A Scalable Discrete Optimization Algorithm for Heat Integration in Early Design ⋆

Eric S Fraga[1] and Ken I M McKinnon[2]

[1] Department of Chemical Engineering, University College London, London WC1E 7JE, England, United Kingdom (e.fraga@ucl.ac.uk)
[2] Department of Mathematics & Statistics, University of Edinburgh, Edinburgh EH9 3JZ, Scotland, United Kingdom (k.i.m.mckinnon@ed.ac.uk)

Abstract. The design of a chemical plant at the earliest stages of design is most successful when a range of issues are considered simultaneously. Amongst these is the minimization of utility consumption, both for heating and cooling process streams. Chemical plants are big consumers of utilities. Plants which are able to most efficiently use the energy available within the process are those which are most economical to run. This paper describes a discrete programming approach, incorporated into an automated process synthesis procedure, which is able to identify plant designs which are amenable to heat integration. An existing algorithm with exponential growth in time has been modified and extended. When used as an exploration tool in the early stages of chemical plant design, the new algorithm exhibits polynomial growth. This is achieved by targeting a different search space, one which is better suited for early design. The new algorithm incorporates the previous one and so all benefits, especially for later design use, are retained. Furthermore, the new approach minimizes the a priori decisions required of the user and the implementation makes efficient reuse of computation.

1 Introduction

The automated synthesis of process flowsheets is a useful tool in the design of a chemical plant. The usefulness of the tool is enhanced by the possibility of generating processes which are both realistic and complete. The generation of heat integrated processes is a step towards both these goals.

Heat integration is necessary on the basis of both economic and environmental criteria. This is particularly true for distillation sequences which are an essential part of most processes of interest to the chemical processing industry. However, heat integration may also be of interest in processes involving reactors and other special units. A process synthesis procedure which is able to cope with the heat integration requirements of a wide range of processes is a valuable tool.

This paper describes an algorithm for heat integration incorporated within the *Jacaranda* implicit enumeration process synthesis package [3].

⋆ Appeared in Scientific Computing in Chemical Engineering II: Simulation, Image Processing, Optimization and Control, F Keil, W Mackens, H Voss & J Werther, Editors, Springer, pages 306-313

2 Discrete Programming Synthesis Method

The Jacaranda process synthesis system used implicit enumeration, combined with discrete programming, to automatically and simultaneously generate and search the superstructure which encodes the solutions for a synthesis problem. The advantages of this method include the ability to work with nonconvex models, the generation of N-best solutions, the efficient implementation on parallel computers, and its ease of use [4,5,2].

The implicit enumeration procedure is motivated by the the cost function:

$$f(P) = \min_{\mathrm{u}} \left\{ c(u) + \sum_{i \in D(u)} f(P_{\mathrm{i}}) \right\} \tag{1}$$

where P is a particular problem, usually defined by a stream which needs to be processed, u ranges over all allowable units and the particular designs possible for that stream, $c(u)$ is the cost of a specific unit design, $D(u)$ are the indices of the subproblems corresponding to the output streams of unit u, and $f(\cdot)$ is the cost of the solution to the subproblem. This cost function leads automatically to a finite recursive search procedure, relying on the fact that the graph with directed edges from problem to subproblem is acyclic. Note that the same subproblem may be generated by different parent problems and this can be used to save work in the search.

Although describing an acyclic graph, (1) can be used to represent a wide range of process synthesis problems, including problems which require complex recycle structures for their solution. Application to specific classes of problems requires appropriate formulations for P and P_i in (1) [4,5,2,3] and the choice of these formulations affects the efficiency of the underlying search procedure.

2.1 Heat Integration

The use of implicit enumeration with dynamic programming leads to the development of an efficient, robust, and easy to use procedure for synthesis. The attractiveness of this method provides an impetus to consider its application to the more complete problem of the heat-integrated process synthesis. The difficulty with heat integration is the decoupling of the solution to a subproblem, as defined by a stream that requires processing, from the solution to other subproblems which together may form part of the final solution.

The cost of a solution to the non-integrated synthesis problem, (1), is the sum of the costs of the units in that solution. The cost of a unit includes both the capital cost of the base unit and the capital and operating costs of the heat exchangers for the heating and cooling requirements of the unit. For heat integrated designs, the cost of a pair of coupled units decreases due to the reduction in both utility consumption and possibly due to the reduction in the number of heat exchangers required. This coupling requires a modification to the search procedure.

