

Applying CBR and Object Database Techniques in Chemical Process Design

Timo Seuranen¹, Elina Pajula², Markku Hurme¹

¹ Helsinki University of Technology, Laboratory of Chemical Engineering and Plant Design, P.O. Box 6100, FIN-02015 HUT, Finland
timo.seuranen@hut.fi

² Oy Keskuslaboratorio – Centrallaboratorium Ab, P.O. Box 70, FIN-02151 ESPOO, Finland

Abstract. The aim of this paper is to introduce a new method for finding and reusing process equipment design and inherently safer process configurations by case-based reasoning (CBR) and object database techniques. CBR is based on finding most alike existing solutions and applying the knowledge of their properties for solving new problems in the early phases of design. This supports design engineer's knowledge by allowing a systematic reuse of existing experience in order to improve the quality and safety of new designs. The possibilities of CBR and object database techniques in chemical process engineering field have been illustrated by two prototype applications.

1. Introduction

Case-based reasoning (CBR) is a widely used problem solving technique, which has been successfully applied to a large range of different tasks [1]. CBR is based on the reuse of proven solutions when solving new problems. The current problem is defined as a query by giving the essential parameters e.g. reactor type and relief system. Based on these, the similarity is calculated and a user-defined number of the most similar cases are retrieved. The user can select a case and launch adaptation routines, e.g. scale-up calculations or adapt the case using other simulators. The more similar cases can be found the fewer accurate simulations are needed because a large part of the design can utilize data already available in the existing cases.

Process design applications like process synthesis are very complex problems and currently available commercial CBR tools don't support all the properties needed. For instance, the database engine of the tools is based on relational database technique, which limits representing complex problems. An object-oriented database approach would allow more flexible application developing [2].

CBR and object database prototype applications in this paper are developed with C++ programming language and an object database programming language. The first prototype application was developed to aid heat exchange equipment design. The program generates necessary input data including mechanical configuration for heat exchanger design simulators. The second prototype application was developed for

conceptual process design, especially to find out if known inherently safer solutions to the current process design problem exist. The found safer alternatives can then be used as an evolutionary manner to substitute less safe process features in different levels of detail.

1.1 Case-based reasoning

The method presented is based on case-based reasoning (CBR) and an object database approach. The database contains collection design cases collected from open literature and existing process designs.

To find the nearest existing design a set of parameter values, e.g. reactor type, of all cases in the case base are compared to the input data given by the user. The retrieval phase uses several distance functions [1]. The idea is to define a distance between case values and input values. In case of numerical data types, distance can be determined following way [3]. If the input value is smaller than the minimum or greater than the maximum of the case values $C_{ij}(X_i < \min C_{ij} \text{ or } X_i > \max C_{ij})$:

$$D_{i,j} = \left(\frac{C_{i,j} - X_i}{C_{i,j}} \right)^2 \quad (1)$$

where $D_{i,j}$ is distance between the case value and the input value, $C_{i,j}$ is value of parameter i of case j , and X_i is the input value of parameter i . If the input value is smaller than the minimum or greater than the maximum of the case values $C_{ij}(\min C_{ij} \leq X_i \leq \max C_{ij})$:

$$D_{i,j} = \left(\frac{C_{i,j} - X_i}{\max C_{i,j} - \min C_{i,j}} \right)^2 \quad (2)$$

Similarity between different cases can be presented with a normalized distance:

$$Y_{i,j} = 1 - \frac{D_{i,j}}{\max D_{i,j}} \quad (3)$$

where $Y_{i,j}$ is similarity of the parameter i for the case j . In case of string data types similarity is simply:

$$\begin{aligned} X_{i,j} = C_{i,j} &\Rightarrow Y_{i,j} = 1 \\ X_{i,j} \neq C_{i,j} &\Rightarrow Y_{i,j} = 0 \end{aligned} \quad (4)$$

In case of linguistic type data types fuzzy set operations can be used for the determination of a similarity. Linguistic values can be transformed into membership functions, and the similarity can be expressed as an intersection of fuzzy sets: [3]

$$Y_{i,j} = \min\{\mathbf{m}_x(z), \mathbf{m}_t(z)\} \quad (5)$$

where \mathbf{m}_t is value of the linguistic parameter i for the case j .

