat capacity flowrate of cold stream
      /     J1      20
            J2      40    /
*
```

*BRANCH FLOW __ [PARAMETER].....

FHP_P(I,K,BH) Branch flow parameter of hot stream

FCP_P(J,K,BC) Branch flow parameter of cold stream

FBHPT_P(I,K)

FBCPT_P(J,K)

*BINARY PARAMETER __ [PARAMETER].....

Z_P(I,J,K,BH,BC,SK) Binary parameter of exchanger existence

ZH_P(I,K)

ZC_P(J,K)

ZCU_P(I) Binary parameter of cold utIllty existence

ZHU_P(J) Binary parameter of hot utIllty existence

NEW_Z_P(I,J,K,BH,BC,SK)

NEW_ZCU_P(I)

NEW_ZHU_P(J)

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....

QUP_P(I,J,K,BH,BC,SK) Upper bound of heat exchange

QLO_P(I,J,K,BH,BC,SK) Lower bound of heat exchange

*_____

_____

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....

OMEGA(I,J) Upper bound for heat exchange

HCT_P(I) Heat content of hot stream

CCT_P(J) Heat content of cold stream

GAMMA(I,J) Upper bound for temperature dIfference

GAMMAH(I) Upper bound for temperature dIfference of hot stream

GAMMAC(J) Upper bound for temperature dIfference of cold stream

BETA(I,J) exponent for area costs of HX I-J

BETACU(I) exponent for area costs of cooler

BETAHU(J) exponent far area costs of heater

*OVERALL HEAT TRANSFER COEFFICIENT

U(I,J) overall heat transfer coeff. of heat exchanger of I-J [KW*(m2*K)-1]

UCU(I) overall heat transfer coeff. of cooler

UHU(J) overall heat transfer coeff. of heater

*U(I,J) = [H(I)*H(J)]/[H(I)+H(J)]

DTCUP(I)

DTHUP(J)

*****RETROFIT*****

EX_Ai(I,J,K)

EX_ACUi(I)

EX_AHUi(J)

EX_Zi(I,J,K)

EX_ZCUi(I)

EX_ZHUi(J)

EX_Ai_MAX(I,J,K)

EX_qcui(I)

EX_qhui(J)

;

DTHUP(J) = THUIN-TCOUT(J);

DTCUP(I) = THOUT(I)-TCUIN;

BETA(I,J) = 0.6;

BETACU(I) = 0.6;

BETAHU(J) = 0.6;

U(I,J) = 0.8;

UCU(I) = 0.8;

UHU(J) = 0.8;

*****RETROFIT*****

EX_Ai('I2','J2','KFIRST') = 46.74;
EX_Ai('I2','J2','K2') = 0;
EX_Ai('I2','J2','K3') = 0;
EX_Ai('I2','J2','KLAST') = 0;
EX_Ai('I2','J1','K2') = 68.72;
EX_Ai('I2','J1','KFIRST') = 0;
EX_Ai('I2','J1','K3') = 0;
EX_Ai('I2','J1','KLAST') = 0;

EX_Ai('I1','J1','KFIRST') = 38.31;
EX_Ai('I1','J1','K2') = 0;
EX_Ai('I1','J1','K3') = 0;
EX_Ai('I1','J1','KLAST') = 0;
EX_Ai('I1','J2',K) = 0;

EX_ACUi('I1') = 40.23;
EX_ACUi('I2') = 0;

EX_AHUi('J2') = 35.0;
EX_AHUi('J1') = 0;

EX_Zi('I2','J2','KFIRST') = 1;
EX_Zi('I2','J2','K2') = 0;
EX_Zi('I2','J2','K3') = 0;
EX_Zi('I2','J2','KLAST') = 0;

EX_Zi('I2','J1','K2') = 1;
EX_Zi('I2','J1','KFIRST') = 0;
EX_Zi('I2','J1','K3') = 0;
EX_Zi('I2','J1','KLAST') = 0;

EX_Zi('I1','J1','KFIRST') = 1;

EX_Zi('I1','J1','K2') = 0;

EX_Zi('I1','J1','K3') = 0;

EX_Zi('I1','J1','KLAST') = 0;

EX_ZCUi('I1') = 1;

EX_ZCUi('I2') = 0;

EX_ZHUi('J2') = 1;

EX_ZHUi('J1') = 0;

EX_qcui('I1') = 1900;

EX_qhui('J2') = 1500;

VARIABLES

*TEMPERATURE OF PROCESS STREAM.....

th(I,K) Temperature of hot stream at stage K

thp(I,K,BH,SK) Temperature of hot stream at sub-stage SK In stage K

tc(J,K) Temperature of cold stream at stage K

tcp(J,K,BC,SK) Temperature of cold stream at sub-stage SK In stage K

thpi(I,J,K)

tcpi(I,J,K)

*HEAT CAPACITY FLOWRATE OF PROCESS STREAM.....

f(I,J,K)

g(I,J,K)

fhp(I,K,BH) Branch flow parameter of hot stream

fcp(J,K,BC) Branch flow parameter of cold stream

*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE (T*FCP).....

fhpt(I,K,BH,SK) MultIple of temperature and heat flow of hot stream

fcpt(J,K,BC,SK) MultIple of temperature and heat flow of cold stream

fbhpt(I,K)

fbcpt(J,K)

*HEAT EXCHANGE.....

qi(I,J,K)

qh(K,I,BH,SK) Heat exchange of hot splIttIng BH In sub-stage SK

qhKb(K,I,BH) Heat exchange of hot splIttIng BH at stage K

qhK(K,I) Heat exchange of hot (I-J) at stage K

qhc(K) Heat exchange of hot stream I In stage K

qc(K,J,BC,SK) Heat exchange of hot and cold (I-J) In sub-stage SK

qcKb(K,J,BC) Heat exchange of cold splIttIng BC at stage K

qcK(K,J) Heat exchange of cold (I-J) at stage K

qch(K) Heat exchange of cold stream I In stage K

TOTALqex Total heat exchange

TOTAL_HU Total hot utIlIty

TOTAL_CU Total cold utIlIty

*RETROFIT _____ "ADDITIONAL HEAT EXCHANGE"

add_qcui(I)

add_qhui(J)

add_qcu(I)

add_qhu(J)

*RETROFIT _____ "COST".....

counthxi

counthx

counthx_ri

counthx_r

costhx_ri
costhx_r
costhui
costhu
costhu_ri
costhu_r
costcui
costcu
costcu_ri
costcu_r
fixcosthxi
fixcosthx
addareai
addarea
costareaaddi
costareaadd
costareahui
costareahu
costareacui
costareacu
areacosti
areacost
capitalcosti
capitalcost

costi1
costi2
costi3
costii
cost1
cost2
cost3

OBJcost1

OBJcost2

OBJcost3

costi

;

BINARY VARIABLES

*EXISTENCE OF EXCHANGER.....

zi(I,J,K)

zcui(I)

zhui(J)

new_zi(I,J,K)

new_zcui(I)

new_zhui(J)

z(I,J,K,BH,BC,SK) ExIstence of exchanger I-J In each sK

zcu(I) ExIstence of cold utIlIty

zhu(J) ExIstence of hot utIlIty

new_z(I,J,K,BH,BC,SK)

new_zcu(I)

new_zhu(J)

;

POSITIVE VARIABLES

*TEMPERATURE APPROACH.....

dthi(I,J,K)

dtci(I,J,K)

dtcui(I)

dthui(J)

dth(I,J,K,BH,BC,SK) Temperature dIfference at "hot end" of exchanger

dtc(I,J,K,BH,BC,SK) Temperature dIfference at "cold end" of exchanger

dtcu(I) Temperature dIfference of cold utIIIty

dthu(J) Temperature dIfference of hot utIIIty

*AREA.....

*LOG MEAN TEMPERATURE DIFFERENCE.....

LMTDHX(I,J,K,BH,BC,SK) Log mean emperature dIfference of exchanger I-J

LMTCU(I) Log mean emperature dIfference of cold utIIIty

LMTHU(J) Log mean emperature dIfference of hot utIIIty

LMTDHXi(I,J,K)

LMTCUi(I)

LMTHUi(J)

*AREA.....

ai(I,J,K)

acui(I)

ahui(J)

a(I,J,K,BH,BC,SK) Heat exchange area of process exchanger

acu(I) Heat exchange area of cold utIIIty

ahu(J) Heat exchange area of hot utIIIty

*ADDITIONAL AREA.....

add_ai(I,J,K)

add_acui(I)

add_ahui(J)

add_a(I,J,K,BH,BC,SK)

add_acu(I)

add_ahu(J)

```
*HEAT EXCHANGE.....
      qi(I,J,K)
      qcui(I)
      qhui(J)
      q(I,J,K,BH,BC,SK)    Heat exchange between process stream I-J
      qcu(I)        Heat exchange of cold utIIIty
      qhu(J)        Heat exchange of hot utIIIty

*NUMBER OF HEAT EXCHANGER IN EACH STAGE K
      nexh(I,K)       ExIstence of exchanger I-J of stream I In each K
      nexc(J,K)       ExIstence of exchanger I-J of stream J In each K

SCALARS HI,CJ;
HI=1;
CJ=1;
*LOOP OF HOT PROCESS STREAM.....
FOR(HI=1 to CARD(I),
      HCT_P(I)$[ORD(I) = HI] = FH(I)*(THIN(I)-THOUT(I));
      GAMMAH(I)$[ORD(I) = HI] = THIN(I)-THOUT(I);
*TEMPERATURE
      th.lo(I,K)$[ORD(I) = HI] = THOUT(I);
      th.up(I,K)$[ORD(I) = HI] = THIN(I);
      thp.lo(I,K,BH,SK)$[ORD(I) = HI] = TMIN;
      thp.up(I,K,BH,SK)$[ORD(I) = HI] = TMAX;
*HEAT CAPACITY FLOWRATE
      fhp.lo(I,K,BH)$[ORD(I) = HI] = 0;
      fhp.up(I,K,BH)$[ORD(I) = HI] = FH(I);
*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE
(T*FCP)
      fhpt.lo(I,K,BH,SK)$[ORD(I) = HI] = 0;
      fhpt.up(I,K,BH,SK)$[ORD(I) = HI] = FH(I)*TMAX;
      fbhpt.lo(I,K)$[ORD(I) = HI] = 0;
```

```
        fbhpt.up(I,K)$[ORD(I) = HI] =FH(I)*THIN(I);
*HEAT EXCHANGE
        qh.lo(K,I,BH,SK)$[ORD(I) = HI] = 0;
        qh.up(K,I,BH,SK)$[ORD(I) = HI] = HCT_P(I);
        qhKb.lo(K,I,BH)$[ORD(I) = HI] = 0;
        qhKb.up(K,I,BH)$[ORD(I) = HI] = HCT_P(I);
        qhK.lo(K,I)$[ORD(I) = HI] = 0;
        qhK.up(K,I)$[ORD(I) = HI] = HCT_P(I);
        qhc.lo(K) = 0;
        qhc.up(K) = CUMAX;
        qcu.lo(I)$[ORD(I) = HI] = 0;
        qcu.up(I)$[ORD(I) = HI] = HCT_P(I);

        qcui.up(I)$[ORD(I) = HI] = 0;
        qcui.up(I)$[ORD(I) = HI] = HCT_P(I);
);


*LOOP OF COLD PROCESS STREAM.....
For(CJ=1 to CARD(J),
        CCT_P(J)$[ORD(J) = CJ] = FC(J)*(TCOUT(J)-TCIN(J));
        GAMMAC(J)$[ORD(J) = CJ] = TCOUT(J)-TCIN(J);
*TEMPERATURE
        tc.lo(J,K)$[ORD(J) = CJ] = TCIN(J);
        tc.up(J,K)$[ORD(J) = CJ] = TCOUT(J);
        tcp.lo(J,K,BC,SK)$[ORD(J) = CJ] = TMIN;
        tcp.up(J,K,BC,SK)$[ORD(J) = CJ] = TMAX;
*HEAT CAPACITY FLOWRATE
        fcp.lo(J,K,BC)$[ORD(J) = CJ] = 0;
        fcp.up(J,K,BC)$[ORD(J) = CJ] = FC(J);
*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE
(T*FCP)
```

```
        fcpt.lo(J,K,BC,SK)$[ORD(J) = CJ] = 0;
        fcpt.up(J,K,BC,SK)$[ORD(J) = CJ] = FC(J)*TMAX;
        fbcpt.lo(J,K)$[ORD(J) = CJ] = 0;
        fbcpt.up(J,K)$[ORD(J) = CJ] = FC(J)*TCOUT(J);
*HEAT EXCHANGE
        qc.lo(K,J,BC,SK)$[ORD(J) = CJ] = 0;
        qc.up(K,J,BC,SK)$[ORD(J) = CJ] = CCT_P(J);
        qcKb.lo(K,J,BC)$[ORD(J) = CJ] = 0;
        qcKb.up(K,J,BC)$[ORD(J) = CJ] = CCT_P(J);
        qcK.lo(K,J)$[ORD(J) = CJ] = 0;
        qcK.up(K,J)$[ORD(J) = CJ] = CCT_P(J);
        qch.lo(K) = 0;
        qch.up(K) = HUMAX;
        qhu.lo(J)$[ORD(J) = CJ] = 0;
        qhu.up(J)$[ORD(J) = CJ] = CCT_P(J);

        qhui.lo(J)$[ORD(J) = CJ] = 0;
        qhui.up(J)$[ORD(J) = CJ] = CCT_P(J);
);

*-------------------------------------------------------------------------
-------------

*..... BOUNDS .....

*-------------------------------------------------------------------------
-------------

qi.lo(I,J,K) = 0;
qi.up(I,J,K) = MIN(HCT_P(I),CCT_P(J));

f.lo(I,J,K) = 0;
```

```
f.up(I,J,K) = FH(I);
g.lo(I,J,K) = 0;
g.up(I,J,K) = FC(J);

thpi.lo(I,J,K) = TMIN;
thpi.up(I,J,K) = TMAX;
tcpi.lo(I,J,K) = TMIN;
tcpi.up(I,J,K) = TMAX;

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....
      QUP_P(I,J,K,BH,BC,SK) = MIN(HCT_P(I),CCT_P(J),MAX(0,(THIN(I)-
TCIN(J)-EMAT)*MIN(FH(I),FC(J))));
      QLO_P(I,J,K,BH,BC,SK) = 1*U(I,J)*EMAT;

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....
      OMEGA(I,J) = MIN(HCT_P(I),CCT_P(J));
      GAMMA(I,J) = MAX[0,(THIN(I)-TCIN(J)),(THIN(I)-
TCOUT(J)),(THOUT(I)-TCIN(J)),
            (THOUT(I)-TCOUT(J)),(TCIN(J)-THIN(I)),(TCIN(J)-THOUT(I)),
            (TCOUT(J)-THIN(I)),(TCOUT(J)-THOUT(I))];

*_____
_____

*..... BQUATION .....