Dhallu & Johns [1] introduced the concept of *heatbits* for simple heat integration. Their implementation of the synthesis procedure, in GENPROC, was based on the use of bit encodings for streams. These bit encodings defined integer indices for lookups in a dynamic programming cost table. By extending the bit encodings to include discrete heat fluxes, they were able to introduce heat integration into the search.

Each discrete heat flux defined a temperature T and a pressure P for a virtual heat stream. In the bit encoding, there were two bits for each heat flux, one used to represent the virtual heat stream as a sink of heat (i.e. one that could be used for cooling a process stream) and one as a heat source (for heating). By ensuring that these heat bits were only ever created in pairs, overall energy balance was ensured.

These heatbits were treated as components in streams: the units were able to use the source and sink bits to reduce their utility requirements. Exchanger costs were defined to ensure global optimality (relative to the choice of the discrete fluxes used) through the use of over-estimates for each half of integrated exchangers. Unused heatbits were passed through the unit and all combinations of assignments to outputs were considered.

2.2 Virtual Heat Links

The approach for heat integration proposed in [1] was extended and implemented in CHiPS [4], the precursor to Jacaranda. This implementation improved the efficiency and the functionality of the approach:

- A unit model interface was defined and the heat integration aspects were moved to the optimization procedure so as to simplify unit model design.
- Sensible heat exchange was allowed and both in-series and in-parallel exchanges were considered.
- A unit model design cache was implemented to make efficient re-use of computation. A discrete approach to heat integration leads to the design of units which differ only in the exchangers for heating and cooling the requirements of the base unit.

In CHiPS, heatbits were known as *virtual heat links* (VHLs).

2.3 Computational Aspects for Early Design

The design of a chemical plant typically takes a long period of time. The actual design of the plant, as represented by a flowsheet, with specifications on the units and streams in that flowsheet, is one that evolves over the length of the project. However, much of the evolution consists of refinements on the base flowsheet which is often fixed early in the design process. Achieving a good early design is crucial to the economic success of the resulting plant.

A good initial design is best achieved by exploring the search space and by considering as many of the criteria that define the quality of a design simultaneously

Table 1. Statistics for 5 component separation problem for different numbers of heat links

n_v	0	1	2
Problems	48	139	453
Nodes	528	20017	1024740
Search time	7	93	4972

and as early as possible. Exploration is encouraged by a combination of usability features in the synthesis package, including problem specification, how easy it is to pose problems when little is known about the overall design desired, and the efficiency with which solutions are generated. In previous work, we have targeted all of these aspects to some degree, including both implementation efficiency [4] and ease of use for early design [2,6,3].

Combinatorial approaches quickly exceed the computational resources available. For example, consider distillation sequence synthesis for separation. Using n_v heat links leads to 3^{4n_v} different alternative allocations for the heat link to the two heat transfer requests (condenser and reboiler) and the two outputs of the unit (which have to be further processed). Each heat link can take on three values: 1. source of heat available, 2. sink for heat available, and 3. no source or sink available. For each subproblem that appears in the non-heat integrated case, there are now three subproblems. Therefore, we expect the time complexity of the search procedure to be $O\left(3^{n_v}\right)$ although the actual growth will be affected by the pruning operations.

Table 1 shows the observed growth in the number of subproblems, the search nodes, and CPU time required as a function of the number of VHLs for a 5 component separation problem [7]. The number of problems grows as predicted. CPU time also behaves as expected: the search time is proportional to the number of problems (the time for unit designs is less trivial to predict due to the use of a unit design cache [4]). This exponential growth limits the size of problems that can be solved.

3 Multilevel Virtual Heat Links

The choice of VHL parameters is critical. A series of synthesis runs are needed to determine appropriate quantities for both temperature and duty parameters. It is useful, therefore, to use a variety of heat links simultaneously so as to give the optimizer the best opportunity to determine the optimal choice. Unfortunately, the growth of the search space precludes the use of more than 1 heat link for large problems. Therefore, we present an alternative approach to handling heat links, one which exhibits a polynomial growth rate but which still allows the use of multiple heat links if desired. This approach, implemented in Jacaranda, uses multilevel virtual heat links (or mVHLs) based on *distributable* duties and multiple temperature levels.