The quality of reasoning increases if importance of selection parameters can be altered. In case of inherent safety in exothermic reactor design, reaction enthalpy and available heat transfer area may be more important selection parameters than type of impeller used. Weighted similarity can be expressed: [3]

$$R_{i,j} = W_i Y_{i,j} \quad (6)$$

where W_i is the weight factor of the selection parameter i evaluated by the user. Overall similarity can be calculated for a case j based on the number of parameters and based on parametric similarities:

$$S_j = \frac{\sum_{i=1}^N R_{i,j}}{N} \quad (7)$$

where N is number of parameters. Including an estimation of design quality for the case retrieval can also increase the quality of reasoning. Eq. (7) can be substituted with:

$$S_j = DQ_j \frac{\sum_{i=1}^N R_{i,j}}{N} \quad (8)$$

where DQ_j is design quality factor for case j .

1.2 Object database

Chemical processes have very complex structure. The major problem in modeling such systems is the management of relationships between e.g. processes, equipment, components, and materials. Objects are well suited to the natural modeling of relationships. The semantic relationships among objects can be used to design programs that are modular, contain well-defined interfaces, and are structured along the lines of the problem to be solved. Using the object-oriented programming model intelligent way results in programs that are well structured and easier to understand [4].

An object database supports the object-oriented model. Like an object-oriented programming language, it is designed to express the relationships among data. Like a conventional database, it is designed to manage large amounts of data and perform fast value-based queries. In object database programming the system developer is not forced to use any pre-described data storing formalism. Domain specific, complex data can be declared as data types of their own [2]. This is benefit especially in conceptual process design since the process structure is complicated and a flexible data structure is needed to represent the information. In this presentation the data types are object classes identified as entities in the knowledge related to real world processes and process design. An object database is a collection of user defined object classes, e.g. reactor, heat exchanger, that are declared persistent.

A part of the basic structure of the persistent class diagram of the safety case-base (second application) is represented with UML-object modeling language (fig. 1).