*_____
_____

EQUATIONS

      thinassign(I) assignment of inlet hot temperatures
```

tcinassign(J) assignment of inlet cold temperature

hotk(K,I) heat balance for hot stream at each stage

coldk(K,J) heat balance for cold stream at each stage

qhnoniso(I,J,K)

qcnoniso(I,J,K)

thoutst(I,K)

tcoutst(J,K)

qcooler(I) cold utility load

qheater(J) hot utility load

hoteqi(I) overall heat balance for hot stream

coldeqi(J) overall heat balance for cold stream

logif(I,J,K)

logig(I,J,K)

sumf(I,K)

sumg(J,K)

monoth(I,K) monotonic decrease in temperature for hot stream at each stage

monothout(I) monotonic decrease in temperature for hot stream at outlet of superstructure

monotc(J,K) monotonic decrease in temperature for cold stream at each stage

monotcout(J) monotonic decrease in temperature for cold stream at outlet of superstructure

monothpmin(I,J,K)

monothpmax(I,J,K)

monotcpmin(I,J,K)

monotcpmax(I,J,K)

logiqi(I,J,K) logical constraint for qi(ijk)

logiqcu(I) logical constraint for qcu(i)

logiqhu(J) logical constraint for qhu(j)

dthimin(I,J,K) minimum temperature approach for hot end of HX i-j at stage k

dtcimin(I,J,K) minimum temperature approach for cold end of HX i-j at stage k

dtcuimin(I) minimum temperature approach for cooler

dthuimin(J) minimum temperature approach for heater

logidthi(I,J,K) logical constraint for dt(ijk) at inlet of stage k

logidtci(I,J,K) logical constraint for dt(ijk) at outlet of stage k

logidtcu(I) logical constraint for dtcu(i)

logidthu(J) logical constraint for dthu(j)

LMTHci(I,J,K)

LMTDi(I,J,K)

LMTCUci(I)

LMTDCUi(I)

LMTDHUi(J)

LMTHUci(J)

AREAi(I,J,K)

AREACUi(I)

AREAHUi(J)

add_areai_lo(I,J,K)

add_areai_hi(I,J,K)

add_areacui_lo(I)

add_areacui_hi(I)

add_areahui_lo(J)

add_areahui_hi(J)

add_hui(I)

add_cui(J)

add_hxi1(I,J,K)

add_hxi2(I,J,K)

add_cui1(I)

add_cui2(I)

add_hui1(J)

add_hui2(J)

numberHXi

numberHX_ri

hxcost_ri

hucosti

hucost_ri

cucosti

cucost_ri

fixhxcosti

areaaddi

areaaddcosti

areahucosti

areacucosti

totalareacosti

capcostperyri

obji1 objective function to be minimized

obji2

obji3

objii

```
**********************************************************************
**************
*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM
ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BH,BC)

* 1. OVERALL HEAT BALANCE FOR EACH
STREAM****************************************
HOTEQ(I)
COLDEQ(J)

* 2. HEAT BALANCE AT EACH STAGE
K*************************************************
HOTCOLD_K(K)

QHC_K(K)
QHK_K(K,I)
QCH_K(K)
QCK_K(K,J)

* 3. HEAT BALANCE AT EACH SUB-STAGE
SK*********************************************
*[VARIABLE]
QHK_SK(K,I)
SUMSK_QH(K,I,BH)
SUMJ_BC_Q(I,K,BH,SK)
QH_SK(I,K,BH,SK)
HOT_SK(I,K,BH,SK)
FHPT_SK(I,K,BH,SK)
QCK_SK(K,J)
SUMSK_QC(K,J,BC)
SUMI_BH_Q(J,K,BC,SK)
```

QC_SK(J,K.BC,SK)
COLD_SK(J,K,BC,SK)
FCPT_SK(J.K,BC,SK)
*[FLOW PARAMETER]
HOT_SK_P(I,K,BH,SK)
FHPT_SK_P(I,K,BH,SK)
COLD_SK_P(J,K,BC,SK)
FCPT_SK_P(J,K,BC,SK)

* 4. COLD/HOT
UTILITY**********
***
QCULOAD(I)
QHULOAD(J)

* 5. TEMPERATURE
ASSIGNMENT**********
ASSIGNTH_K(I)
ASSIGNTH_SK(I,K)
ASSIGNTH_BYPASSH(I,K)
HMIX(I,K)
ASSIGNTH_TSK(I,K,BH)
ASSIGNTC_K(J)
ASSIGNTC_SK(J,K)
ASSIGNTH_BYPASSC(J,K)
CMIX(J,K)
ASSIGNTC_TSK(J,K,BC)
*[FLOW PARAMETER]
ASSIGNTH_SK_P(I,K)
ASSIGNTH_BYPASSH_P(I,K)
HMIX_P(I,K)
ASSIGNTC_SK_P(J,K)

ASSIGNTH_BYPASSC_P(J,K)
CMIX_P(J,K)

* 7. TEMPERATURE
FEASIBILITY**********
THFEAS_K(I,K)
THPFEAS_SK(I,K,BH,SK)
THFEAS_KLAST(I)
THPFEAS_SK_MIN(I,K,BH,SK)
THPFEAS_SK_MAX(I,K,BH,SK)
TCFEAS_K(J,K)
TCPFEAS_SK(J,K,BC,SK)
TCFEAS_KFIRST(J)
TCPFEAS_SK_MIN(J,K,BC,SK)
TCPFEAS_SK_MAX(J,K,BC,SK)

* 8. FLOW
FEASIBILITY**********
****
*[VARIABLE]
SUMFHP(I,K)
SUMFCP(J,K)
*[FLOW PARAMETER]
SUMFHP_P(I,K)
SUMFCP_P(J,K)

* 9.LOGICAL CONSTRAINTS [HEAT EXCHANGE
,BRANCH+BYPASS]**********
**********HEAT EXCHANGE**********
*[Z-VARIABLE]
LOGq(I,J,K,BH,BC,SK)
LOGQCU(I)

LOGQHU(J)

*[Z-PARAMETER]

LOGQ_P(I,J,K,BH,BC,SK)

LOGQCU_P(I)

LOGQHU_P(J)

*10.OTHER CONSTRAINTS**********

**********MAXIMUM MATCHING**********

*[Z- VARIABLE]

CONZ1(I,K,BH,SK)

CONZ2(J,K,BC,SK)

CONZ3(I,J,K)

CONZH(I,K)

CONZC(J,K)

*[Z-PARAMETER]

CONZ1_P(I,K,BH,SK)

CONZ2_P(J,K,BC,SK)

CONZ3_P(I,J,K)

CONZH_P(I,K)

CONZC_P(J,K)

**********FLOW CONSTRAINT**********

*[VARIABLES]

CON_FLOWH1(I,K,BH)

CON_FLOWH2(I,K,BH)

CON_FLOWC1(J,K,BC)

CON_FLOWC2(J,K,BC)

*[FLOW PARAMETER]

CON_FLOWH1_F_P(I,K,BH)

CON_FLOWH2_F_P(I,K,BH)

CON_FLOWC1_F_P(J,K,BC)

CON_FLOWC2_F_P(J,K,BC)

*[MATCH (Z) -PARAMETER]

*CON_FLOWH3_Z_P(I,K,BH,SK)

*CON_FLOWC3_Z_P(J,K,BC,SK)

*[MATCH+FLOW (Z) -PARAMETER]

*CON_FLOWH3_ZF_P(I,K,BH,SK)

*CON_FLOWC3_ZF_P(J,K,BC,SK)

**********MASS BALANCE AT EACH STAGE*****

MASSK_H1(I,K)

MASSK_C1(J,K)

**********CONSTRAINT**********

CON_CU1

CON_CU2

CON_CU3(I)

CON_CU4

CON_HU1

CON_HU2

CON_HU3(J)

CON_HU4

CONQ_UP(I,J,K,BH,BC,SK)

CONQ_LO(I,J,K,BH,BC,SK)

CON_EX

* 11. CALCULATION OF APPROACH

TEMPERATURE**********

*[Z- VARIABLE]

DTHMIN(I,J,K,BH,BC,SK)

DTHMAX(I,J,K,BH,BC,SK)

DTCMIN(I,J,K,BH,BC,SK)

DTCMAX(I,J,K,BH,BC,SK)

DTCUMIN(I)
DTCUMAX(I)
DTHUMIN(J)
DTHUMAX(J)
*[Z- PARAMETER]
DTHMIN_P(I,J,K,BH,BC,SK)
DTHMAX_P(I,J,K,BH,BC,SK)
DTCMIN_P(I,J,K,BH,BC,SK)
DTCMAX_P(I,J,K,BH,BC,SK)

DTCUMIN_P(I)
DTCUMAX_P(I)
DTHUMIN_P(J)
DTHUMAX_P(J)

* 12. AREA
EQUATION**********
******
LMTD(I,J,K,BH,BC,SK)
LMTHc(I,J,K,BH,BC,SK)
LMTDCU(I)
LMTCUc(I)
LMTDHU(J)
LMTHUc(J)

AREA(I,J,K,BH,BC,SK)
AREACU(I)
AREAHU(J)

********** RETROFIT
****

add_area_lo(I,J,K,BH,BC,SK)
add_area_lo_P(I,J,K,BH,BC,SK)
add_area_hi(I,J,K,BH,BC,SK)
add_area_hi_P(I,J,K,BH,BC,SK)
add_areacu_lo(I)
add_areacu_lo_P(I)
add_areacu_hi(I)
add_areacu_hi_P(I)
add_areahu_lo(J)
add_areahu_lo_P(J)
add_areahu_hi(J)
add_areahu_hi_P(J)

add_hu(I)
add_cu(J)
add_hx1(I,J,K,BH,BC,SK)
add_hx1_P(I,J,K,BH,BC,SK)
add_hx2(I,J,K,BH,BC,SK)
add_hx2_P(I,J,K,BH,BC,SK)
add_cu1(I)
add_cu1_P(I)
add_cu2(I)
add_cu2_P(I)
add_hu1(J)
add_hu1_P(J)
add_hu2(J)
add_hu2_P(J)

numberHX

numberHX_r
hxcost_r
hucost
hucost_r
cucost
cucost_r
fixhxcost
areaadd
areaaddcost
areahucost
areacucost
totalareacost
capcostperyr

obj1 objective function to be minimized
obj2
obj3
OBJFN1
OBJFN2
OBJFN3
obji

;

****

****

****

****

hoteqi(I).. FH(I)*(THIN(I)-THOUT(I)) =e= sum((J,K),qi(I,J,K)) + qcui(I);

coldeqi(J).. FC(J)*(TCOUT(J)-TCIN(J)) =e= sum((I,K),qi(I,J,K)) + qhui(J);

hotK(K,I)$(ORD(K) NE CARD(K)).. FH(I)*(th(I,K) -
th(I,K+1)) =e= sum(J,qi(I,J,K)) ;

```
coldK(K,J)$(ORD(K) NE CARD(K)).. FC(J)*(tc(J,K) -
tc(J,K+1)) =e= sum(I,qi(I,J,K));

qcooler(I).. FH(I)*(th(I,'KLAST')-THOUT(I)) =e= qcui(I) ;
qheater(J).. FC(J)*(TCOUT(J)-tc(J,'KFIRST')) =e= qhui(J) ;

qhnoniso(I,J,K)$(ORD(K) NE
CARD(K)).. qi(I,J,K) =l= (f(I,J,K)+0.001)*(th(I,K)-thpi(I,J,K));
qcnoniso(I,J,K)$(ORD(K) NE
CARD(K)).. qi(I,J,K) =l= (g(I,J,K)+0.001)*(tcpi(I,J,K)-tc(J,K+1));

thoutst(I,K)$(ORD(K) NE
CARD(K)).. th(I,K+1)*FH(I) =l= sum(J,f(I,J,K)*thpi(I,J,K));
tcoutst(J,K)$(ORD(K) NE
CARD(K)).. tc(J,K)*FC(J) =l= sum(I,g(I,J,K)*tcpi(I,J,K));

logif(I,J,K)$(ORD(K) NE CARD(K)).. f(I,J,K) =l= FH(I);
logig(I,J,K)$(ORD(K) NE CARD(K)).. g(I,J,K) =l= FC(J);

sumf(I,K)$(ORD(K) NE CARD(K)).. sum(J,f(I,J,K)) =l= FH(I);
sumg(J,K)$(ORD(K) NE CARD(K)).. sum(I,g(I,J,K)) =l= FC(J);

monoth(I,K)$(ORD(K) NE CARD(K)).. th(I,K) =g= th(I,K+1);
monothpmax(I,J,K)$(ORD(K) NE
CARD(K)).. thpi(I,J,K) =g= TMIN;
monothpmin(I,J,K)$(ORD(K) NE
CARD(K)).. thpi(I,J,K) =l= TMAX;
monothout(I).. th(I,'KLAST') =g= THOUT(I);
monotc(J,K)$(ORD(K) NE CARD(K)).. tc(J,K) =g= tc(J,K+1);
monotcpmax(I,J,K)$(ORD(K) NE CARD(K)).. tcpi(I,J,K) =g= TMIN;
monotcpmin(I,J,K)$(ORD(K) NE CARD(K)).. tcpi(I,J,K) =l= TMAX;
monotcout(J) . tc(J,'KFIRST') =l= TCOUT(J);
```

```
thinassign(I)..   THIN(I)      =e=   th(I,'KFIRST');
tcinassign(J)..   TCIN(J)      =e=   tc(J,'KLAST');

logiqi(I,J,K)$(ORD(K) NE
CARD(K))..   qi(I,J,K)          =l=        OMEGA(I,J)*zi(I,J,K);
logiqcu(I)..        qcui(I)          =l=        HCT_P(I)*zcui(I);
logiqhu(J)..        qhui(J)          =l=        CCT_P(J)*zhui(J);

dthimin(I,J,K)$(ORD(K) NE CARD(K))..  dthi(I,J,K)        =g=        EMAT;
dtcimin(I,J,K)$(ORD(K) NE CARD(K))..  dtci(I,J,K)        =g=        EMAT;
dtcuimin(I)..   dtcui(I)         =g=        EMAT;
dthuimin(J)..   dthui(J)         =g=        EMAT;

logidthi(I,J,K)$(ORD(K) NE CARD(K))..   dthi(I,J,K)        =l=        th(I,K)-
tcpi(I,J,K)+GAMMA(I,J)*(1-zi(I,J,K));
logidtci(I,J,K)$(ORD(K) NE
CARD(K))..   dtci(I,J,K)         =l=        thpi(I,J,K)-
tc(J,K+1)+GAMMA(I,J)*(1-zi(I,J,K));
logidtcu(I)..       dtcui(I)         =l=        th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-zcui(I));
logidthu(J)..       dthui(J)         =l=        THUOUT-
tc(J,'KFIRST')+GAMMAC(J)*(1-zhui(J));

*LMTHci(I,J,K)$(ORD(K) NE CARD(K)) .. LMTDHXi(I,J,K) =L=
0.5*(dthi(I,J,K)+dtci(I,J,K));
LMTDi(I,J,K)$(ORD(K) NE CARD(K)).. LMTDHXi(I,J,K) =E=
(2/3)*((dthi(I,J,K)+0.0001)**0.5)*((dtci(I,J,K)+0.0001)**0.5)+(1/6)*dthi(I,J,K)+(1/
6)*dtci(I,J,K);
*LMTCUci(I) .. LMTCUi(I) =L= 0.5*(dtcui(I)+(th(I,'KLAST')-THOUT(I)));
LMTDCUi(I).. LMTCUi(I) =E=
(2/3)*((dtcui(I)+0.0001)**0.5)*((DTCUP(I)+0.0001)**0.5)+(1/6)*dtcui(I)+(1/6)*D
```

```
TCUP(I);

LMTDHUi(J).. LMTHUi(J) =E=
(2/3)*((dthui(J)+0.0001)**0.5)*((DTHUP(J)+0.0001)**0.5)+(1/6)*dthui(J)+(1/6)*D
THUP(J);
*LMTHUci(J) .. LMTHUi(J) =L= 0.5*(dthui(J)+(TCOUT(J)- tc(J,'KFIRST')));