Table 2. Problem size statistics and computer time for 5 component separation problem with 1 heat link with varying numbers of distributable levels

n_q	1	2	3	4
Problems	139	233	325	423
Nodes	20017	144053	539222	1459505
Search time (s)	93	664	2 604	7 175

3.1 Distributable Duties

Each mVHL is defined with a base amount of duty, Q_b, and the number of levels, n_q, used to calculate the actual duty for a given instance of the mVHLs. The duty represented by a link is then one of $i \times Q_b,\ i = -n_q, \ldots, n_q$. The allocation of heat links cycles through all different distributions, ensuring that the overall energy remains balanced.

A reduction in the search space growth is achieved by changing the actual search space itself. In the old approach, the number of exchanges between any two sub-trees in the search space was limited to the number of VHLs available, although the total number of exchanges could be greater than this. The use of one link meant that any two sub-trees could only share one integrated exchanger. By adding the concept of distributable quantities to a heat link, the number cross-links is increased. The maximum number of links between any two sub-trees is the number of levels, n_q. Any number of these links may end up at the same node in either sub-tree but there is no requirement that this be the case. The restriction now is that across any two sub-trees, the total amount of exchange is limited by the total amount associated with the heat link used and the number of exchangers in each sub-tree for a given link is less than or equal to n_q.

The size of the search space is now a function of both the number of heat links and the number of levels for each heat link. The number of subproblems, assuming that the number of levels, $l = 2 \times n_q + 1$, per link is the same, should grow as $O\left(l^{n_v}\right)$. The number of nodes, for the distillation example, is $O\left(l^{4n_v}\right)$. The new approach exhibits polynomial growth in l (and hence in n_q), assuming a constant n_v, confirmed experimentally in Table 2.

These results indicate that with a single mVHL we can handle up to 4 discrete duty amounts for approximately the same amount of computer time as 2 single level VHLs. The intention is that the number of actual links required for identifying suitable process flowsheets in early design can be greatly reduced, albeit with an increase in the number of levels per link.

3.2 Multiple Temperatures

The second extension embodied in mVHLs is the use of multiple temperature levels. Instead of just one temperature per heat link, the mVHL has a discrete set of temperatures, $\{T_i, i = 1, \ldots, n_t\}$. The cartesian product of the temperature set with the

Table 3. Feed specification for the 10 component separation problem with flows in kmol/hr

Component	Fraction	Flow	Component	Fraction	Flow
Ethane	0.181	245	iso-Pentane	0.044	80
Propane	0.037	50	Hexane	0.074	100
Butane	0.052	150	Heptane	0.258	350
iso-Butane	0.111	70	Octane	0.148	200
Pentane	0.059	60	Nonane	0.037	50

discrete duty amounts,

$$\{[T_1, -n_\mathrm{q} \times Q_\mathrm{b}], [T_1, -(n_\mathrm{q} - 1)Q_\mathrm{b}], \ldots, [T_{n_\mathrm{t}}, n_\mathrm{q} \times Q_\mathrm{b}]\} \ ,$$

represents the $l = 2 \times n_\mathrm{t} \times n_\mathrm{q} + 1$ different heat links embodied in a single mVHL. Although there are n_t different encodings in the cartesian product which represent no heat transfer at all, for efficiency in enumeration, these all collapse to one single representation in the implementation. The growth rate in the search space expected is the same as given in the previous section. The only difference is the reduction in the number of actual feasible allocations because the energy balance must now include the temperature as well. In general, there will be less feasible allocations as a function of l in this case.

With temperature levels, we should be able to tackle problems in which the temperature varies significantly over the different units in the possible flowsheets. Note, however, that if a mVHL has a single temperature and a single duty level then it is identical to the previous concept of a heat link.

4 Results

The new method has been used to find the best distillation sequence for a 10 component hydrocarbon feed stream [5] (Table 3). The models used for the distillation columns and heat exchangers, the set of discrete utilities available, and all cost formulations are described in [7]. The design parameters for the distillation units include the pressure (1–30 atm) and reflux rate. Suitable temperatures and duties are not known a priori so this problem is an ideal test for the new approach: we would like to consider running this problem with a variety of temperature and duty combinations.