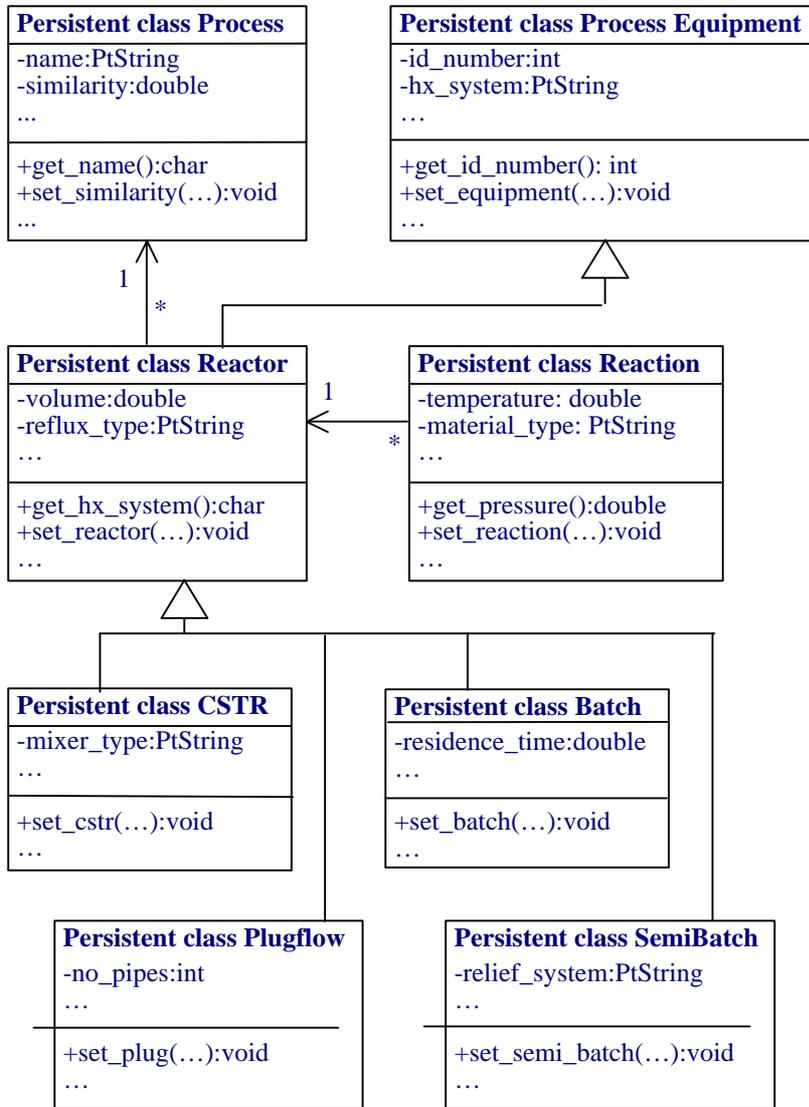


Fig. 1. A part of the persistent class diagram of the safety case-base.

2. Existing Process Design Approaches

The existing chemical process design approaches can be divided to:

1. Heuristic and engineering experience based methods which use often an hierarchical approach [5]
2. Optimization approaches using either mixed integer nonlinear programming (MINLP) [6] or genetic algorithms (GA) [7].
3. CBR methods where existing design cases are reused and adapted to solve new design problems. This approach has been used to some extent to equipment design in chemical engineering [8,9,10] but very little to process design.

The problem in heuristic approaches is that no computer-based support systems are usually available to support in heuristic conceptual methods. Therefore the approach relies totally on the designer.

The optimization-based methods require that an explicit objective function have to be defined for the method. However many design criteria such as safety or operability are difficult or impossible to quantify explicitly. The optimization approaches are not very interactive; the optimization cannot be interfered or guided easily by the user. For these reasons the optimization methods often lead to impractical or even impossible solutions. The use of optimization requires that the optimization alternatives and area has to be limited by the user. This task called 'generation of superstructure' has to be done by the user and can be a tedious procedure, which if done improperly has a detrimental effect on the result.

Therefore the main differences of the approaches are related to the interaction with the user, requirement of superstructure and the possibility of combinatorial explosion. In MINLP and GA a superstructure is required for the optimization algorithm. To our experience the differences of methods can be summarized as follows:

| | MINLP | GA | CBR |
|-------------------------|-------|------|-----|
| superstructure required | yes | yes | no |
| combinatorial explosion | yes | some | no |
| noninteractive method | yes | yes | no |

3. The Prototype Application Programs

The prototype application programs have been built using MS Visual C++ 6.0 programming language. The object databases have been built by POET (Persistent Objects and Extended database Technology) an object database programming language and fully integrated with C++-language.

3.1 Shell-and-Tube Heat Exchanger Selection Application

The basic idea of the heat exchanger selection application is to use existing designs of heat exchangers for creating input parameters for heat exchanger design or dynamic process simulation programs. The most similar existing case is retrieved from the case-base. Results from retrieval are necessary input data to for example dynamic process simulators, which require information on the type and mechanical dimensions of the heat exchangers. Also a rigorous exchanger design, which is normally done by simulation programs such as HTRI or HTFS, can benefit from CBR by the reuse of design and operation information on the existing heat exchangers. Many aspects such as fouling and the feasible exchanger types are experience-based information. The detailed adaptation can be done using a heat exchanger simulator. Such simulator consists of necessary thermal design calculations for detailed heat exchanger design. The major benefit of CBR when applied in the equipment design is that it offers the experience of earlier designs needed in many engineering design tasks.

The heat exchanger model in the case-base is very detailed. Input parameters are in this case: fluid types, mass flows, operating pressures, possible phase changes, temperature differences between inflow and outflow and fluid temperatures of inflows etc. Output parameters define detailed fluid data, heat exchanger operating data, and heat exchanger mechanical design data including design quality. Retrieval calculations are constructed as described in section 1.1.

Results from retrieval are the TEMA type for heat exchangers, position (vertical/horizontal), shells per unit, number of units, and number of passes. The TEMA type defines front-end head types, rear end head types, and shell types of shell-and-tube heat exchangers.