AREAi(I,J,K)$(ORD(K) NE CARD(K))..qi(I,J,K)-
((ai(I,J,K)+0.0001)**(1/BETA(I,J))*U(I,J)*LMTDHXi(I,J,K)) =L= 0;
AREACUi(I)..    qcui(I)-((acui(I)+0.0001)**(1/BETACU(I))*UCU(I)*LMTCUi(I))
=l=    0;
AREAHUi(J)..    qhui(J)-
((ahui(J)+0.0001)**(1/BETAHU(J))*UHU(J)*LMTHUi(J)) =l=    0;

add_areai_lo(I,J,K)$(ORD(K) NE CARD(K)).. add_ai(I,J,K)-
(0.0001**(1/0.6)+4.308869E-7)   =L=    AREA_MAX-
(ai(I,J,K)+0.0001)**(1/BETA(I,J))+EX_Ai(I,J,K)*zi(I,J,K);
add_areai_hi(I,J,K)$(ORD(K) NE CARD(K)).. add_ai(I,J,K)-
(0.0001**(1/0.6)+4.308869E-7)   =G=    (ai(I,J,K)+0.0001)**(1/BETA(I,J))-
EX_Ai(I,J,K)*zi(I,J,K);
add_areacui_lo(I)..     add_acui(I)-(0.0001**(1/0.6)+4.308869E-
7)   =L=    AREA_MAX-
(acui(I)+0.0001)**(1/BETACU(I))+EX_ACUi(I)*zcui(I);
add_areacui_hi(I)..     add_acui(I)-(0.0001**(1/0.6)+4.308869E-
7)   =G=    (acui(I)+0.0001)**(1/BETACU(I))-EX_ACUi(I)*zcui(I);
add_areahui_lo(J)..     add_ahui(J)-(0.0001**(1/0.6)+4.308869E-
7)   =L=    AREA_MAX-
(ahui(J)+0.0001)**(1/BETAHU(J))+EX_AHUi(J)*zhui(J);
add_areahui_hi(J)..     add_ahui(J)-(0.0001**(1/0.6)+4.308869E-
7)   =G=    (ahui(J)+0.0001)**(1/BETAHU(J))-EX_AHUi(J)*zhui(J);

add_hui(I)..   add_qcui(I)   =G=    qcui(I)-EX_qcui(I)*EX_Zcui(I);
```

```
add_cui(J)..    add_qhui(J)    =G=    qhui(J)-EX_qhui(J)*EX_Zhui(J);

add_hxi1(I,J,K)$(ORD(K) NE CARD(K)).. zi(I,J,K)-
EX_zi(I,J,K) =E=    new_zi(I,J,K);
add_hxi2(I,J,K)$(ORD(K) NE
CARD(K)).. EX_zi(I,J,K)+new_zi(I,J,K)      =L=    1;
add_cui1(I)..    zcui(I)      =E=    EX_zcui(I)+new_zcui(I);
add_cui2(I)..    EX_zcui(I)+new_zcui(I)      =L=    1;
add_hui1(J)..    zhui(J)      =E=    EX_zhui(J)+new_zhui(J);
add_hui2(J)..    EX_zhui(J)+new_zhui(J)      =L=    1;

numberHXi..    counthxi      =e=    sum((I,J,K),zi(I,J,K))+sum(I,zcui(I))+sum(J,z
hui(J));
numberHX_ri..    counthx_ri      =e=    sum((I,J,K),new_zi(I,J,K))+sum(I,new_zcui
(I))+sum(J,new_zhui(J));
hxcost_ri..    costhx_ri      =e=    sum((I,J,K),NEW_CFHX*new_zi(I,J,K));
hucosti..    costhui      =e=    sum(J,NEW_CHU*qhui(J));
hucost_ri..    costhu_ri      =e=    sum(J,NEW_CHU*(EX_qhui(J)+add_qhui(J)));
cucosti..    costcui      =e=    sum(I,NEW_CCU*qcui(I));
cucost_ri..    costcu_ri      =e=    sum(I,NEW_CCU*(EX_qcui(I)+add_qcui(I)));
fixhxcosti..    fixcosthxi      =e=    NEW_CFHX*counthx_ri;

areaaddi..    addareai      =e=    sum((I,J,K),add_ai(I,J,K));
areaaddcosti..    costareaaddi      =e=    sum((I,J,K),NEW_ACHX*(add_ai(I,J,K)+0.0
001)**(BETA(I,J)));
areahucosti..    costareahui      =e=    sum(J,NEW_ACHU*(add_ahui(J)+0.0001)**(
BETAHU(J)));
areacucosti..    costareacui      =e=    sum(I,NEW_ACCU*(add_acui(I)+0.0001)**(B
ETACU(I)));
totalareacosti..    areacosti      =e=    costareaaddi+costareahui+costareacui;
capcostperyri..    capitalcosti      =e=    (fixcosthxi+areacosti);
*obji1 :  H+C utility cost
```

```
*obji2 :  Fixed cost of "NEW" HX
*obji3 :  Additional area cost (assume 5 yrs)
*objii : Total annual cost
obji1..          costi1   =e=       costhu_ri+costcu_ri;
obji2..          costi2   =e=       (fixcosthxi);
obji3..          costi3   =e=       (costareaaddi+costareahui+costareacui);
objii..          costii   =e=       costi1+costi2+costi3;

************************** END RETROFIT
|************************************

*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM
ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BH,BC)..  q(I,J,K,BH,BC,'SKLAST') =E=    0;

* 1. OVERALL HEAT BALANCE FOR EACH
STREAM*****************************************
HOTEQ(I)..      FH(I)*[THIN(I)-
THOUT(I)]       =E=    SUM((K,BH,J,BC,SK)$[(ORD(K) NE
CARD(K))AND(ORD(SK) NE CARD(SK))],q(I,J,K,BH,BC,SK)) + qcu(I);
COLDEQ(J)..     FC(J)*[TCOUT(J)-
TCIN(J)]        =E=    SUM((K,BC,I,BH,SK)$[(ORD(K) NE
CARD(K))AND(ORD(SK) NE CARD(SK))],q(I,J,K,BH,BC,SK)) + qhu(J);

* 2. HEAT BALANCE AT EACH STAGE
K*************************************************
QHC_K(K)$[ORD(K) NE CARD(K)]..          qhc(K)  =E=    SUM(I,qhK(K,I));
QHK_K(K,I)$[ORD(K) NE CARD(K)]..        FH(I)*[th(I,K) -
th(I,K+1)]     =E=    qhK(K,I);
QHK_SK(K,I)$[ORD(K) NE
CARD(K)]..              qhK(K,I)        =E=    SUM(BH,qhKb(K,I,BH));
SUMJ_BC_Q(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
```

```
CARD(SK))].. qh(K,I,BH,SK) =E= SUM((J,BC),q(I,J,K,BH,BC,SK));
SUMSK_QH(K,I,BH)$[ORD(K) NE
CARD(K)].. qhKb(K,I,BH) =E= SUM(SK$(ORD(SK) NE
CARD(SK)),qh(K,I,BH,SK));
QCH_K(K)$[ORD(K) NE CARD(K)].. qch(K) =E= SUM(J,qcK(K,J));
QCK_K(K,J)$[ORD(K) NE CARD(K)].. FC(J)*[tc(J,K) -
tc(J,K+1)] =E= qcK(K,J);
QCK_SK(K,J)$[ORD(K) NE
CARD(K)].. qcK(K,J) =E= SUM(BC,qcKb(K,J,BC));
SUM1_BH_Q(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qc(K,J,BC,SK) =E= SUM((I,BH),q(I,J,K,BH,BC,SK));
SUMSK_QC(K,J,BC)$[ORD(K) NE
CARD(K)].. qcKb(K,J,BC) =E= SUM(SK$(ORD(SK) NE
CARD(SK)),qc(K,J,BC,SK));

HOTCOLD_K(K)$[ORD(K) NE CARD(K)].. qhc(K) =E= qch(K);

* 3. HEAT BALANCE AT EACH SUB-STAGE
SK*****************************************
*[VARIABLE]
QH_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qh(K,I,BH,SK) =E= fhpt(I,K,BH,SK) - fhpt(I,K,BH,SK+1);
HOT_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fhpt(I,K,BH,SK) -
fhpt(I,K,BH,SK+1) =E= fhp(I,K,BH)*[thp(I,K,BH,SK)-thp(I,K,BH,SK+1)];
FHPT_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fhp(I,K,BH)*thp(I,K,BH,SK) =E= fhpt(I,K,BH,SK);
QC_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. qc(K,J,BC,SK) =E= fcpt(J,K,BC,SK) -
fcpt(J,K,BC,SK+1);
COLD_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fcpt(J,K,BC,SK) -
```

```
fcpt(J,K,BC,SK+1)     =E=    fcp(J,K,BC)*[tcp(J,K,BC,SK)-tcp(J,K,BC,SK+1)];
FCPT_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. fcp(J,K,BC)*tcp(J,K,BC,SK)     =E=    fcpt(J,K,BC,SK);
*[FLOW PARAMETER]
HOT_SK_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FHP_P(I,K,BH)*[thp(I,K,BH,SK)-
thp(I,K,BH,SK+1)]     =E=    fhpt(I,K,BH,SK) - fhpt(I,K,BH,SK+1);
FHPT_SK_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FHP_P(I,K,BH)*thp(I,K,BH,SK)    =E=    fhpt(I,K,BH,SK);
COLD_SK_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FCP_P(J,K,BC)*[tcp(J,K,BC,SK)-
tcp(J,K,BC,SK+1)]     =E=    fcpt(J,K,BC,SK) - fcpt(J,K,BC,SK+1);
FCPT_SK_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. FCP_P(J,K,BC)*tcp(J,K,BC,SK)    =E=    fcpt(J,K,BC,SK);

* 4. COLD/HOT
UTILITY**********************************************************
***
QCULOAD(I).. qcu(I)       =E=    [th(I,'KLAST')-THOUT(I)]*FH(I) ;
QHULOAD(J).. qhu(J)       =E=    [TCOUT(J)-tc(J,'KFIRST')]*FC(J) ;

* 5. TEMPERATURE
ASSIGNMENT*****************************************************
ASSIGNTH_K(I)..                    THIN(I)       =E=    th(I,'KFIRST');
ASSIGNTH_SK(I,K)$[ORD(K) NE
CARD(K)]..          FH(I)*th(I,K)  =E=    SUM(BH,fhpt(I,K,BH,'SKFIRST'))+
fbhpt(I,K);
ASSIGNTH_BYPASSH(I,K)$[ORD(K) NE
CARD(K)]..          fbhpt(I,K)     =E=    (FH(I)-SUM(BH,fhp(I,K,BH)))*th(I,K);
HMIX(I,K)$[ORD(K) NE
CARD(K)]..          FH(I)*th(I,K+1) =E=    SUM(BH,fhpt(I,K,BH,'SKLAS
T'))+fbhpt(I,K);
```

```
ASSIGNTH_TSK(I,K,BH)$[ORD(K) NE CARD(K)]
..          TH(I,K)       =E=    thp(I,K,BH,'SKFIRST');
ASSIGNTC_K(J)..                              TCIN(J)       =E=    tc(J,'KLAST');
ASSIGNTC_SK(J,K)$[ORD(K) NE
CARD(K)]..              FC(J)*tc(J,K+1) =E=    SUM(BC,fcpt(J,K,BC,'SKLAST'))
+fbcpt(J,K+1);
ASSIGNTH_BYPASSC(J,K)$[ORD(K) NE
CARD(K)]..          fbcpt(J,K+1)    =E=    (FC(J)-
SUM(BC,fcp(J,K,BC)))*tc(J,K+1);
CMIX(J,K)$[ORD(K) NE
CARD(K)]..                FC(J)*tc(J,K)   =E=    SUM(BC,fcpt(J,K,BC,'SKFIRS
T'))+fbcpt(J,K+1);
ASSIGNTC_TSK(J,K,BC)$[ORD(K) NE CARD(K)]
..          TC(J,K+1)       =E=    tcp(J,K,BC,'SKLAST');

*[FLOW PARAMETER]
ASSIGNTH_SK_P(I,K)$[ORD(K) NE
CARD(K)]..          FH(I)*th(I,K)   =E=    SUM(BH,FHP_P(I,K,BH)*thp(I,K,BH,'S
KFIRST'))+FBHPT_P(I,K);
ASSIGNTH_BYPASSH_P(I,K)$[ORD(K) NE
CARD(K)]..          FBHPT_P(I,K)    =E=    (FH(I)-
SUM(BH,FHP_P(I,K,BH)))*th(I,K);
HMIX_P(I,K)$[ORD(K) NE
CARD(K)]..                FH(I)*th(I,K+1) =E=    SUM(BH,FHP_P(I,K,BH)*thp(I
,K,BH,'SKLAST'))+(FH(I)-SUM(BH,FHP_P(I,K,BH)))*th(I,K);
ASSIGNTC_SK_P(J,K)$[ORD(K) NE
CARD(K)]..          FC(J)*tc(J,K+1) =E=    SUM(BC,FCP_P(J,K,BC)*tcp(J,K,BC,'
SKLAST'))+FBCPT_P(J,K+1);
ASSIGNTH_BYPASSC_P(J,K)$[ORD(K) NE
CARD(K)]..          FBCPT_P(J,K+1)  =E=    (FC(J)-
SUM(BC,FCP_P(J,K,BC)))*tc(J,K+1);
CMIX_P(J,K)$[ORD(K) NE
```

```
CARD(K)]..          FC(J)*tc(J,K)  =E=  SUM(BC,FCP_P(J,K,BC)*tcp(J,
K,BC,'SKFIRST'))+(FC(J)-SUM(BC,FCP_P(J,K,BC)))*tc(J,K+1);

**********FLOW CONSTRAINT**********
*[VARIABLES]
CON_FLOWH1(I,K,BH)$[ORD(K) NE
CARD(K)]..      fhp(I,K,BH)  =L=  FH(I);
CON_FLOWH2(I,K,BH)$[ORD(K) NE CARD(K)]..      fhp(I,K,BH)  =G=  0;
CON_FLOWC1(J,K,BC)$[ORD(K) NE
CARD(K)]..      fcp(J,K,BC)  =L=  FC(J);
CON_FLOWC2(J,K,BC)$[ORD(K) NE CARD(K)]..      fcp(J,K,BC)  =G=  0;
*[FLOW PARAMETER]
CON_FLOWH1_F_P(I,K,BH)$[ORD(K) NE
CARD(K)]..   FHP_P(I,K,BH)  =L=  FH(I);
CON_FLOWH2_F_P(I,K,BH)$[ORD(K) NE
CARD(K)]..   FHP_P(I,K,BH)  =G=  0;
CON_FLOWC1_F_P(J,K,BC)$[ORD(K) NE
CARD(K)]..   FCP_P(J,K,BC)  =L=  FC(J);
CON_FLOWC2_F_P(J,K,BC)$[ORD(K) NE
CARD(K)]..   FCP_P(J,K,BC)  =G=  0;
*[ MATCH (Z) -PARAMETER]
*CON_FLOWH3_Z_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))$(ORD(SK) NE
CARD(SK))]..      fhp(I,K,BH)  =L=  SUM((J,BC),Z_P(I,J,K,BH,BC,SK))*FH(
I);
*CON_FLOWC3_Z_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))$(ORD(SK) NE
CARD(SK))]..      fcp(J,K,BC)  =L=  SUM((I,BH),Z_P(I,J,K,BH,BC,SK))*FC(
J);
*[ MATCH+FLOW (Z) -PARAMETER]
*CON_FLOWH3_ZF_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))$(ORD(SK) NE
CARD(SK))]..      FHP_P(I,K,BH)  =L=  SUM((J,BC),Z_P(I,J,K,BH,BC,SK))*
FH(I);
*CON_FLOWC3_ZF_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))$(ORD(SK) NE
```

```
CARD(SK))]..      FCP_P(J,K,BC)   =L=   SUM((I,BH),Z_P(I,J,K,BH,BC,SK))*
FC(J);

* 6. TEMPERATURE
FEASIBILITY**********************************************************
THFEAS_K(I,K)$[ORD(K) NE CARD(K)]..                th(I,K) =G=   th(I,K+1);
THPFEAS_SK(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..      thp(I,K,BH,SK)   =G=   thp(I,K,BH,SK+1);
THFEAS_KLAST(I)..                          th(I,'KLAST')   =G=   THOUT(I);
TCFEAS_K(J,K)$[ORD(K) NE CARD(K)]..                tc(J,K) =G=   tc(J,K+1);
TCPFEAS_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..      tcp(J,K,BC,SK)   =G=   tcp(J,K,BC,SK+1);
TCFEAS_KFIRST(J)..                         tc(J,'KFIRST')   =L=   TCOUT(J);

* 7. FLOW
FEASIBILITY**********************************************************
********
*[VARIABLE]
SUMFHP(I,K)$[ORD(K) NE
CARD(K)]..      SUM(BH,fhp(I,K,BH))        =L=   FH(I);
SUMFCP(J,K)$[ORD(K) NE
CARD(K)]..      SUM(BC,fcp(J,K,BC))        =L=   FC(J);
*[FLOW PARAMETER]
SUMFHP_P(I,K)$[ORD(K) NE
CARD(K)]..    SUM(BH,FHP_P(I,K,BH))        =L=   FH(I);
SUMFCP_P(J,K)$[ORD(K) NE
CARD(K)]..    SUM(BC,FCP_P(J,K,BC))        =L=   FC(J);


