HAL
open science

Understanding the Dynamics of State-problems in Distributed and Collaborative Design Process

Reza Movahed-Khah, Egon Ostrosi, Olivier Garro

To cite this version:

Reza Movahed-Khah, Egon Ostrosi, Olivier Garro. Understanding the Dynamics of State-problems in Distributed and Collaborative Design Process. Concurrent Engineering: Research and Applications, 2007, 15 (1), pp.33-41. 10.1177/1063293X07076701 . hal-00571204

HAL Id: hal-00571204

https://hal.science/hal-00571204v1

Submitted on 1 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CONCURRENT ENGINEERING: Research and Applications

Understanding the Dynamics of State-problems in Distributed and Collaborative Design Process

Reza Movahed-Khah,* **Egon Ostrosi and Olivier Garro**

Laboratoire de recherche Mécatronique3M, Université de Technologie de Belfort-Montbéliard, 90010 Belfort Cedex, France

Abstract: The implementation of multi-agents systems aided collaborative and distributed design requires a deeper understanding of the real interactions between actors, inside of the multidisciplinary teams. In this view, it is considered that for the implementation of these systems, it is necessary to observe, model, and analyze this process with finer granularities levels. This study presents an approach of collaborative and distributed design process analysis. The core of this approach is the modeling of interactions between actors during the design process. The approach consists in discerning, from the real interactions and the emergence of concepts, the different characteristic state-problems of the collaborative and distributed design process. The analysis of state-problems, in a real experience of collaborative and distributed design process, shows that their dynamics is rather characterized by the numerous irregular leaps. These leaps, probably unforeseeable, show that the design process is far from harmonious. Indeed, these leaps depend strongly on either the structural causes, as the auto-organization inside the group, or on the human actions, as the creative character of the design process.

Key Words: collaborative and distributed design process, multi-agents systems, teamwork, experimental analysis, interactions analysis, data analysis.

1. Introduction

In the multi-agents systems (MAS) aided collaborative and distributed design, the agents are supposed to be intelligent. In this context, each agent must interact effectively with the other agents and thus achieve the total goals of the collaborative and distributed design process. The implementation of the MAS aided collaborative and distributed design requires a thorough comprehension of the real interactions between the actors inside the multidisciplinary teams. For that, it is necessary to observe, model, and analyze the collaborative and distributed design process to fine levels of granularity [17].

Although, several researches were dedicated to the study of collaborative design process [11], there is little information available about the analysis of collaborative and distributed design process in order to develop MAS aided collaborative and distributed design. Indeed, the collaborative and distributed design is a complex process [3,15,16]. This complexity results from the conjugation of a great number of heterogeneous data (domains, actors, organizations, methods) that are interacting between them [8]. Moreover, the variety of the points of view results in multiple goals to attempt during the design process. The interaction between the actors, during the collaborative and distributed design process, shows that this one is a key variable. In a majority of cases, the results issued from the interactions must be consensual in order to be accepted. Under these conditions, the final solution of the design process can result only from one consensus. The comprehension of convergence towards an acceptable solution, as a whole, requires a modeling of variables intervening for the period of the interactions between the different actors [4], the aims and the relations which they maintain during the design process.

According to Akman et al. [1] only systems which embed advanced reasoning capabilities will be able to deal with the complexity arising from the management of large quantities of design data. The research question then is how best to support the 'reasoning capabilities' the design information management system should possess. If designers spend, on the one hand, 80% of the time generating and retrieving their data [2], and on the other hand, 93% of the time assessing information on a non-quantitative level of abstraction, then

*Author to whom correspondence should be addressed.
E-mail: reza.movahedkhah@univ-angers.fr
Figures 4, 5, 7 and 9–15 appear in color online: http://cer.sagepub.com

1063-293X/07/01 0033–9 $10.00/0 DOI: 10.1177/1063293X07076701
© 2007 SAGE Publications

information tools should offer external memory aids to retrieve these data [7]. In this direction, Dong et al. [5] developed a computable learning method to extract the content of the design model to facilitate the information sharing between designers. This problem is strongly related to the perception of how teams exploit shared and organized comprehension [10], as well as the mental representation of knowledge about the overall design problem [6,18]. Inside the design teams, as the designers must communicate their thoughts between them, the verbal communication offers a fairly direct path to the state of the design process and its related problem, which is called micro state-problem.

In this study, an approach is proposed that, beginning with the real interactions and the concepts emergence, discerns the different micro state-problems and their dynamics characterizing the collaborative and distributed design process. In the second section, the framework of this approach is presented. Here, the identification of the micro state-problems and their evolutions in time permits deepened understanding of the dynamics of the problems that occur during the collaborative and distributed design process. The developed approach is demonstrated by a collaborative design experience of the GRACC group (**GRACC: G**roupe de **R**echerche sur l'**A**ctivité de **C**onception **C**oopérative). In the third section the important results of this approach are summarized.