The demonstration of the heat exchanger program is shown in Figs. 2, 3 and 4. The selected input parameters with weight factors are shown in Fig. 2. The results from the retrieval with input values are presented in Fig. 3. Heat exchanger configuration of the most similar case with a user given input data is used for heat exchanger simulator calculations, Fig. 4. The heat exchanger selection is done, when results from the simulation are satisfactory. The heat exchangers in the case-base can also be studied by built-in queries.

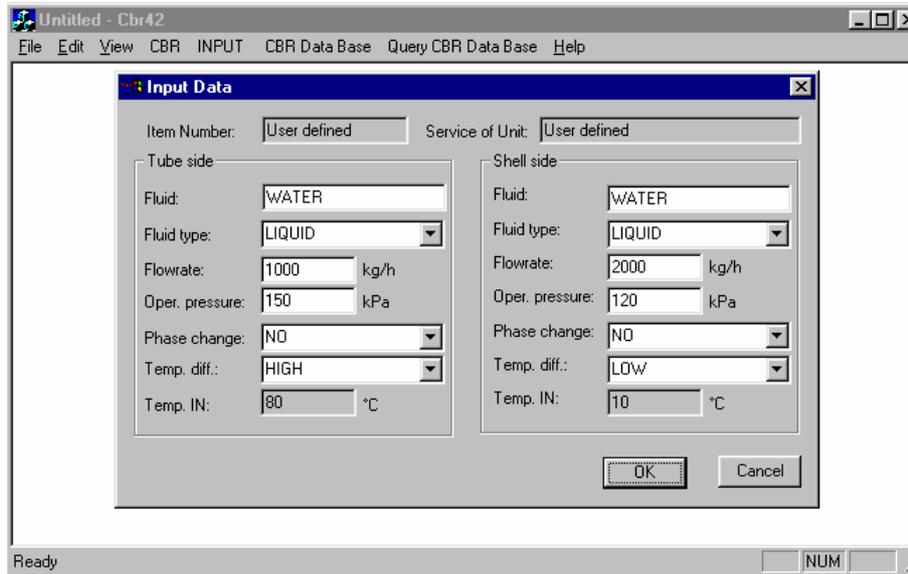


Fig. 2. User input data

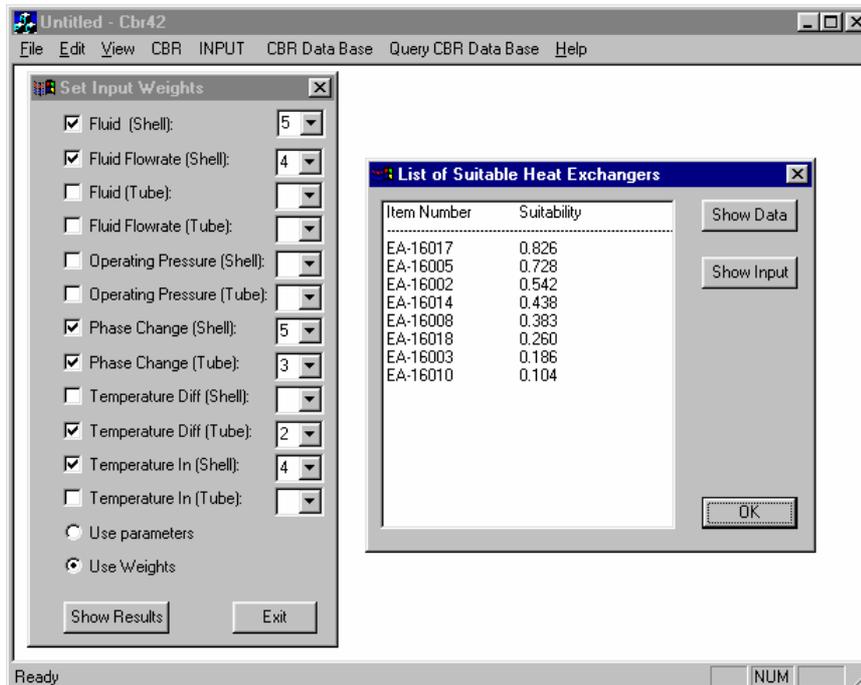


Fig. 3. Input parameters and weight factors for the query, and query result window.