* 8.LOGICAL CONSTRAINTS [HEAT EXCHANGE
,BRANCH+BYPASS]*****************************
***********HEAT EXCHANGE***********
```

```
*[ Z-VARIABLE]
LOGq(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. q(I,J,K,BH,BC,SK) - OMEGA(I,J)*z(I,J,K,BH,BC,SK)  =L=  0;
LOGQCU(I).. qcu(I) - HCT_P(I)*zcu(I)  =L=  0;
LOGQHU(J).. qhu(J) - CCT_P(J)*zhu(J)  =L=  0;
*[Z-PARAMETER]
LOGQ_P(I,J.K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. q(I,J,K,BH,BC,SK) -
OMEGA(I,J)*Z_P(I,J,K,BH,BC,SK)  =L=  0;
LOGQCU_P(I).. qcu(I) - HCT_P(I)*ZCU_P(I)  =L=  0;
LOGQHU_P(J).. qhu(J) - CCT_P(J)*ZHU_P(J)  =L=  0;

*9.OTHER
CONSTRAINTS**************************************************
********
**********MAXIMUM MATCHING**********
*[ Z- VARIABLE]
CONZ1(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))] .  SUM((J,BC),z(I,J,K,BH,BC,SK))  =L=  1;
CONZ2(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((I,BH),z(I,J,K,BH,BC,SK))  =L=  1;
CONZ3(I,J,K)$[ORD(K) NE CARD(K)].. nexh(I,K)  =E=  nexc(J,K);
CONZH(I,K)$[ORD(K) NE CARD(K)]..
nexh(I,K)  =E=  SUM((J,SK,BH,BC),z(I,J,K,BH,BC,SK));
CONZC(J,K)$[ORD(K) NE CARD(K)]..
nexc(J,K)  =E=  SUM((I,SK,BH,BC),z(I,J,K,BH,BC,SK));
*[ Z-PARAMETER]
CONZ1_P(I,K,BH,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((J,BC),Z_P(I,J,K,BH,BC,SK))  =L=  1;
CONZ2_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((I,BH),Z_P(I,J,K,BH,BC,SK))  =L=  1;
CONZ3_P(I,J,K)$[ORD(K) NE CARD(K)].. ZH_P(I,K)  =E=  ZC_P(J,K);
```

```
CONZH_P(I,K)$[ORD(K) NE
CARD(K)].. ZH_P(I,K) =E= SUM((J,SK,BH,BC),Z_P(I,J,K,BH,BC,SK));
CONZC_P(J,K)$[ORD(K) NE
CARD(K)].. ZC_P(J,K) =E= SUM((I,SK,BH,BC),Z_P(I,J,K,BH,BC,SK));

*MASS BALANCE AT EACH STAGE
MASSK_H1(I,K)$(ORD(K) NE CARD(K)).. SUM(BH,fhp(I,K,BH))+(FH(I)-
SUM(BH,fhp(I,K,BH))) =E= FH(I);
MASSK_C1(J,K)$(ORD(K) NE CARD(K)).. SUM(BC,fcp(J,K,BC))+(FC(J)-
SUM(BC,fcp(J,K,BC))) =E= FC(J);

* Heat Exchange and Heat & Cold utIlIty constraInts.....
* Cold utIlIty....
CON_CU1 .. SUM(I,qcu(I))=G= CUMIN;
CON_CU2 .. SUM(I,qcu(I))=L= CUMAX;
CON_CU3(I) .. qcu(I) =L= zcu(I)*HCT_P(I);
CON_CU4 .. TOTAL_CU =E= SUM(I,qcu(I));
* Hot utIlIty....
CON_HU1 .. SUM(J,qhu(J))=G= HUMIN;
CON_HU2 .. SUM(J,qhu(J))=L= HUMAX;
CON_HU3(J) .. qhu(J) =L= zhu(J)*CCT_P(J);
CON_HU4 .. TOTAL_HU =E= SUM(J,qhu(J));

**********HEAT EXCHANGE**********
*[ Z- VARIABLE]
CON_EX.. SUM((I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))],q(I,J,K,BH,BC,SK)) =G= HUMAX-TOTAL_HU;
CONQ_UP(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]
.. q(I,J,K,BH,BC,SK) =L= z(I,J,K,BH,BC,SK)*QUP_P(I,J,K,BH,BC,SK);
CONQ_LO(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]
```

```
..  q(I,J,K,BH,BC,SK) =G=  z(I,J,K,BH,BC,SK)*QLO_P(I,J,K,BH,BC,SK);

* 10. CALCULATION OF APPROACH
TEMPERATURE**************************************
*[ Z- VARIABLE]
DTHMAX(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  dth(I,J,K,BH,BC,SK) =L=  thp(I,K,BH,SK)-
tcp(J,K,BC,SK)+GAMMA(I,J)*(1-z(I,J,K,BH,BC,SK));
DTCMAX(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  dtc(I,J,K,BH,BC,SK) =L=  thp(I,K,BH,SK+1)-
tcp(J,K,BC,SK+1)+GAMMA(I,J)*(1-z(I,J,K,BH,BC,SK));
DTCUMAX(I)..  dtcu(I)  =L=  th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-zcu(I));
DTHUMAX(J)..  dthu(J)  =L=  THUOUT-
tc(J,'KFIRST')+GAMMAC(J)*(1-zhu(J));

DTHMIN(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  dth(I,J,K,BH,BC,SK) =G=  EMAT;
DTCMIN(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  dtc(I,J,K,BH,BC,SK) =G=  EMAT;
DTCUMIN(I)..  dtcu(I)  =G=  EMAT;
DTHUMIN(J)..  dthu(J)  =G=  EMAT;

*[ Z- PARAMETER]
DTHMAX_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))]..  dth(I,J,K,BH,BC,SK) =L=  thp(I,K,BH,SK)-
tcp(J,K,BC,SK)+GAMMA(I,J)*(1-Z_P(I,J,K,BH,BC,SK));
DTCMAX_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))]..  dtc(I,J,K,BH,BC,SK) =L=  thp(I,K,BH,SK+1)-
tcp(J,K,BC,SK+1)+GAMMA(I,J)*(1-Z_P(I,J,K,BH,BC,SK));
DTCUMAX_P(I)..  dtcu(I)  =L=  th(I,'KLAST')-
TCUOUT+GAMMAH(I)*(1-ZCU_P(I));
```

```
DTHUMAX_P(J)..          dthu(J)      =L=    THUOUT-
tc(J,'KFIRST')+GAMMAC(J)*(1-ZHU_P(J));

DTHMIN_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))]..     dth(I,J,K,BH,BC,SK)  =G=    EMAT;
DTCMIN_P(I,J,K,BH,BC,SK)$[(ORD(SK) NE CARD(SK)) AND (ORD(K) NE
CARD(K))]..     dtc(I,J,K,BH,BC,SK)  =G=    EMAT;
DTCUMIN_P(I)..          dtcu(I)      =G=    EMAT;
DTHUMIN_P(J)..          dthu(J)      =G=    EMAT;

* 11. AREA
EQUATION**************************************************************
******
LMTHc(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))] .. LMTDHX(I,J,K,BH,BC,SK) =L=
0.5*(dth(I,J,K,BH,BC,SK)+dtc(I,J,K,BH,BC,SK));
LMTD(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. LMTDHX(I,J,K,BH,BC,SK) =E=
                    (2/3)*((dth(I,J,K,BH,BC,SK)+0.001)**0.5)*((dtc(I,J,K,BH,BC,
SK)+0.001)**0.5)+(1/6)*dth(I,J,K,BH,BC,SK)+(1/6)*dtc(I,J,K,BH,BC,SK);
LMTCUc(I) .. LMTCU(I) =L= 0.5*(dtcu(I)+(th(I,'KLAST')-THOUT(I)));
LMTDCU(I).. LMTCU(I) =E=
(2/3)*((dtcu(I)+0.001)**0.5)*((DTCUP(I)+0.001)**0.5)+(1/6)*dtcu(I)+(1/6)*DTCU
P(I);

LMTDHU(J).. LMTHU(J) =E=
(2/3)*((dthu(J)+0.001)**0.5)*((DTHUP(J)+0.001)**0.5)+(1/6)*dthu(J)+(1/6)*DTH
UP(J);
LMTHUc(J) .. LMTHU(J) =L= 0.5*(dthu(J)+(TCOUT(J)- tc(J,'KFIRST')));

AREA(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..q(I,J,K,BH,BC,SK)-
```

```
(a(I,J,K,BH,BC,SK)**(1/BETA(I,J))*U(I,J)*LMTDHX(I,J,K,BH,BC,SK)) =L= 0;
AREACU(I)..     qcu(I)-(acu(I)**(1/BETACU(I))*UCU(I)*LMTCU(I)) =l=    0;
AREAHU(J)..     qhu(J)-(ahu(J)**(1/BETAHU(J))*UHU(J)*LMTHU(J)) =l=    0;

************************** RETROFIT
*****************************************

add_area_lo(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BH,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
                        =L=    AREA_MAX-
(a(I,J,K,BH,BC,SK)+0.0001)**(1/BETA(I,J))+EX_Ai(I,J,K)*z(I,J,K,BH,BC,SK);
add_area_lo_P(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BH,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
                        =L=    AREA_MAX-
(a(I,J,K,BH,BC,SK)+0.0001)**(1/BETA(I,J))+EX_Ai(I,J,K)*Z_P(I,J,K,BH,BC,SK)
;
add_area_hi(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BH,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
                        =G=    (a(I,J,K,BH,BC,SK)+0.0001)**(1/BETA(I,J))-
EX_Ai(I,J,K)*z(I,J,K,BH,BC,SK);
add_area_hi_P(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BH,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
                        =G=    (a(I,J,K,BH,BC,SK)+0.0001)**(1/BETA(I,J))-
EX_Ai(I,J,K)*Z_P(I,J,K,BH,BC,SK);

add_areacu_lo(I)..      add_acu(I)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-(acu(I)+0.0001)**(1/BETACU(I))+EX_ACUi(I)*zcu(I);
add_areacu_lo_P(I)..    add_acu(I)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-
(acu(I)+0.0001)**(1/BETACU(I))+EX_ACUi(I)*ZCU_P(I);
add_areacu_hi(I)..      add_acu(I)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (acu(I)+0.0001)**(1/BETACU(I))-EX_ACUi(I)*zcu(I);
```

```
add_areacu_hi_P(I)..    add_acu(I)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (acu(I)+0.0001)**(1/BETACU(I))-EX_ACUi(I)*ZCU_P(I);
add_areahu_lo(J)..      add_ahu(J)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-
(ahu(J)+0.0001)**(1/BETAHU(J))+EX_AHUi(J)*zhu(J);
add_areahu_lo_P(J)..    add_ahu(J)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-
(ahu(J)+0.0001)**(1/BETAHU(J))+EX_AHUi(J)*ZHU_P(J);
add_areahu_hi(J)..      add_ahu(J)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (ahu(J)+0.0001)**(1/BETAHU(J))-EX_AHUi(J)*zhu(J);
add_areahu_hi_P(J)..    add_ahu(J)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (ahu(J)+0.0001)**(1/BETAHU(J))-EX_AHUi(J)*ZHU_P(J);

add_hu(I)..    add_qcu(I)    =G=    qcu(I)-EX_qcui(I)*EX_Zcui(I);
add_cu(J)..    add_qhu(J)    =G=    qhu(J)-EX_qhui(J)*EX_Zhui(J);

add_hx1(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  z(I,J,K,BH,BC,SK)-
EX_zi(I,J,K)        =E=    new_z(I,J,K,BH,BC,SK);
add_hx1_P(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  Z_P(I,J,K,BH,BC,SK)-
EX_zi(I,J,K)        =E=    NEW_Z_P(I,J,K,BH,BC,SK);
add_hx2(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  EX_zi(I,J,K)+new_z(I,J,K,BH,BC,SK)    =L=    1;
add_hx2_P(I,J,K,BH,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  EX_zi(I,J,K)+NEW_Z_P(I,J,K,BH,BC,SK)    =L=    1;
add_cu1(I)..    zcu(I)                =E=    EX_zcui(I)+new_zcu(I);
add_cu1_P(I)..  ZCU_P(I)              =E=    EX_zcui(I)+NEW_ZCU_P(I);
add_cu2(I)..    EX_zcui(I)+new_zcu(I)      =L=    1;
add_cu2_P(I)..  EX_zcui(I)+NEW_ZCU_P(I)    =L=    1;
add_hu1(J)..    zhu(J)                =E=    EX_zhui(J)+new_zhu(J);
add_hu1_P(J)..  ZHU_P(J)              =E=    EX_zhui(J)+NEW_ZHU_P(J);
```

```
add_hu2(J)..    EX_zhui(J)+new_zhu(J)      =L=    1;
add_hu2_P(J)..  EX_zhui(J)+NEW_ZHU_P(J)    =L=    1;

numberHX..      counthx      =E=  sum((I,J,K,BH,BC,SK),z(I,J,K,BH,BC,SK))+s
um(I,zcu(I))-sum(J,zhu(J));
numberHX_r..    counthx_r    =E=  sum((I,J,K,BH,BC,SK),new_z(I,J,K,BH,BC,
SK))+sum(I,new_zcu(I))+sum(J,new_zhu(J));
hxcost_r..      costhx_r     =E=  sum((I,J,K,BH,BC,SK),NEW_CFHX*new_z(I,J,
K,BH,BC,SK));
hucost..        costhu       =E=  sum(J,NEW_CHU*qhu(J));
hucost_r..      costhu_r     =E=  sum(J,NEW_CHU*(EX_qhui(J)+add_qhu(J)));
cucost..        costcu       =E=  sum(I,NEW_CCU*qcu(I));
cucost_r..      costcu_r     =E=  sum(I,NEW_CCU*(EX_qcui(I)+add_qcu(I)));
fixhxcost..     fixcosthx    =E=  NEW_CFHX*counthx_r;

areaadd..       addarea      =E=  sum((I,J,K,BH,BC,SK),add_a(I,J,K,BH,BC,SK));
areaaddcost..   costareaadd  =E=  sum((I,J,K,BH,BC,SK),NEW_ACHX*(add_a(
I,J,K,BH,BC,SK)+0.0001)**(BETA(I,J)));
areahucost..    costareahu   =E=  sum(J,NEW_ACHU*(add_ahu(J)+0.0001)**(B
ETAHU(J)));
areacucost..    costareacu   =E=  sum(I,NEW_ACCU*(add_acu(I)+0.0001)**(BE
TACU(I)));
totalareacost.. areacost     =E=  costareaadd+costareahu+costareacu;
capcostperyr..  capitalcost  =E=  (fixcosthx+areacost);

*obj1 :  H+C utility cost
*obj2 :  Fixed cost of "NEW" HX
*obj3 :  Additional area cost
*obji : Total annual cost

obj1..          cost1   =E=  costhu_r+costcu_r;
obj2..          cost2   =E=  fixcosthx;
```

```
obj3..          cost3   =E=  areacost;

OBJFN1..        OBJcost1   =E=  cost1;
OBJFN2..        OBJcost2   =E=  cost1+cost2;
OBJFN3..        OBJcost3   =E=  cost1+cost3;
obji..          costi      =E=  cost1+cost2+cost3;

************************** END RETROFIT
2************************************
OPTION sysout = on;
OPTION IterlIm = 1e+09;
OPTION reslIm = 5e+06;


MODEL NONISO1 "NONISO1 MINLP"
/
        hoteqi,coldeqi,hotk,coldk,qhnoniso,qcnoniso,qcooler,qheater,thinassign,tcina
ssign,
monoth,monothout,monotc,monotcout,thoutst,tcoutst,logif,logig,sumf,sumg,logiqi,lo-
giqcu,logiqhu,
dthimin,dtcimin,dtcuimin,dthuimin,logidthi,logidtci,logidtcu,logidthu,LMTDi,LMT
DCUi,LMTDHUi,
AREAi,AREACUi,AREAHUi,
*****RETROFIT*****
add_hui,add_cui,add_areai_hi,add_areai_lo,
add_areacui_hi,add_areacui_lo,add_areahui_hi,add_areahui_lo,
add_hxi2,add_hxi1,add_cui2,add_cui1,add_hui2,add_hui1,
numberHXi,numberHX_ri,hxcost_ri,hucosti,hucost_ri,cucosti,cucost_ri,fixhxcosti,
areaaddi,areaaddcosti,areahucosti,areacucosti,totalareacosti,capcostperyri
obji1,obji2,obji3,objii /