Analysis on the data generated synthesis without heat integration indicates that heating requests are mostly for temperatures less than 425 K while cooling requests are for temperatures greater than 330. The majority of duty requests lie in the 0–10 MW range. Based on this analysis, we attempted this problem using both the old approach and the new approach. For the new, we tried a single multi-level heat link: $T \in \{360, 390\}$, $Q_\mathrm{b} = 5$ MW and $n_\mathrm{q} = 2$, giving a total of 4 discrete duty amounts. To ensure at least as good a solution with the old approach would require 4 VHLs, which is computationally intractable. However, we tried the equivalent four single

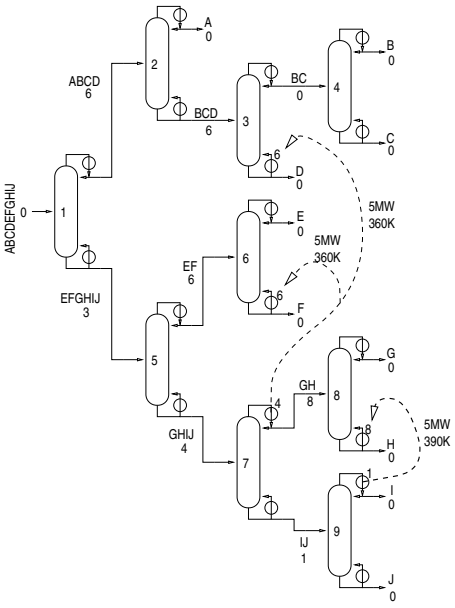

Fig. 1. Best process identified with multi-level virtual heat link showing heat link allocations and integrated exchangers

Table 4. mVHL level identification used in Figure 1 for 10 component synthesis problem

Id	1	2	3	4	5	6	7	8
T (K)	390	390	360	360	360	360	390	390
Q (MW)	-5	-10	-5	-10	10	5	10	5

VHL problems. In all cases, the process identified as optimal was then passed to a genetic algorithm based post-synthesis optimization procedure for fine tuning [8].

The non-heat integrated case as well as all four single VHL cases resulted in the same separation sequence with a final annualized cost of 1.77 M$/yr. The mVHL approach identified a process, shown in Fig. 1, with a cost of 1.73 M$/yr, 2.3% cheaper than the other solutions. In the figure, the streams have been labelled with both the composition and the particular mVHL associated with the stream, using the assignments in Table 4. Heat-integrated columns are identified by dashed lines showing flow of heat.

Heat links are balanced around each unit. For example, the feed to unit 5 has a link, 3, which represents a sink of 5 MW of heat at 360 K. The bottom product has a sink of 10 MW at 360 K and the tops a source link of 5 MW of heat at 360 K. This particular node also illustrates one of the features of the new heat link approach: because there can be two different quantities associated with any particular temperature, one end of a link can have two corresponding matching ends. In this case, the sink link ends up at unit 7. It is matched with two source links at units 3 and 6. The old approach would have required 2 VHLs to achieve the same result.

Another integration is identified further down the bottom branch. This one uses the link's other temperature: 390 K. The temperatures at the bottom of the process

tend to be higher than those at the top and for a problem of this size, a single temperature would not have been sufficient to provide a reasonable search space. Combined with the distributable nature of the new mVHLs, a single multi-level heat link is able to identify a process which would require the simultaneous use of at least three different VHLs.

The post-synthesis processing leads to a similar heat exchanger network except that excess heat from unit 9 is used for heating in unit 2 instead of unit 8. This could not be identified by the single heat link instance due to the limitation in the number of matches across any two sub-trees. However, the overall transfer from unit 9 is the same: approximately 5 MW so the overall reduction in energy consumption by this process is similar to the pre-tuned process. There is a completely new match found, from unit 6 to unit 4, but it is only for a small amount of approximately 0.2 MW.

5 Conclusions

We have presented a novel approach for incorporating heat integration into a discrete process synthesis procedure. The new approach implicitly generates a different search space than previous methods. This search space is more suited for early design and the result is an algorithm with polynomial time requirements. For early design, time is a particularly important human factor as exploration must be encouraged.

Acknowledgements. The authors would like to acknowledge the support provided by the EPSRC.

References

1. Dhallu, N S & Johns, W R, 1988, IChemE Symp Ser **109**:23–42.
2. Fraga, E S, 1996, Chem Eng Res & Des **74**(A):249–260.
3. Fraga, E S, 1998, Chem Eng Res & Des **76**(A1):45–54.
4. Fraga, E S & McKinnon, K I M, 1994, Chem Eng Res & Des **72**(A3):389–394.
5. Fraga, E S & McKinnon, K I M, 1995, Computers chem. Engng **19**(6/7):759–773.
6. Laing, D M & Fraga, E S, 1997, Computers chem. Engng **21**(Suppl.):S53–S58.
7. Rathore, R N S, van Wormer, K A & Powers, G J, 1974, AIChE J **20**(3):491–502.
8. C Stair & E S Fraga, 1995, Proc. 1995 IChemE Research Event, 95–97.