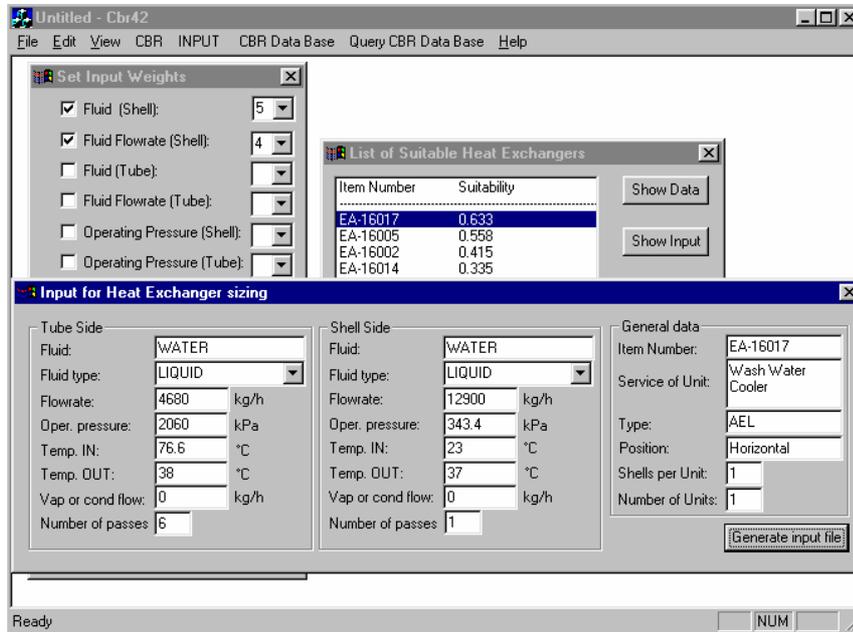


Fig. 4. Input data for dynamic heat exchanger design simulators.

3.2 Emulsion Polymerization Process

Process design by case-based reasoning methods is a very new approach even it has an obviously potential. CBR would combine the traditional engineering experience-based methods with computer based design support systems, which enable the systematic reuse of exiting design information and hereby can form an institutional memory of a company. The systematic use of existing information and the feed back of the successful existing designs is a method of continuous improvement of engineering work in a company.

There are very few CBR applications in process design in literature. Surma and Braunschweig [11] and Heikkilä et al. [12] used CBR for finding similar process designs in database but they did not use CBR for process design itself. King et al. [13] used CBR for designing azeotropic distillation systems by using residue curve approach. However the design domain was very limited. Pajula et al. [14,15] presented a general approach for using CBR for separation process synthesis.

A polymerization reaction system case study is given here to demonstrate, how the CBR method is used to improve the inherent safety of the process concept in an evolutionary manner by reasoning on several levels of detail. First the reasoning is made on a process level to find safe general concepts for the process. Then the different systems are studied one by one. If safer alternatives to existing design are found in database, they are substituted to the design. The evaluation can be based on users engineering judgment or safety indices such as the Inherent Safety Index [16]. The value of the index can be included into the cases of database.

The process of case study is an emulsion polymerization, in which unsaturated monomers or their solutions are dispersed in a continuous phase with the aid of an emulsifier [17]. The product is a dispersion of polymers, latex. The raw materials are highly flammable unsaturated hydrocarbons and the reaction is exothermic.

A case base was formed using information based on the safety properties, accidents, design recommendations and existing designs of polymerization processes. The information includes also general design recommendations for reactor systems. The main variables in the emulsion polymerization process are [17]; 1) premixing, 2) pre-heating, 3) reactor type and mode of operation, 4) number of reactors, 5) reactor construction material, 6) reactor mixing, 7) baffles in reactor, 8) reactor heat transfer system, 9) method of liquid transfers, 10) relief system location, 11) relief equipment type, 12) reaction stopping method, 13) relief recovery system, 14) vent treatment equipment.