MODEL FLOW "MIN TAC1 = UC _____NLP"
```

```
/
        QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,S
UMSK_QH,
QCH_K,QCK_K,QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,QH_SK,HO
T_SK,FHPT_SK,
QC_SK,COLD_SK,FCPT_SK,ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPA
SSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,
CON_FLOWH1,CON_FLOWH2,SUMFHP,CON_FLOWC1,CON_FLOWC2,SUM
FCP,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEA
S_KFIRST,
QCULOAD,QHULOAD,LOGQ_P,LOGQCU_P,LOGQHU_P,MASSK_H1,MASS
K_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,
DTHMAX_P,DTCMAX_P,DTCUMAX_P,DTHUMAX_P,DTHMIN_P,DTCMIN_
P,DTCUMIN_P,DTHUMIN_P
***** RETROFIT ******
add_hu,add_cu,add_area_hi_P,add_area_lo_P,
add_areacu_hi_P,add_areacu_lo_P,add_areahu_hi_P,add_areahu_lo_P,
add_hx2_P,add_hx1_P,add_cu2_P,add_cu1_P,add_hu2_P,add_hu1_P,
hucost,hucost_r,cucost,cucost_r,obj1,OBJFN1        /

MODEL STRUCTURE "MIN TAC2 = UC+NoHX _____MILP"
/
        QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QCH_K,QCK_K,QHK_SK
,SUMJ_BC_Q,SUMSK_QH,
QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,
QH_SK,HOT_SK_P,FHPT_SK_P,QC_SK,COLD_SK_P,FCPT_SK_P,
ASSIGNTH_K,ASSIGNTH_SK_P,ASSIGNTH_BYPASSH_P,HMIX_P,ASSIGNT
H_TSK,
ASSIGNTC_K,ASSIGNTC_SK_P,ASSIGNTH_BYPASSC_P,CMIX_P,ASSIGNT
```

C_TSK,
CON_FLOWH1_F_P,CON_FLOWH2_F_P,CON_FLOWC1_F_P,CON_FLOWC2_F_P,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEAS_KFIRST,
QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU3,CON_CU4,CON_HU1,CON_HU2,CON_HU3,CON_HU4,CON_EX,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,DTHUMIN,
***** RETROFIT *******
add_hu,add_cu,add_hx2,add_hx1,add_cu2,add_cu1,add_hu2,add_hu1,
numberHX,numberHX_r,hxcost_r,hucost,hucost_r,cucost,cucost_r,obj1,obj2,OBJFN2 /

MODEL FLOWarea "MIN TAC3 = UC+AC _____NLP"
/
QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,SUMSK_QH,
QCH_K,QCK_K,QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,
QH_SK,HOT_SK,FHPT_SK,QC_SK,COLD_SK,FCPT_SK,
ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPASSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,
CON_FLOWH1,CON_FLOWH2,SUMFHP,CON_FLOWC1,CON_FLOWC2,SUMFCP,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEAS_KFIRST,
QCULOAD,QHULOAD,LOGQ_P,LOGQCU_P,LOGQHU_P,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,
DTHMAX_P,DTCMAX_P,DTCUMAX_P,DTHUMAX_P,DTHMIN_P,DTCMIN_P,DTCUMIN_P,DTHUMIN_P,

```
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU,
***** RETROFIT *****
add_hu,add_cu,add_area_hi_P,add_area_lo_P,
add_areacu_hi_P,add_areacu_lo_P,add_areahu_hi_P,add_areahu_lo_P,
add_hx2_P,add_hx1_P,add_cu2_P,add_cu1_P,add_hu2_P,add_hu1_P,
hucost,hucost_r,cucost,cucost_r,areaadd,areaaddcost,areahucost,areacucost,totalareac
ost,
obj1,obj2,obj3,OBJFN3                /

MODEL NONLINEAR  "FINAL"
/
        QSKLAST,HOTEQ,COLDEQ,QHC_K,QHK_K,QHK_SK,SUMJ_BC_Q,S
UMSK_QH,
QCH_K,QCK_K,QCK_SK,SUMI_BH_Q,SUMSK_QC,HOTCOLD_K,
QH_SK,HOT_SK,FHPT_SK,QC_SK,COLD_SK,FCPT_SK,
ASSIGNTH_K,ASSIGNTH_SK,ASSIGNTH_BYPASSH,HMIX,ASSIGNTH_TSK,
ASSIGNTC_K,ASSIGNTC_SK,ASSIGNTH_BYPASSC,CMIX,ASSIGNTC_TSK,
CON_FLOWH1,CON_FLOWH2,SUMFHP,CON_FLOWC1,CON_FLOWC2,SUM
FCP,
THFEAS_K,THPFEAS_SK,THFEAS_KLAST,TCFEAS_K,TCPFEAS_SK,TCFEA
S_KFIRST,
QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,MASSK_H1,MASSK_C1,
CON_CU1,CON_CU2,CON_CU4,CON_HU1,CON_HU2,CON_HU4,CON_EX,C
ONQ_UP,CONQ_LO,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D
THUMIN,
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU

***** RETROFIT ******
add_hu,add_cu,add_area_hi,add_area_lo,
add_areacu_hi,add_areacu_lo,add_areahu_hi,add_areahu_lo,
add_hx2,add_hx1,add_cu2,add_cu1,add_hu2,add_hu1,
```

```
numberHX,numberHX_r,hxcost_r,hucost,hucost_r,cucost,cucost_r,fixhxcost,
areaadd,areaaddcost,areahucost,areacucost,totalareacost,capcostperyr,
obj1,obj2,obj3,OBJFN1,OBJFN2,OBJFN3,obji       /
;


*****************************ASSIGN INITIAL
PT.*********************************
f.l(I,J,K)      =       FH(I);
g.l(I,J,K)      =       FC(J);
qi.l(I,J,K)$(ORD(K) NE CARD(K)) =       f.l(I,J,K)*(THIN(I) - TCIN(J));
qcui.l(I)       =       500;
qhui.l(J)       =       200;
dtcui.l(I)      =       THIN(I) - TCUOUT;
dthui.l(J)      =       THUOUT - TCIN(J);
dthi.l(I,J,K)   =       THIN(I) - TCIN(J);
dtci.l(I,J,K)   =       THIN(I) - TCIN(J);
***************************


PARAMETER
      AHX_INT(I,J,K,BH,BC,SK)       Area of HX I-J (Area-lInear)
      ACU_INT(I)                    Area of HX I-CU (Area-lInear)
      AHU_INT(J)                    Area of HX J-HU (Area-lInear)

      TH_INT(I,K)
      THP_INT(I,K,BH,SK)
      TC_INT(J,K)
      TCP_INT(J,K,BC,SK)

      Q_INT(I,J,K,BH,BC,SK)
      QCU_INT(I)
      QHU_INT(J)
```

```
      QHK_INT(K,I)
      QCK_INT(K,J)

      FHP_INT(K,I,BH)
      FCP_INT(J,K,BC)

      Z_INT(I,J,K,BH,BC,SK)
      ZCU_INT(I)
      ZHU_INT(J)

      FBHPT_INT(I,K)
      FBCPT_INT(J,K)

      DTH_INT(I,J,K,BH,BC,SK)
      DTC_INT(I,J,K,BH,BC,SK)
      DTCU_INT(I)
      DTHU_INT(J)
****************** RETROFIT PARAMETER FOR SOLVE STRATEGY
************************
      ADD_AHX_INT(I,J,K,BH,BC,SK)
      ADD_ACU_INT(I)
      ADD_AHU_INT(J)
      ADD_Q_INT(I,J,K,BH,BC,SK)
      ADD_QCU_INT(I)
      ADD_QHU_INT(J)

      ADD_Z_INT(I,J,K,BH,BC,SK)
      ADD_ZCU_INT(I)
      ADD_ZHU_INT(J)
;

*########################    SOLVE
```

```
STRATEGY    ##############################
****************************** SOLVE NONISO1
**********************************
NONISO1.optfile = 2;
option reslim = 5e9;
option iterlim = 2e9;
OPTION SYSOUT=ON;
************************
$onecho>dicopt.op2
STOP 0
MAXCYCLES 350
epsmip 50
$offecho
************************
SOLVE NONISO1 USING MINLP MINIMIZING costii;
DISPLAY
th.l,thpi.l,tc.l,tcpi.l,zi.l,new_zi.l,ai.l,add_ai.l,qi.l,zcui.l,acui.l,new_zcui.l,add_acui.l,qc
ui.l,
add_qcui.l,zhui.l,ahui.l,new_zhui.l,add_ahui.l,qhui.l,add_qhui.l,f.l,g.l,counthx_ri.l,co
sthx_ri.l,
costhui.l,costhu_ri.l,costcui.l,costcu_ri.l,fixcosthxi.l,addareai.l,costareaaddi.l,costarea
hui.l,
costareacui.l,areacosti.l,capitalcosti.l,costi1.l,costi2.l,costi3.l,costii.l;
**********************************END OF SOLVE
NONISO1**************************

AHX_INT(I,J,K,BH,BC,SK) = ai.L(I,J,K);
ACU_INT(I) = acui.L(I);
AHU_INT(J) = ahui.L(J);
ADD_AHX_INT(I,J,K,BH,BC,SK) = add_ai.L(I,J,K);
ADD_ACU_INT(I) = add_acui.L(I);
ADD_AHU_INT(J) = add_ahui.L(J);
```

```
a.L(I,J,K,BH,BC,SK)=AHX_INT(I,J,K,BH,BC,SK);
acu.L(I) = ACU_INT(I);
ahu.L(J) = AHU_INT(J);
add_a.L(I,J,K,BH,BC,SK) = ADD_AHX_INT(I,J,K,BH,BC,SK);
add_acu.L(I) = ADD_ACU_INT(I);
add_ahu.L(J) = ADD_AHU_INT(J);

DTH_INT(I,J,K,BH,BC,SK) = dthi.L(I,J,K);
DTC_INT(I,J,K,BH,BC,SK) = dtci.L(I,J,K);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);

dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcui.L(I);
QHU_INT(J) = qhui.L(J);
Q_INT(I,J,K,BH,BC,SK) = qi.L(I,J,K);
QHK_INT(K,I) = sum(J,qi.L(I,J,K));
QCK_INT(K,J) = sum(I,qi.L(I,J,K));
ADD_QCU_INT(I) = add_qcui.L(I);
ADD_QHU_INT(J) = add_qhui.L(J);

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);
qhK.L(K,I) = QHK_INT(K,I);
qcK.L(K,J) = QCK_INT(K,J);
add_qcu.L(I) = ADD_QCU_INT(I);
```

```
add_qhu.L(J) = ADD_QHU_INT(J);

Z_P(I,J,K,BH,BC,SK) = zi.L(I,J,K);
ZCU_P(I) = zcui.L(I);
ZHU_P(J) = zhui.L(J);
NEW_Z_P(I,J,K,BH,BC,SK) = new_zi.L(I,J,K);
NEW_ZCU_P(I) =  new_zcui.L(I);
NEW_ZHU_P(J) =  new_zhui.L(J);

**************************
FLOW.optfile = 3;
option reslim = 10000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
**************************
$onecho>dicopt.op3
STOP 0
MAXCYCLES 350
mipoptfile cplex.opt 1
epsmip 20
$offecho
**************************
SOLVE FLOW USING NLP MINIMIZING OBJcost1;
DISPLAY
th.l,thp.l,tc.l,tcp.l,fhp.l,fcp.l,a.l,acu.l,ahu.l,q.l,qcu.l,qhu.l,Z_P,ZCU_P,ZHU_P
NEW_Z_P,NEW_ZCU_P,NEW_ZHU_P,add_a.l,add_acu.l,add_ahu.l,add_qcu.l,add
_qhu.l,cost1.l;
**********************************END OF SOLVE FLOW
**************************

TC_INT(J,K) = tc.l(J,K);
TCP_INT(J,K,BC,SK) = tcp.l(J,K,BC,SK);
```

```
TH_INT(I,K)  = th.l(I,K);
THP_INT(I,K,BH,SK) = thp.l(I,K,BH,SK);

tc.l(J,K) = TC_INT(J,K);
tcp.l(J,K,BC,SK) = TCP_INT(J,K,BC,SK);
th.l(I,K) = TH_INT(I,K);
thp.l(I,K,BH,SK) = THP_INT(I,K,BH,SK);

DTH_INT(I,J,K,BH,BC,SK) = dth.L(I,J,K,BH,BC,SK);
DTC_INT(I,J,K,BH,BC,SK) = dtc.L(I,J,K,BH,BC,SK);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);

dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcu.L(I);
QHU_INT(J) = qhu.L(J);
Q_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK);
ADD_QCU_INT(I) = add_qcu.L(I);
ADD_QHU_INT(J) = add_qhu.L(J);

qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);
add_qcu.L(I) = ADD_QCU_INT(I);
add_qhu.L(J) = ADD_QHU_INT(J);

z.L(I,J,K,BH,BC,SK) = Z_P(I,J,K,BH,BC,SK);
zcu.L(I) = ZCU_P(I);
```

```
zhu.L(J) = ZHU_P(J);
new_z.L(I,J,K,BH,BC,SK) = NEW_Z_P(I,J,K,BH,BC,SK);
new_zcu.L(I) = NEW_ZCU_P(I);
new_zhu.L(J) = nEW_ZHU_P(J);

*************************
FLOWarea.optfile = 4;
option reslim = 10000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
*************************
$onecho>dicopt.op4
STOP 0
MAXCYCLES 500
mipoptfile cplex.opt 1
epsmip 20
$offecho
*************************
SOLVE FLOWarea USING NLP MINIMIZING OBJcost3;
DISPLAY th.l,thp.l,tc.l,tcp.l,a.l,acu.l,ahu.l,q.l,qcu.l,qhu.l,Z_P,ZCU_P,ZHU_P
      NEW_Z_P,NEW_ZCU_P,NEW_ZHU_P,add_a.l,add_acu.l,add_ahu.l
      add_qcu.l,add_qhu.l,cost1.l,cost2.l,cost3.l;
*********************************END OF SOLVE FLOWarea
*************************

FHP_INT(K,I,BH) = fhp.l(I,K,BH);
FCP_INT(J,K,BC) = fcp.l(J,K,BC);
FBHPT_INT(I,K)  = fbhpt.l(I,K);
FBCPT_INT(J,K+1) = fbcpt.l(J,K+1);

fhp.l(I,K,BH) = FHP_INT(K,I,BH);
fcp.l(J,K,BC) = FCP_INT(J,K,BC);
```

```
fbhpt.l(I,K)  = FBHPT_INT(I,K);
fbcpt.l(J,K+1)= FBCPT_INT(J,K+1);

AHX_INT(I,J,K,BH,BC,SK) = a.l(I,J,K,BH,BC,SK);
ACU_INT(I) = acu.l(I);
AHU_INT(J) = ahu.l(J);
ADD_AHX_INT(I,J,K,BH,BC,SK) = add_a.l(I,J,K,BH,BC,SK);
ADD_ACU_INT(I) = add_acu.l(I);
ADD_AHU_INT(J) = add_ahu.l(J);

a.l(I,J,K,BH,BC,SK) = AHX_INT(I,J,K,BH,BC,SK);
acu.l(I) = ACU_INT(I);
ahu.l(J) = AHU_INT(J);
add_a.L(I,J,K,BH,BC,SK) = ADD_AHX_INT(I,J,K,BH,BC,SK);
add_acu.L(I) = ADD_ACU_INT(I);
add_ahu.L(J) = ADD_AHU_INT(J);

TC_INT(J,K) = tc.l(J,K);
TCP_INT(J,K,BC,SK) = tcp.l(J,K,BC,SK);
TH_INT(I,K)  = th.l(I,K);
THP_INT(I,K,BH,SK) = thp.l(I,K,BH,SK);

tc.l(J,K) = TC_INT(J,K);
tcp.l(J,K,BC,SK) = TCP_INT(J,K,BC,SK);
th.l(I,K) = TH_INT(I,K);
thp.l(I,K,BH,SK) = THP_INT(I,K,BH,SK);

DTH_INT(I,J,K,BH,BC,SK) = dth.L(I,J,K,BH,BC,SK);
DTC_INT(I,J,K,BH,BC,SK) = dtc.L(I,J,K,BH,BC,SK);
DTCU_INT(I) = dtcui.L(I);
DTHU_INT(J) = dthui.L(J);
```

```
dth.L(I,J,K,BH,BC,SK) = DTH_INT(I,J,K,BH,BC,SK);
dtc.L(I,J,K,BH,BC,SK) = DTC_INT(I,J,K,BH,BC,SK);
dtcu.L(I) = DTCU_INT(I);
dthu.L(J) = DTHU_INT(J);

QCU_INT(I) = qcu.L(I);
QHU_INT(J) = qhu.L(J);
Q_INT(I,J,K,BH,BC,SK) = q.L(I,J,K,BH,BC,SK);
ADD_QCU_INT(I) = add_qcu.L(I);
ADD_QHU_INT(J) = add_qhu.L(J);


qcu.L(I) =  QCU_INT(I);
qhu.L(J) = QHU_INT(J);
q.L(I,J,K,BH,BC,SK) = Q_INT(I,J,K,BH,BC,SK);
add_qcu.L(I) = ADD_QCU_INT(I);
add_qhu.L(J) = ADD_QHU_INT(J);

z.L(I,J,K,BH,BC,SK) = Z_P(I,J,K,BH,BC,SK);
zcu.L(I) = ZCU_P(I);
zhu.L(J) = ZHU_P(J);
new_z.L(I,J,K,BH,BC,SK) = NEW_Z_P(I,J,K,BH,BC,SK);
new_zcu.L(I) = NEW_ZCU_P(I);
new_zhu.L(J) = NEW_ZHU_P(J);
*************************
NONLINEAR.optfile = 5;
option reslim = 100000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
*************************
$onecho>dicopt.op5
STOP 0
```

```
MAXCYCLES 700
mipoptfile cplex.opt 1
epsmip 500
$offecho
************************
SOLVE NONLINEAR USING MINLP MINIMIZING costi;
DISPLAY
th.l,thp.l,tc.l,tcp.l,fhp.l,fcp.l,a.l,acu.l,ahu.l,q.l,qcu.l,qhu.l,z.l,zcu.L,zhu.L,new_z.l,new
_zcu.L,
new_zhu.L,add_a.l,add_acu.l,add_ahu.l,add_qcu.l,add_qhu.l,cost1.l,cost2.l,cost3.l,co
sti.l;
```

Appendix E: Retrofit for CDU-case study

```
SETS
    I     Hot stream /I1*I10/
    J     Cold stream /J1*J3/
    CU    Cold utility /CU1,CU2,CU3/
    HU    Hot utility /HU1,HU2/
    K     Major stage /KFIRST,K2,KLAST/
    SK    Sub-stage /SKFIRST,SK2*SK4,SKLAST/
    BC    Branch of hot splitting stream /BC1*BC2/

;

SCALARS
    EMAT          Exchange minimum approach temperature /5/
    TMIN          Minimum temperature in HEN /30/
    TMAX          Maximum temperature in HEN /347.3/
    AREA_MAX      Miximum area for retrofit HEN /4000/
    NEW_ACHX      Area cost coefficient of "NEW" process heat exchanger
/389/
    NEW_ACCU      Area cost coefficient of "NEW" cold utility /389/
    NEW_ACHU      Area cost coefficient of "NEW" hot utility /389/

    NEW_CFAREA    Fixed charges of "Additional-reductional" area /13230/
    NEW_redAREA   Area cost coefficient of "Additional-reductional" area /0.5/
    NEW_CFHX      Fixed charges of "NEW" exchanger /26462/
    NEW_CFCU      Fixed charges of "NEW" cold utility /26462/
    NEW_CFHU      Fixed charges of "NEW" hot utility /26462/
    SPLITcost     Fixed cost for splitting streams /20000/

;
```

PARAMETERS

*TEMPERATURE OF STREAM.....

THIN(I) Inlet temperature of hot stream

/ I1 319.4

I2 73.24

I3 347.3

I4 263.5

I5 297.4

I6 248

I7 73.24

I8 231.8

I9 167.1

I10 146.7 /

THOUT(I) Outlet temperature of hot stream

/I1 224.1

I2 30

I3 45

I4 180.2

I5 110

I6 50

I7 40

I8 120

I9 69.55

I10 73.24 /

TCIN(J) Inlet temperature of cold stream

/J1 30

J2 232.2

J3 226.2 /

TCOUT(J) Outlet temperature of cold stream

/J1 232.2

J2 343.3

J3 231.8 /

TCUIN(CU) Inlet temperature of cold utility

/CU1 20

CU2 124

CU3 174 /

TCUOUT(CU) Outlet temperature of cold utility

/CU1 25

CU2 125

CU3 175 /

THUIN(HU) Inlet temperature of hot utility

/HU1 250

HU2 1000 /

THUOUT Outlet temperature of hot utility

/HU1 249

HU2 500 /

*HEAT CAPACITY FLOWRATE OF PROCESS STREAM.....

FH(I) Heat capacity flowrate of hot stream

/I1 136.186

I2 6.842

I3 197.495

I4 123.06

I5 20.722

I6 63.166

I7 57.687

I8 48.526

I9 165.278

I10 253.551

/

FC(J) Heat capacity flowrate of cold stream

/J1 373.238

J2 488.127

J3 392.55 /

NEW_CCU(CU) Per unit cost of cold utility ($*kJ-1)per yr

/CU1 6.713

CU2 23.4

CU3 45.9/

NEW_CHU(HU) Per unit cost of hot utility ($*kJ-1)per yr

/HU1 71.09

HU2 134/

*---

*BRANCH FLOW __ [PARAMETER].....

FCP_P(J,K,BC) Branch flow parameter of cold stream

FBCPT_P(J,K)

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....

QUP_P(I,J,K,BC,SK) Upper bound of heat exchange

QLO_P(I,J,K,BC,SK) Lower bound of heat exchange

*---

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....

OMEGA(I,J) Upper bound for heat exchange

HCT_P(I) Heat content of hot stream

CCT_P(J) Heat content of cold stream

GAMMA(I,J) Upper bound for temperature dIfference

GAMMAH(I) Upper bound for temperature dIfference of hot stream

GAMMAC(J) Upper bound for temperature dIfference of cold stream

BETA(I,J) exponent for area costs of HX I-J

BETACU(I) exponent for area costs of cooler

BETAHU(J) exponent far area costs of heater

*OVERALL HEAT TRANSFER COEFFICIENT

Hh(I)

/ I1 1.293

I2 5.063

I3 0.756888191

I4 0.633

I5 1.199533618

I6 1.202472727

I7 1.099

I8 1.371456172

I9 1.373207073

I10 1.172944732 /

Hc(J)

/ J1 0.597369683

J2 0.788

J3 3.190160714 /

HCU(CU)

/ CU1 3.75

CU2 6

CU3 6 /

HHU(HU)

/ HU1 6

HU2 0.111 /

U(I,J) overall heat transfer coeff. of heat exchanger of I-J [KW*(m2*K)-1]

UCU(I,CU) overall heat transfer coeff. of cooler

UHU(J,HU) overall heat transfer coeff. of heater

*U(I,J) = [H(I)*H(J)]/[H(I)+H(J)]

```
      DTCUP(I,CU)
      DTHUP(J,HU)

*****RETROFIT*****
      EX_Aij(I,J)
      EX_ACU(I,CU)
      EX_AHU(J,HU)
      EX_Zij(I,J)
      EX_ZCU(I,CU)
      EX_ZHU(J,HU)

      EX_QCU(I,CU)
      EX_QHU(J,HU)

;


      DTHUP(J,HU) = max(0,THUIN(HU)-TCOUT(J));
      DTCUP(I,CU) = max(0,THOUT(I)-TCUIN(CU));

      BETA(I,J) = 1;
      BETACU(I) = 1;
      BETAHU(J) = 1;
      U(I,J)    = 1/(1/Hh(I)+1/Hc(J));
      UCU(I,CU)    = 1/(1/Hh(I)+1/HCU(CU));
      UHU(J,HU)    = 1/(1/Hc(J)+1/HHU(HU));

*****RETROFIT*****
      EX_Aij(I,J)          =     0;
      EX_Zij('I3','J1')    =     1;
      EX_Zij('I5','J1')    =     1;
      EX_Zij('I6','J1')    =     1;
      EX_Zij('I5','J2')    =     1;
```

```
EX_Zij('I3','J2')      =     1;
EX_Zij('I5','J1')      =     1;

EX_ZCU(I,CU)           =     0;
EX_ZCU('I7','CU1')     =     1;
EX_ZCU('I8','CU1')     =     1;
EX_ZCU('I5','CU1')     =     1;
EX_ZCU('I2','CU1')     =     1;
EX_ZCU('I6','CU1')     =     1;
EX_ZCU('I9','CU1')     =     1;
EX_ZCU('I4','CU2')     =     1;
EX_ZCU('I1','CU3')     =     1;
EX_ZCU('I10','CU1')    =     1;

EX_Aij('I3','J1')      =     3280;
EX_Aij('I5','J1')      =     27.4;
EX_Aij('I6','J1')      =     21.2;
EX_Aij('I5','J2')      =     67.6;
EX_Aij('I3','J2')      =     688;
EX_Aij('I5','J1')      =     36;

EX_ACU('I7','CU1')     =     62.6;
EX_ACU('I8','CU1')     =     33.6;
EX_ACU('I5','CU1')     =     4.08;
EX_ACU('I2','CU1')     =     5.63;
EX_ACU('I6','CU1')     =     153;
EX_ACU('I9','CU1')     =     182.57;
EX_ACU('I4','CU2')     =     101.27;
EX_ACU('I1','CU3')     =     93.8;
EX_ACU('I10','CU1')    =     250.9;

EX_AHU('J1','HU1')     =     1071;
```

```
EX_AHU('J3','HU1')    =    51.7;
EX_AHU('J2','HU2')    =    942;

EX_ZHU(J,HU)          =    0;
EX_ZHU('J1','HU1')    =    1;
EX_ZHU('J3','HU1')    =    1;
EX_ZHU('J2','HU2')    =    1;

EX_QCU(I,CU)          =    0;
EX_QCU('I7','CU1')    =    1917.516;
EX_QCU('I8','CU1')    =    4770.517;
EX_QCU('I5','CU1')    =    331.068;
EX_QCU('I2','CU1')    =    295.574;
EX_QCU('I6','CU1')    =    12506.86;
EX_QCU('I9','CU1')    =    16122.869;
EX_QCU('I4','CU2')    =    10250.898;
EX_QCU('I1','CU3')    =    10254.8;
EX_QCU('I10','CU1')   =    18625.856;

EX_QHU(J,HU)          =    0;
EX_QHU('J1','HU1')    =    28207;
EX_QHU('J3','HU1')    =    2198.28;
EX_QHU('J2','HU2')    =    37582.97;
```

VARIABLES

*TEMPERATURE OF PROCESS STREAM.....

th(I,K) Temperature of hot stream at stage K

thp(I,K,SK) Temperature of hot stream at sub-stage SK In stage K

tc(J,K) Temperature of cold stream at stage K

tcp(J,K,BC,SK) Temperature of cold stream at sub-stage SK In stage K

*HEAT CAPACITY FLOWRATE OF PROCESS

STREAM.....

fcp(J,K,BC) Branch flow parameter of cold stream

*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE (T*FCP).....

fcpt(J,K,BC,SK) MultIple of temperature and heat flow of cold stream

fbcpt(J,K)

*HEAT EXCHANGE.....

qhK(K,I) Heat exchange of hot (I-J) at stage K

qcK(K,J) Heat exchange of cold (I-J) at stage K

counthx

OBJcost1

costi

;

BINARY VARIABLES

*EXISTENCE OF EXCHANGER.....

z(I,J,K,BC,SK) ExIstence of exchanger I-J In each sK

zcu(I,CU) ExIstence of cold utIlIty

zhu(J,HU) ExIstence of hot utIlIty

new_z(I,J,K,BC,SK)

new_zcu(I,CU) ExIstence of NEW cold utIlIty

new_zhu(J,HU) ExIstence of NEW hot utIlIty

;

POSITIVE VARIABLES

*TEMPERATURE APPROACH.....

dth(I,J,K,BC,SK) Temperature dIfference at "hot end" of exchanger

dtc(I,J,K,BC,SK) Temperature dIfference at "cold end" of exchanger

dtcu(I,CU) Temperature dIfference of cold utIlIty

dthu(J,HU) Temperature dIfference of hot utIlIty

*AREA.....

*LOG MEAN TEMPERATURE DIFFERENCE.....

LMTDHX(I,J,K,BC,SK) Log mean emperature dIfference of exchanger I-J

LMTCU(I,CU) Log mean emperature dIfference of cold utIIIty

LMTHU(J,HU) Log mean emperature dIfference of hot utIIIty

*AREA.....

EX_AREA(I,J,K,BC,SK)

EX_AREAcu(I,CU)

EX_AREAhu(J,HU)

a(I,J,K,BC,SK) Heat exchange area of process exchanger

acu(I,CU) Heat exchange area of cold utIIIty

ahu(J,HU) Heat exchange area of hot utIIIty

*ADDITIONAL AREA

add_a(I,J,K,BC,SK)

add_acu(I,CU)

add_ahu(J,HU)

*HEAT EXCHANGE.....

q(I,J,K,BC,SK) Heat exchange between process stream I-J

qcu(I,CU) Heat exchange of cold utIIIty

qhu(J,HU) Heat exchange of hot utIIIty

UC Utility cost

NEX Number of exchanger

AC Area cost

BRANCH

TAC1

TAC2

TAC3

TAC4

TAC5

```
        COST

*RETROFIT _______ "COST".....

        add_qcu(I,CU)
        add_qhu(J,HU)
        costhu
        costcu
        fixcosthx
        addarea
        costareaadd
        costareahu
        costareacu
        areacost
        capitalcost

        cost1
        cost2
        cost3
        OBJcost2
        OBJcost3
        totalsavings
;

SCALARS HI,CJ;
HI=1;
CJ=1;
*LOOP OF HOT PROCESS STREAM.....
FOR(HI=1 to CARD(I),
        HCT_P(I)$[ORD(I) = HI] = FH(I)*(THIN(I)-THOUT(I));
        GAMMAH(I)$[ORD(I) = HI] = THIN(I)-THOUT(I);
*TEMPERATURE
```

```
        th.lo(I,K)$[ORD(I) = HI] = THOUT(I);
        th.up(I,K)$[ORD(I) = HI] = THIN(I);
        thp.lo(I,K,SK)$[ORD(I) = HI] = TMIN;
        thp.up(I,K,SK)$[ORD(I) = HI] = TMAX;
*HEAT EXCHANGE
        qhK.lo(K,I)$[ORD(I) = HI] = 0;
        qhK.up(K,I)$[ORD(I) = HI] = HCT_P(I);
        qcu.lo(I,CU)$[ORD(I) = HI] = 0;
        qcu.up(I,CU)$[ORD(I) = HI] = HCT_P(I);
);
*LOOP OF COLD PROCESS STREAM.....