The process can be studied in various levels to find out which kinds of design cases (recommendations on good designs or warnings on bad case; e.g. accidents) are available to improve the design. First the study is made in an upper (process) level and then in the more detailed levels. The user interface has windows for both reactor and reaction data (Fig. 5). By altering which parameters in use and how they are weighted, the user can make searches in various levels.

The query results are shown in a separate window (Fig. 6). In this query cases containing recommendations in process level are found. According to the found case the application of a semibatch reactor process is advisable to minimize a runaway hazard apparent in this case (Fig. 7). In the system level recommendations for the reactor + cooling system concept can be searched for exothermic reactions. The found cases can be adapted for reactor size, heat of reaction, heat capacity and heat transfer area to correspond the existing case. In equipment level, the possible relief equipment can be studied using queries in the case-base.

In every level of reasoning the found potentially safer solutions are used to substitute the features in existing design. In this way the design is improved gradually in an evolutionary way. The assessment which design is better to another can be based for instance to safety indices [16] or users own judgment. Also the textual case descriptions include qualitative evaluations on the safety features of designs found in the database.

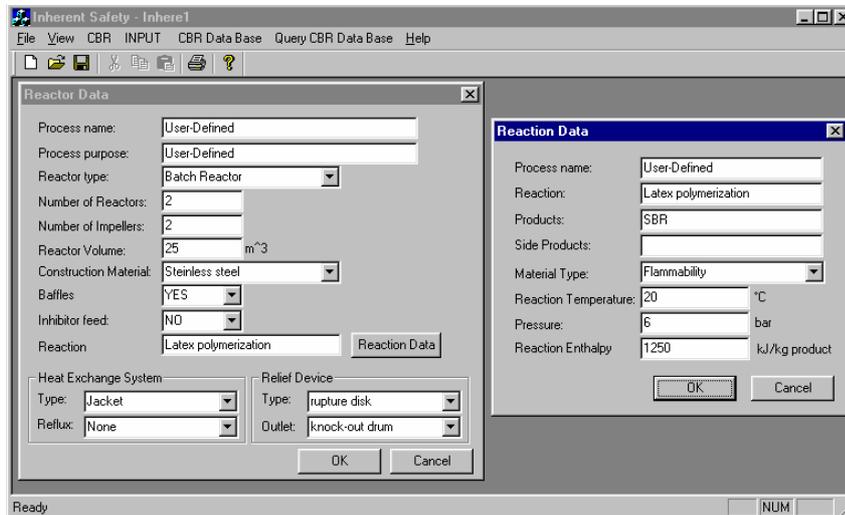


Fig. 5. User input data.

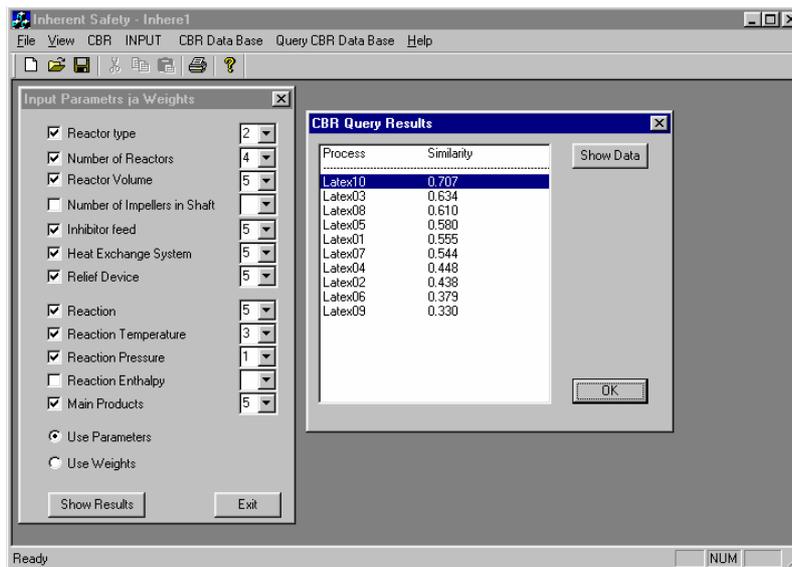


Fig. 6. Input parameters and weight factors for the query, and query result window.