For(CJ=1 to CARD(J),
        CCT_P(J)$[ORD(J) = CJ] = FC(J)*(TCOUT(J)-TCIN(J));
        GAMMAC(J)$[ORD(J) = CJ] = TCOUT(J)-TCIN(J);
*TEMPERATURE
        tc.lo(J,K)$[ORD(J) = CJ] = TCIN(J);
        tc.up(J,K)$[ORD(J) = CJ] = TCOUT(J);
        tcp.lo(J,K,BC,SK)$[ORD(J) = CJ] = TMIN;
        tcp.up(J,K,BC,SK)$[ORD(J) = CJ] = TMAX;
*HEAT CAPACITY FLOWRATE
        fcp.lo(J,K,BC)$[ORD(J) = CJ] = 0;
        fcp.up(J,K,BC)$[ORD(J) = CJ] = FC(J);
*MULTIPLE OF TEMPERATURE AND HEAT CAPACITY FLOWRATE
(T*FCP)
        fcpt.lo(J,K,BC,SK)$[ORD(J) = CJ] = 0;
        fcpt.up(J,K,BC,SK)$[ORD(J) = CJ] = FC(J)*TMAX;
        fbcpt.lo(J,K)$[ORD(J) = CJ] = 0;
        fbcpt.up(J,K)$[ORD(J) = CJ] = FC(J)*TCOUT(J);
*HEAT EXCHANGE
        qcK.lo(K,J)$[ORD(J) = CJ] = 0;
        qcK.up(K,J)$[ORD(J) = CJ] = CCT_P(J);
```

```
        qhu.lo(J,HU)$[ORD(J) = CJ] = 0;
        qhu.up(J,HU)$[ORD(J) = CJ] = CCT_P(J);
)
;

*_____
_____

*..... BOUNDS .....

*_____
_____

*BOUND OF HEAT EXCHANGE __ [PARAMETER].....
        QUP_P(I,J,K,BC,SK) = MIN(HCT_P(I),CCT_P(J),MAX(0,(THIN(I)-TCIN(J)-
EMAT)*MIN(FH(I),FC(J))));
        QLO_P(I,J,K,BC,SK) = 1*U(I,J)*EMAT;

*FOR LOGICAL CONSTRAINTS __ [PARAMETER].....
        OMEGA(I,J) = MIN(HCT_P(I),CCT_P(J));
        GAMMA(I,J) = MAX[0,(THIN(I)-TCIN(J)),(THIN(I)-
TCOUT(J)),(THOUT(I)-TCIN(J)),
                (THOUT(I)-TCOUT(J)),(TCIN(J)-THIN(I)),(TCIN(J)-THOUT(I)),
                (TCOUT(J)-THIN(I)),(TCOUT(J)-THOUT(I))];
        add_a.up(I,J,K,BC,SK) = 0.1*EX_Aij(I,J);
        add_a.up('I5','J1',K,BC,SK) = 0.2*EX_Aij('I5','J1');
        add_acu.up(I,CU) = 0.1*EX_ACU(I,CU);
        add_ahu.up(J,HU) = 0.1*EX_AHU(J,HU);
        a.up(I,J,K,BC,SK) = 4000;
        acu.up(I,CU) = 4000;
        ahu.up(J,HU) = 4000;
```

a.l('I3','J1',K,BC,SK) = 3280;
a.l('I5','J1',K,BC,SK) = 27.4;
a.l('I5','J1',K,BC,SK) = 21.2;
a.l('I6','J1',K,BC,SK) = 23.32;
a.l('I3','J2',K,BC,SK) = 688 ;
a.l('I5','J2',K,BC,SK) = 67.6;
acu.l('I1','CU3') = 93.8;
acu.l('I2','CU1') = 5.63;
acu.l('I4','CU2') = 101.27;
acu.l('I5','CU1') = 4.08;
acu.l('I6','CU1') = 153;
acu.l('I7','CU1') = 62.6;
acu.l('I8','CU1') = 33.6;
acu.l('I9','CU1') = 182.57;
acu.l('I10','CU1') = 250.9;
ahu.l('J1','HU1') = 1071;
ahu.l('J2','HU2') = 942;
ahu.l('J3','HU1') = 51.7;

*_____
_____

*..... BQUATION .....

*_____
_____

EQUATIONS
*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BC)

```
* 1. OVERALL HEAT BALANCE FOR EACH
STREAM****************************************
HOTEQ(I)
COLDEQ(J)

* 2. HEAT BALANCE AT EACH STAGE
K*************************************************
HOTCOLD_K(K)
QHK_K(K,I)
QCK_K(K,J)

* 3. HEAT BALANCE AT EACH SUB-STAGE
SK*********************************************
*[VARIABLE]
QHK_SK(K,I)
HOT_SK(I,K,SK)
QCK_SK(K,J)
COLD_SK(J,K,BC,SK)
*[FLOW PARAMETER]
COLD_SK_P(J,K,BC,SK)

* 4. COLD/HOT
UTILITY***************************************************************
***
QCULOAD(I)
QHULOAD(J)

* 5. TEMPERATURE
ASSIGNMENT************************************************************
ASSIGNTH_K(I)
ASSIGNTH_TSK(I,K)
ASSIGNTC_K(J)
```

CMIX(J,K)
ASSIGNTC_TSK(J,K,BC)
*[FLOW PARAMETER]
CMIX_P(J,K)

CON_FLOWC1(J,K)
CON_FLOWC2(J,K,BC)
CON_FLOWC1_F_P(J,K)
CON_FLOWC2_F_P(J,K,BC)

* 7. TEMPERATURE
FEASIBILITY**********
THFEAS_K(I,K)
THPFEAS_SK(I,K,SK)
THFEAS_KLAST(I)
TCFEAS_K(J,K)
TCPFEAS_SK(J,K,BC,SK)
TCFEAS_KFIRST(J)

* 9.LOGICAL CONSTRAINTS [HEAT EXCHANGE
,BRANCH+BYPASS]**********
**********HEAT EXCHANGE**********
*[Z-VARIABLE]
LOGq(I,J,K,BC,SK)
LOGQCU(I,CU)
LOGQHU(J,HU)

*10.OTHER
CONSTRAINTS**********
****
**********MAXIMUM MATCHING**********

*[Z- VARIABLE]

CONZ1(I,K)

CONZ2(J,K,BC,SK)

**********CONSTRAINT**********

CON_CU2

CON_HU2

* 11. CALCULATION OF APPROACH TEMPERATURE**********

*[Z- VARIABLE]

DTHMIN(I,J,K,BC,SK)

DTHMAX(I,J,K,BC,SK)

DTCMIN(I,J,K,BC,SK)

DTCMAX(I,J,K,BC,SK)

DTCUMIN(I,CU)

DTCUMAX(I,CU)

DTHUMIN(J,HU)

DTHUMAX(J,HU)

* 12. AREA EQUATION**********

******

LMTD(I,J,K,BC,SK)

LMTHc(I,J,K,BC,SK)

LMTDCU(I,CU)

LMTCUc(I,CU)

LMTDHU(J,HU)

LMTHUc(J,HU)

AREA(I,J,K.BC,SK)

AREACU(I,CU)

AREAHU(J,HU)

* 13. OBJECTIVE FUNCTION**********
**

OBJ1_NoHX

OBJ2_UTIL

OBJ3_AREA

OBJ4_BRANCH

OBJFN1

OBJFN2

OBJFN3

OBJFN4

OBJFN5

TOTALCOST

BOUNDTAC

********** RETROFIT **********

EX_AREAHX(I,J,K,BC,SK)

EX_AREAC(I,CU)

EX_AREAH(J,HU)

add_area_lo(I,J,K,BC,SK)

add_area_hi(I,J,K,BC,SK)

add_areacu_lo(I,CU)

add_areacu_hi(I,CU)

add_areahu_lo(J,HU)

add_areahu_hi(J,HU)

add_hu(I,CU)
add_cu(J,HU)

add_hx1(I,J,K,BC,SK)
add_hx2(I,J,K,BC,SK)
add_cu1(I,CU)
add_cu2(I,CU)
add_hu1(J,HU)
add_hu2(J,HU)

numberHX
hucost
cucost
fixhxcost
areaadd
areaaddcost
areahucost
areacucost
totalareacost
capcostperyr

obj1 objective function to be minimized
obj2
obj3

OBJFN_R1
OBJFN_R2
OBJFN_R3
obj_Ri

```
**********************************************************************
***********

;

*HEAT CONSTRAINT FOR LAST SUB-STAGE (ALL SPLITTING STREAM
ARE MERGED,NO HEAT TRANSFERRING)
QSKLAST(I,J,K,BC)..  q(I,J,K,BC,'SKLAST') =E=    0;

* 1. OVERALL HEAT BALANCE FOR EACH
STREAM**************************************
HOTEQ(I)..     FH(I)*[THIN(I)-
THOUT(I)]      =E=    SUM((K,J,BC,SK)$[(ORD(K) NE
CARD(K))AND(ORD(SK) NE CARD(SK))],q(I,J,K,BC,SK)) +
SUM(CU,qcu(I,CU));
COLDEQ(J)..    FC(J)*[TCOUT(J)-
TCIN(J)]       =E=    SUM((K,BC,I,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK)
NE CARD(SK))],q(I,J,K,BC,SK)) + SUM(HU,qhu(J,HU));

* 2. HEAT BALANCE AT EACH STAGE
K***************************************************
QHK_K(K,I)$[ORD(K) NE CARD(K)]..       FH(I)*[th(I,K) -
th(I,K+1)]    =E=    qhK(K,I);
QHK_SK(K,I)$[ORD(K) NE
CARD(K)]..            qhK(K,I)       =E=   SUM((J,BC,SK),q(I,J,K,BC,SK));
QCK_K(K,J)$[ORD(K) NE CARD(K)]..       FC(J)*[tc(J,K) -
tc(J,K+1)]    =E=    qcK(K,J);
QCK_SK(K,J)$[ORD(K) NE
CARD(K)]..            qcK(K,J)       =E=   SUM((I,BC,SK),q(I,J,K,BC,SK));

HOTCOLD_K(K)$[ORD(K) NE
CARD(K)]..     SUM(J,qcK(K,J))  =E=    SUM(I,qhK(K,I));
```

```
* 3. HEAT BALANCE AT EACH SUB-STAGE
SK*****************************************
*[VARIABLE]
HOT_SK(I,K,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM((J,BC),q(I,J,K,BC,SK)) =E= FH(I)*[thp(I,K,SK)-
thp(I,K,SK+1)];
COLD_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM(I,q(I,J,K,BC,SK)) =E= fcp(J,K,BC)*[tcp(J,K,BC,SK)-
tcp(J,K,BC,SK+1)];
*[FLOW PARAMETER]
COLD_SK_P(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM(I,q(I,J,K,BC,SK)) =E= FCP_P(J,K,BC)*[tcp(J,K,BC,S
K)-tcp(J,K,BC,SK+1)];

* 4. COLD/HOT
UTILITY*****************************************************
***
QCULOAD(I).. SUM(CU,qcu(I,CU)) =E= [th(I,'KLAST')-
THOUT(I)]*FH(I) ;
QHULOAD(J).. SUM(HU,qhu(J,HU)) =E= [TCOUT(J)-
tc(J,'KFIRST')]*FC(J) ;

* 5. TEMPERATURE
ASSIGNMENT**************************************************
ASSIGNTH_K(I).. THIN(I) =E= th(I,'KFIRST');
ASSIGNTH_TSK(I,K)$[ORD(K) NE CARD(K)]
.. TH(I,K) =E= thp(I,K,'SKFIRST');
ASSIGNTC_K(J).. TCIN(J) =E= tc(J,'KLAST');
CMIX(J,K)$[ORD(K) NE
CARD(K)].. FC(J)*tc(J,K) =E= SUM(BC,fcp(J,K,BC)*tcp(J,K,
BC,'SKFIRST'));
```

```
ASSIGNTC_TSK(J,K,BC)$[ORD(K) NE CARD(K)]
..          TC(J,K+1)       =E=    tcp(J,K,BC,'SKLAST');

*[FLOW PARAMETER]
CMIX_P(J,K)$[ORD(K) NE
CARD(K)]..                  FC(J)*tc(J,K+1)  =E=    SUM(BC,FCP_P(J,K,BC)*tcp(J
,K,BC,'SKLAST'));
**********FLOW CONSTRAINT***********
*[VARIABLES]
CON_FLOWC1(J,K)$[ORD(K) NE
CARD(K)]..          SUM(BC,fcp(J,K,BC))    =L=    FC(J);
CON_FLOWC2(J,K,BC)$[ORD(K) NE CARD(K)]..          fcp(J,K,BC)    =G=    0;
*[FLOW PARAMETER]
CON_FLOWC1_F_P(J,K)$[ORD(K) NE
CARD(K)]..      SUM(BC,FCP_P(J,K,BC))    =L=    FC(J);
CON_FLOWC2_F_P(J,K,BC)$[ORD(K) NE
CARD(K)]..      FCP_P(J,K,BC)    =G=    0;

* 6. TEMPERATURE
FEASIBILITY*************************************************
THFEAS_K(I,K)$[ORD(K) NE CARD(K)]..                th(I,K) =G=    th(I,K+1);
THPFEAS_SK(I,K,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..        thp(I,K,SK)    =G=    thp(I,K,SK+1);
THFEAS_KLAST(I)..                          th(I,'KLAST')    =G=    THOUT(I);
TCFEAS_K(J,K)$[ORD(K) NE CARD(K)]..                tc(J,K) =G=    tc(J,K+1);
TCPFEAS_SK(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..        tcp(J,K,BC,SK)    =G=    tcp(J,K,BC,SK+1);
TCFEAS_KFIRST(J)..                         tc(J,'KFIRST')    =L=    TCOUT(J);

* 8.LOGICAL CONSTRAINTS [HEAT EXCHANGE
,BRANCH+BYPASS]*****************************
***********HEAT EXCHANGE***********
```

```
*[ Z-VARIABLE]
LOGq(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. q(I,J,K,BC,SK) - OMEGA(I,J)*z(I,J,K,BC,SK) =L= 0;
LOGQCU(I,CU).. qcu(I,CU) - HCT_P(I)*zcu(I,CU) =L= 0;
LOGQHU(J,HU).. qhu(J,HU) - CCT_P(J)*zhu(J,HU) =L= 0;

*9.OTHER
CONSTRAINTS**********************************************************
********
***********MAXIMUM MATCHING***********
*[ Z- VARIABLE]
CONZ1(I,K)$[(ORD(K) NE
CARD(K))].. SUM((J,BC,SK),z(I,J,K,BC,SK)) =L= 1;
CONZ2(J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. SUM(I,z(I,J,K,BC,SK)) =L= 1;

* Heat Exchange and Heat & Cold utIlIty constraInts.....
* Cold utIlIty....
CON_CU2 .. SUM((I,CU),qcu(I,CU))=L= 75076.08;
* Hot utIlIty....
CON_HU2 .. SUM((J,HU),qhu(J,HU))=L= 48000;

* 10. CALCULATION OF APPROACH
TEMPERATURE*****************************************
*[ Z- VARIABLE]
DTHMAX(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dth(I,J,K,BC,SK) =L= thp(I,K,SK)-
tcp(J,K,BC,SK)+GAMMA(I,J)*(1-z(I,J,K,BC,SK));
DTCMAX(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. dtc(I,J,K,BC,SK) =L= thp(I,K,SK+1)-
tcp(J,K,BC,SK+1)+GAMMA(I,J)*(1-z(I,J,K,BC,SK));
DTCUMAX(I,CU).. dtcu(I,CU) =L= th(I,'KLAST')-
```

```
THOUT(I)+GAMMAH(I)*(1-zcu(I,CU));
DTHUMAX(J,HU)..        dthu(J,HU)      =L=    TCOUT(J)-
tc(J,'KFIRST')+GAMMAC(J)*(1-zhu(J,HU));

DTHMIN(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..    dth(I,J,K,BC,SK)  =G=    EMAT;
DTCMIN(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..    dtc(I,J,K,BC,SK)  =G=    EMAT;
DTCUMIN(I,CU)..         dtcu(I,CU)      =G=    EMAT;
DTHUMIN(J,HU)..         dthu(J,HU)      =G=    EMAT;

* 11. AREA
EQUATION**********************************************************
******
*LMTHc(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))] .. LMTDHX(I,J,K,BC,SK) =L=
0.5*(dth(I,J,K,BC,SK)+dtc(I,J,K,BC,SK));
LMTD(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE CARD(SK))]..
LMTDHX(I,J,K,BC,SK) =E=
                (2/3)*((dth(I,J,K,BC,SK)+0.001)**0.5)*((dtc(I,J,K,BC,SK)+0.0
01)**0.5)+(1/6)*dth(I,J,K,BC,SK)+(1/6)*dtc(I,J,K,BC,SK);
*LMTCUc(I,CU) .. LMTCU(I,CU) =L= 0.5*(dtcu(I,CU)+(th(I,'KLAST')-
THOUT(I)));
LMTDCU(I,CU).. LMTCU(I,CU) =E=
(2/3)*((dtcu(I,CU)+0.001)**0.5)*(max((DTCUP(I,CU)+0.001),0)**0.5)+(1/6)*dtcu(
I,CU)+(1/6)*DTCUP(I,CU);