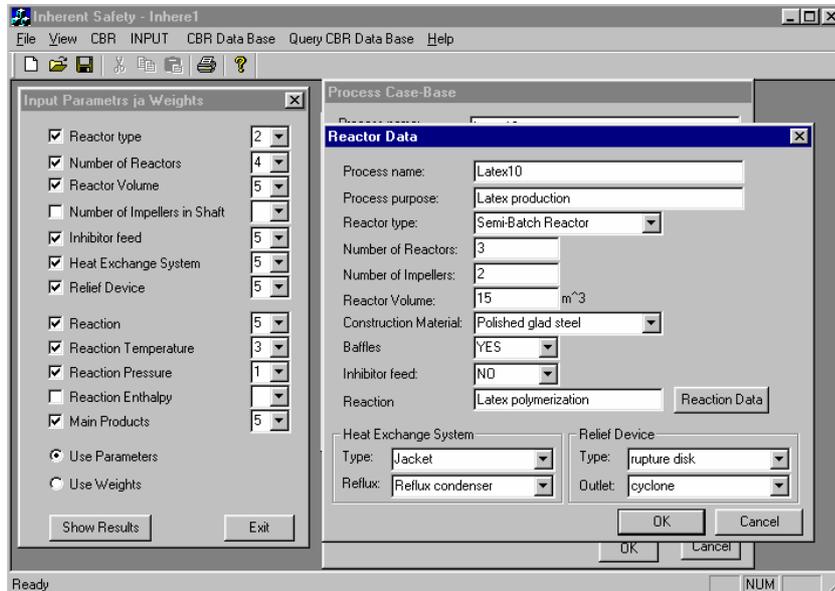


Fig. 7. The best retrieved process.

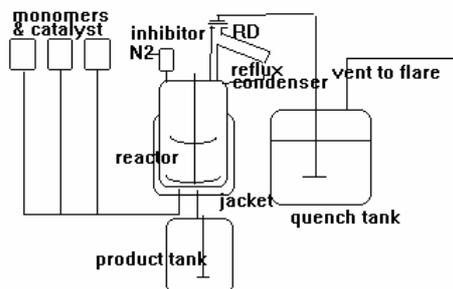


Fig. 8. The improved emulsion polymerisation system.

The final process designed has a semibatch (instead of batch) reactor, which results to a low monomer inventory in the reactor (Fig. 8). In this configuration, the reaction starts immediately when the first monomers are fed to the reactor. There is neither premixing tank for the monomers, which would increase the inventory. Only one large reactor is used instead of several smaller ones to make the system simpler. Two impellers at one shaft are installed to increase the mixing efficiency when the liquid level in tank is changing. Baffles are used for increasing the mixing efficiency. Both a jacket and a reflux condenser accomplish cooling. The reactor construction material is polished glad steel that has better heat transfer properties compared to a glass-lined vessel. Liquid transfers to and from the reactor are accomplished using elevation to reduce number of pumps. The relief system of the reactor includes a rupture disk, which is safer than a relief valve alone in the fouling conditions. The relief is led to a quench

tank, which contains quench liquid to stop the reaction and to separate the liquid phase from the relief. This is safer than an ordinary knock-out drum or cyclone. The vent is led to flare system after the quench tank. Ordinary flare is used, since it has a larger capacity than a controlled collection system or a scrubber. An inhibitor addition to reactor is also included to stop the reaction chemically.

A quantitative verification that the improved process presented in Fig. 8 is better than the conventional batch process described in literature [17] can be based on the calculation of Inherent Safety Index [16] values of both processes. The conventional process has an index value of about 28-30 compared to the index value 16 of the improved process shown in Fig. 8.

4. Conclusions

Two prototype applications have been developed to illustrate the use of CBR and object database techniques in process equipment design and inherently safer process design. Main advantages of CBR approach are the directing of user to a systematic documentation practice, and utilization of documentation. The time consumed for routine design can be reduced as a consequence of efficient reuse of existing knowledge. It is obvious that case-based reasoning is a narrow field of application, because configuration and structure of equipment or processes should be somewhat similar to each other.

In the heat exchanger application, the design quality is included. In this case, design quality is one evaluation parameter in the calculation of case similarity. In fact, design quality should be a combination of several parameters such as equipment safety, operational reliability, and economy. Quality parameters are also time dependent, which leads to following the lifetime of equipment in order to get a good case base.

The benefits of the CBR approach presented in the inherent safety application are the following: Inherent safety is difficult to implement without concrete design examples since the principles are very general. The system presented provides real design cases that can be applied to substitute the current designs in the problem under study. The cases represent both good design practices and known accident cases, which have lead to process improvements. If the processes under design and operation are systematically studied in various levels as presented, the reuse of existing design and safety knowledge is greatly enhanced, less design mistakes are made and consequently conceptually safer processes are created.

Object-oriented approach and object database techniques allow more flexible application developing especially in complex process design cases. Object database combines the semantic of object-oriented approach with data management and query facilities of a database system.

References

1. Kolodner, J., Case-based Reasoning, Morgan Kaufman Publishers Inc., San Mateo, California (1993).
2. Loomis, M., Object Databases: The Essentials, Addison-Wesley Publishing Company, Menlo Park, California (1995).
3. Koironen, T., Hurme, M., Case-based Reasoning Applications in Process Equipment Selection and Design, Scandinavian Conference of Artificial Intelligence SCAI'97, University of Helsinki, Finland (1997).
4. Booch, G., Object Oriented Design with Applications, The Benjamin/Cummings Publishing Company Inc, California (1991).
5. Douglas, J.M., *Conceptual Design of Chemical Processes*, McGraw-Hill, New York 1988.
6. Grossmann, I.E., Kravanja, Z., Mixed-Integer Nonlinear Programming Techniques for Process Systems Engineering *Comp. Chem. Eng.* 19 (1995) Suppl., S189-S204
7. Khalil, M.S., Evolutionary Methods in Chemical Engineering, Plant Design Report Series No. 64, Helsinki University of Technology, Espoo (2000), 143 pp.
8. Pajula, E., Koironen, T., Seuranen, T., Hurme, M., Computer Aided Process Equipment Design from Equipment Parts, *Comp. Chem. Eng.* 23 (1999) Suppl., S683-S686.
9. Kraslawski, A., Koironen, T., Nyström, L., Case-Based Reasoning System for Mixing Equipment Selection, *Comp. Chem. Eng.* 19 (1995) Suppl., S821-S826.
10. Virkki-Hatakka, T., Kraslawski, A., Koironen, T., Nyström, L., Adaptation Phase in Case-Based Reasoning System for Process Equipment Selection, *Comp. Chem. Eng.* 21 (1997) Suppl., S643-S648.
11. Surma, J., Braunschweig, B., Case-Base Retrieval in Process Engineering: Supporting Design by Reusing Flowsheets, *Engineering Applications of Artificial Intelligence* 9 (1996) 385-391.
12. Heikkilä, A.-M., Koironen, T., Hurme, M., Application of case-based reasoning to safety evaluation of process configuration, HAZARDS XIV, Institution of Chemical Engineers Symposium Series No. 144, IChemE, Rugby (1998), 461-473.
13. King, J.M.P., Bañares-Alcántara, R., Zainuddin, A.M, Minimising environmental impact using CBR: An azeotropic distillation case study, *Environmental Modelling & Software* 14 (1999) 395-366
14. Pajula, E., Seuranen, T., Hurme, M. Synthesis of separation sequences by case-based reasoning, *Comp. Chem. Eng.* vol 11, Elsevier 2001, to appear.
15. Pajula, E., Seuranen, T., Koironen, T., Hurme, M. Synthesis of separation processes by using case-based reasoning, *Comp. Chem. Eng.* 25 (2001) 775-782.
16. Heikkilä, A.-M., Hurme, M., Järveläinen, M., Safety Considerations in Process Synthesis, *Comp. Chem. Eng.* 20 (1996) Suppl. S115-S120
17. Kroschwitz, J.I. (Ed.), *Encyclopedia of polymer science and engineering*, Vol 6, Wiley, New York (1986).