LMTDHU(J,HU).. LMTHU(J,HU) =E=
(2/3)*((dthu(J,HU)+0.001)**0.5)*(max((DTHUP(J,HU)+0.001),0)**0.5)+(1/6)*dth
u(J,HU)+(1/6)*DTHUP(J,HU);
*LMTHUc(J,HU) .. LMTHU(J,HU) =L= 0.5*(dthu(J,HU)+(TCOUT(J)-
tc(J,'KFIRST')));
```

```
AREA(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..q(I,J,K,BC,SK)-
U(I,J)*LMTDHX(I,J,K,BC,SK)*(a(I,J,K,BC,SK)**(1/BETA(I,J))) =L= 0;
AREACU(I,CU)..      qcu(I,CU)-
(acu(I,CU)**(1/BETACU(I)))*UCU(I,CU)*LMTCU(I,CU) =l=     0;
AREAHU(J,HU)..      qhu(J,HU)-
(ahu(J,HU)**(1/BETAHU(J)))*UHU(J,HU)*LMTHU(J,HU) =l=     0;


* 13. OBJECTIVE
FUNCTION**********************************************************
**
OBJ1_NoHX..     NEX    =E=    sum((I,J,K,BC,SK),NEW_CFHX*z(I,J,K,BC,SK))
+sum((I,CU),NEW_CFCU*zcu(I,CU))+sum((J,HU),NEW_CFHU*zhu(J,HU));
OBJ2_UTIL..     UC     =E=    sum((I,CU),NEW_CCU(CU)*qcu(I,CU))+sum((J,H
U),NEW_CHU(HU)*qhu(J,HU));
OBJ3_AREA..     AC     =E=    sum((I,J,K,BC,SK),NEW_ACHX*a(I,J,K,BC,SK))
+sum((I,CU),NEW_ACCU*acu(I,CU))+sum((J,HU),NEW_ACHU*ahu(J,HU));

************************** RETROFIT
*****************************************
add_area_lo(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
=L=    AREA_MAX-
(a(I,J,K,BC,SK)+0.0001)**(1/BETA(I,J))+EX_AREA(I,J,K,BC,SK);
add_area_hi(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))].. add_a(I,J,K,BC,SK)-(0.0001**(1/0.6)+4.308869E-7)
=G=    (a(I,J,K,BC,SK)+0.0001)**(1/BETA(I,J))-EX_AREA(I,J,K,BC,SK);
add_areacu_lo(I,CU)..       add_acu(I,CU)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-
(acu(I,CU)+0.0001)**(1/BETACU(I))+EX_AREACU(I,CU);
```

```
add_areacu_hi(I,CU)..      add_acu(I,CU)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (acu(I,CU)+0.0001)**(1/BETACU(I))-EX_AREACU(I,CU);
add_areahu_lo(J,HU)..      add_ahu(J,HU)-(0.0001**(1/0.6)+4.308869E-
7)    =L=    AREA_MAX-
(ahu(J,HU)+0.0001)**(1/BETAHU(J))+EX_AREAHU(J,HU);
add_areahu_hi(J,HU)..      add_ahu(J,HU)-(0.0001**(1/0.6)+4.308869E-
7)    =G=    (ahu(J,HU)+0.0001)**(1/BETAHU(J))-EX_AREAHU(J,HU);

add_hu(I,CU)..    add_qcu(I,CU)    =G=    qcu(I,CU)-
EX_qcu(I,CU)*EX_Zcu(I,CU);
add_cu(J,HU)..    add_qhu(J,HU)    =G=    qhu(J,HU)-
EX_qhu(J,HU)*EX_Zhu(J,HU);

add_hx1(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  z(I,J,K,BC,SK)-EX_Zij(I,J)  =E=    new_z(I,J,K,BC,SK);
add_hx2(I,J,K,BC,SK)$[(ORD(K) NE CARD(K))AND(ORD(SK) NE
CARD(SK))]..  EX_Zij(I,J)+new_z(I,J,K,BC,SK)      =L=    1;
add_cu1(I,CU)..    zcu(I,CU)      =E=    EX_zcu(I,CU)+new_zcu(I,CU);
add_cu2(I,CU)..    EX_zcu(I,CU)+new_zcu(I,CU)      =L=    1;
add_hu1(J,HU)..    zhu(J,HU)      =E=    EX_zhu(J,HU)+new_zhu(J,HU);
add_hu2(J,HU)..    EX_zhu(J,HU)+new_zhu(J,HU)      =L=    1;

numberHX..    counthx      =E=    sum((I,J,K,BC,SK)$(ord(SK) ne
card(SK)),z(I,J,K,BC,SK))+sum((I,CU),zcu(I,CU))+sum((J,HU),zhu(J,HU));
hucost..      costhu       =E=    sum((J,HU),NEW_CHU(HU)*qhu(J,HU));
cucost..      costcu       =E=    sum((I,CU),NEW_CCU(CU)*qcu(I,CU));
fixhxcost..   fixcosthx    =E=    NEW_CFHX*counthx;

areaadd..     addarea      =E=    sum((I,J,K,BC,SK),add_a(I,J,K,BC,SK));
areaaddcost..  costareaadd  =E=    sum((I,J,K,BC,SK),NEW_ACHX*(add_a(I,J,
K,BC,SK)+0.0001)**(BETA(I,J)));
areahucost..  costareahu   =E=    sum((J,HU),NEW_ACHU*(add_ahu(J,HU)+0.0
```

```
001)**(BETAHU(J)));
areacucost..  costareacu   =E=   sum((I,CU),NEW_ACCU*(add_acu(I,CU)+0.00
01)**(BETACU(I)));
totalareacost..  areacost     =E=   costareaadd+costareahu+costareacu;
capcostperyr..  capitalcost   =E=   (fixcosthx+areacost);

*obj1 :  H+C utility cost
*obj2 :  Fixed cost of "NEW" HX
*obj3 :  Additional area cost
*obji : Total annual cost

obj1..        cost1   =E=  costhu+costcu;
obj2..        cost2   =E=  fixcosthx;
obj3..        cost3   =E=  areacost;

OBJFN_R1..       OBJcost1   =E=  (NEX+AC)/5+UC;
obj_Ri..         costi      =E=  totalsavings - capitalcost/5;

************************** END RETROFIT
2************************************
OPTION sysout = on;
OPTION IterlIm = 1e+09;
OPTION reslIm = 5e+06;


MODEL STRUCTURE "MIN TAC = NoHX _____MILP"
/
        HOTEQ,COLDEQ,QHK_K,QCK_K,QHK_SK,QCK_SK,HOTCOLD_K,HO
T_SK,COLD_SK_P,
ASSIGNTH_K,ASSIGNTH_TSK,ASSIGNTC_K,ASSIGNTC_TSK,
CON_FLOWC1_F_P,CON_FLOWC2_F_P,THFEAS_K,THPFEAS_SK,THFEAS_
KLAST,TCFEAS_K,
```

```
TCPFEAS_SK,TCFEAS_KFIRST,
QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,CONZ1,CONZ2,CON_CU2,
CON_HU2,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D
THUMIN,OBJ1_NoHX,
***** RETROFIT *****
hucost,cucost,numberHX      /

MODEL NONLINEAR  "TAC"
/
        HOTEQ,COLDEQ,QHK_K,QHK_SK,QCK_K,QCK_SK,HOTCOLD_K,HO
T_SK,COLD_SK,
ASSIGNTH_K,ASSIGNTH_TSK,ASSIGNTC_K,ASSIGNTC_TSK,
CON_FLOWC1,CON_FLOWC2,THFEAS_K,THPFEAS_SK,THFEAS_KLAST,T
CFEAS_K,TCPFEAS_SK,
TCFEAS_KFIRST,QCULOAD,QHULOAD,LOGq,LOGQCU,LOGQHU,CONZ1,C
ONZ2,CON_CU2,CON_HU2,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D
THUMIN,
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU,
***** RETROFIT *****
OBJ1_NoHX,OBJ2_UTIL,OBJ3_AREA,OBJFN_R1        /


MODEL NONLINEAR_R  "profit"
/
        HOTEQ,COLDEQ,QHK_K,QHK_SK,QCK_K,QCK_SK,HOTCOLD_K,HO
T_SK,COLD_SK,
ASSIGNTH_K,ASSIGNTH_TSK,ASSIGNTC_K,ASSIGNTC_TSK,
CON_FLOWC1,CON_FLOWC2,THFEAS_K,THPFEAS_SK,THFEAS_KLAST,T
CFEAS_K,TCPFEAS_SK,
TCFEAS_KFIRST,QCULOAD,QHULOAD,LOGQ,LOGQCU,LOGQHU,CONZ1,
```

```
CONZ2,CON_CU2,CON_HU2,
DTHMAX,DTCMAX,DTCUMAX,DTHUMAX,DTHMIN,DTCMIN,DTCUMIN,D
THUMIN
LMTD,LMTDCU,LMTDHU,AREA,AREACU,AREAHU
***** RETROFIT *****
numberHX,hucost,cucost,fixhxcost,
areaadd,areaaddcost,areahucost,areacucost,totalareacost,capcostperyr,obj1,obj2,obj3,
OBJFN_R1,obj_Ri    /
;


*****************************ASSIGN INITIAL
PT.*********************************
PARAMETER

      AHX_INT(I,J,K,BC,SK)       Area of HX I-J (Area-lInear)
      ACU_INT(I,CU)              Area of HX I-CU (Area-lInear)
      AHU_INT(J,HU)              Area of HX J-HU (Area-lInear)
      ADD_AHX_INT(I,J,K,BC,SK)
      ADD_ACU_INT(I,CU)
      ADD_AHU_INT(J,HU)
      NOHX_INT
      TAC_INT
      PROFIT_INT
      K_LOOP
;

****************************** SOLVE NONISO1
**********************************
FCP_P('J1','KFIRST','BC1')=0;
FCP_P('J2','KFIRST','BC1')=0;
FCP_P('J3','KFIRST','BC1')=0;
```

```
FCP_P('J1','K2','BC2')=0;
FCP_P('J2','K2','BC2')=0;
FCP_P('J3','K2','BC2')=0;

while ((FCP_P('J1','KFIRST','BC1') le 373.238),
while ((FCP_P('J2','KFIRST','BC1') le 488.127),
while ((FCP_P('J3','KFIRST','BC1') le 392.55),
 FCP_P('J1','KFIRST','BC2')=FC('J1')-FCP_P('J1','KFIRST','BC1');
 FCP_P('J2','KFIRST','BC2')=FC('J2')-FCP_P('J2','KFIRST','BC1');
 FCP_P('J3','KFIRST','BC2')=FC('J3')-FCP_P('J3','KFIRST','BC1');

OPTION q:5:2:3;
OPTION a:5:2:3;
OPTION tcp:3:1:2;
OPTION z:1:2:3;
OPTION fcp:3:1:2;

SOLVE STRUCTURE MINIMIZING counthx USING MIP;
display q.l,qcu.l,qhu.l,th.l,thp.l,tc.l,tcp.l,z.l,zcu.l,zhu.l,fcp.l;
ACU_INT(I,CU) =
qcu.l(I,CU)/((UCU(I,CU)*(2/3)*((dtcu.L(I,CU)+0.001)**0.5)*(max((DTCUP(I,CU)
+0.001),0)**0.5)+(1/6)*dtcu.L(I,CU)+(1/6)*DTCUP(I,CU))+1e-06);
AHU_INT(J,HU) =
qhu.l(J,HU)/((UHU(J,HU)*(2/3)*((dthu.L(J,HU)+0.001)**0.5)*(max((DTHUP(J,H
U)+0.001),0)**0.5)+(1/6)*dthu.L(J,HU)+(1/6)*DTHUP(J,HU))+1e-06);
AHX_INT(I,J,K,BC,SK) = q.L(I,J,K,BC,SK)/((U(I,J)*(
(2/3)*((max(dth.L(I,J,K,BC,SK),0)+0.001)**0.5)*((max(dtc.L(I,J,K,BC,SK),0)+0.0
01)**0.5)+(1/6)*dth.L(I,J,K,BC,SK)+(1/6)*dtc.L(I,J,K,BC,SK)))+1e-06);

ADD_ACU_INT(I,CU) = MAX(((ACU_INT(I,CU)+0.0001)**(1/BETACU(I))-
EX_ACU(I,CU)),0);
ADD_AHU_INT(J,HU) = MAX(((AHU_INT(J,HU)+0.0001)**(1/BETAHU(J))-
```

```
EX_AHU(J,HU)),0);
ADD_AHX_INT(I,J,K,BC,SK) =
MAX(((AHX_INT(I,J,K,BC,SK)+0.0001)**(1/BETA(I,J))-EX_Aij(I,J)),0);


NOHX_INT = counthx.l   ;
TAC_INT =
NEW_CFHX*NOHX_INT+sum((I,J,K,BC,SK),NEW_ACHX*(ADD_AHX_INT(I,
J,K,BC,SK))**(BETAHU(J)))+
      sum((J,HU),NEW_ACHU*(ADD_AHU_INT(J,HU)+0.0001)**(BETAHU(J)))
+sum((I,CU),NEW_ACCU*(ADD_ACU_INT(I,CU)+0.0001)**(BETACU(I)));

PROFIT_INT = 7197629.335-sum((J,HU),NEW_CHU(HU)*qhu.L(J,HU)) -
TAC_INT/5;
fcp.L(J,K,BC) = FCP_P(J,K,BC);
acu.L(I,CU) = ACU_INT(I,CU);
ahu.L(J,HU) = AHU_INT(J,HU);
a.L(I,J,K,BC,SK) = AHX_INT(I,J,K,BC,SK);
add_acu.L(I,CU) = ADD_ACU_INT(I,CU);
add_ahu.L(J,HU) = ADD_AHU_INT(J,HU);
add_a.L(I,J,K,BC,SK)  = ADD_AHX_INT(I,J,K,BC,SK);

**************************************************
NONLINEAR.optfile = 4;
option reslim = 100000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
************************
$onecho>dicopt.op4
STOP 0
MAXCYCLES 20
mipoptfile cplex.opt 1
```

```
epsmip 200
$offecho
************************************************
SOLVE NONLINEAR MINIMIZING OBJcost1 USING MINLP;
display q.l,qcu.l,qhu.l,th.l,thp.l,tc.l,tcp.l,z.l,zcu.l,zhu.l,a.l,acu.l,ahu.l,OBJcost1.l,fcp.l;


FCP_P(J,'KFIRST','BC1') = 0.05*FC(J);
totalsavings.l = 7197629.335+1076897.308 - costhu.L-costcu.L;

EX_AREACU.L(I,CU)    =    EX_ACU(I,CU)*zcu.L(I,CU);
EX_AREAHU.L(J,HU)    =    EX_AHU(J,HU)*zhu.L(J,HU);
EX_AREA.L(I,J,K,BC,SK)    =    EX_Aij(I,J)*z.L(I,J,K,BC,SK);

display EX_AREA.L,EX_AREACU.L,EX_AREAHU.L;
************************************************
NONLINEAR_R.optfile = 5;
option reslim = 100000;
option iterlim = 2e9;
OPTION SYSOUT=ON;
************************
$onecho>dicopt.op5
STOP 0
MAXCYCLES 20
mipoptfile cplex.opt 1
epsmip 300
$offecho
************************************************
SOLVE NONLINEAR_R MAXIMIZING costi USING MINLP;
display
q.l,qcu.l,qhu.l,th.l,thp.l,tc.l,tcp.l,z.l,zcu.l,zhu.l,a.l,acu.l,ahu.l,add_a.l,EX_AREA.L,E
X_AREACU.L,EX_AREAHU.L,add_acu.l,add_ahu.l,OBJcost1.l,fcp.l,totalsavings.l,
```

costi.l,counthx.L,fixcosthx.L,costhu.L,costcu.L,areacost.L ;

);););

ต้นฉบับ หน้าขาดหาย