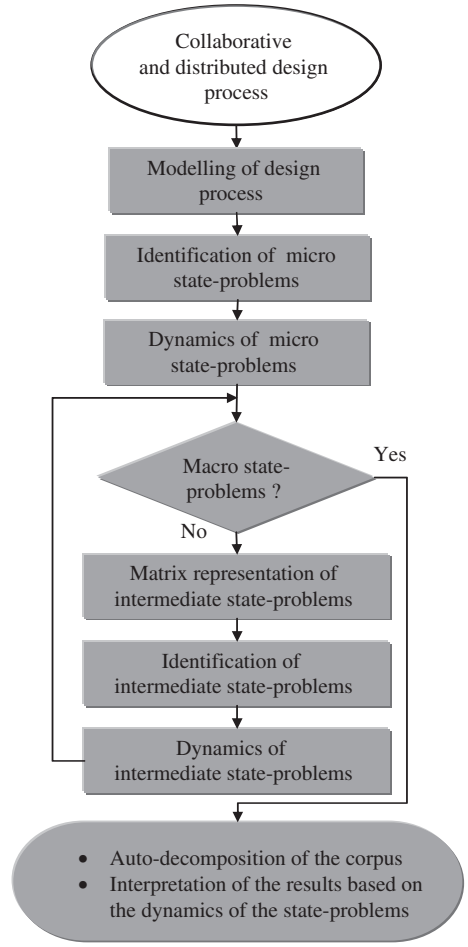

Figure 1. Flow chart of proposed approach.

2. Analysis of the collaborative and distributed design process

The framework of the proposed analysis is given in Figure 1. The approach performs in the following phases:

1. **Modeling of design process.** In this level, the corpus is represented as a set of interactions between the designers, where each interaction is represented as a set of entities of analysis.
2. **Identification of micro state-problems.** In this level, based on the methods of cross-clustering, the concepts and the corresponding micro state-problems are searched.
3. **Dynamics of micro state-problems.** In this level, the evolution of state-problems in the time is represented.
4. **Identification of intermediate and macro state-problems.** In this level, based on the recursive aggregation the intermediate and final state-problems are identified and their dynamics are represented.

2.1 Modeling of design process

During the collaborative and distributed design process, each domain is represented by an actor, and has specific responsibilities in the design process. Thus, each actor is authorized to be an expert on certain fields of knowledge called registers of reference. $\boldsymbol{q}$ is the number of actors and $\boldsymbol{A_k}$ $(k=1,\ldots,q)$ the actor $\boldsymbol{k}$ corresponding to the register of reference k. During the design process, the actors interact. For example, an actor $\boldsymbol{A_k}$ proposes, at the moment $\boldsymbol{t}$, a conjecture related to a problem. It is about a potential solution, candidate to become an entire solution. An actor $\boldsymbol{A_l}$ representing the domain $\boldsymbol{l}$, reacts to this proposition. He advances, for example, a criterion of evaluation of the proposed conjecture. $\boldsymbol{Int_i}$, $i=1,\ldots,n,$ the $\boldsymbol{I}$th interaction, with $\boldsymbol{n}$ the number of interactions. In the most elementary form, an interaction $\boldsymbol{Int_i}$ of an actor $\boldsymbol{A_k}$, at the moment $\boldsymbol{t}$, is characterized by one or several transmitted messages. Thus, a message is considered as being a form of representation of the knowledge domain. It can be characterized by a syntactic element (e.g., verb or noun), with a specific semantic to a knowledge domain. These elements are called entities of analysis and $\boldsymbol{e_j}$, $j=1,\ldots,m$.

$$Corpus = \begin{bmatrix} Int_1 \\ Int_2 \\ Int_3 \\ \dots \\ Int_n \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 & \dots & e_a \\ e_{a+1} & e_{a+2} & e_{a+3} & \dots & e_b \\ e_{b+1} & e_{b+2} & e_{b+3} & \dots & e_c \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & e_m \end{bmatrix}$$

Figure 2. Matrix model of a corpus (interactions-entities of analysis).

$$C[c_{ij}] = \begin{array}{c} \\ Int'_1 \\ Int'_2 \\ Int'_3 \\ \dots \\ Int'_n \end{array} \begin{array}{c} \begin{array}{ccccc} e_1 & e_2 & e_3 & \dots\dots & e_m \end{array} \\ \begin{bmatrix} c_{11} & c_{12} & c_{13} & \dots & c_{1m} \\ c_{21} & c_{22} & c_{23} & \dots & c_{2m} \\ c_{31} & c_{32} & c_{33} & \dots & c_{3m} \\ \dots & \dots & \dots & \dots & \dots \\ c_{n1} & c_{n2} & c_{n3} & \dots & c_{nm} \end{bmatrix} \end{array}$$

Figure 3. Matrix C[c_{ij}].

Figure 4. Children's trailer.

For example, {<tighten>, <screw>, <transmit>, <force>...} are some entities of analysis. Then, the corpus can be represented as a set of interactions, which on the other hand, can be represented as a set of entities of analysis (Figure 2).

Then, for analyzing the collaborative and distributed design process, each interaction is filtered with the aid of the entities of analysis. The filtered interaction $\boldsymbol{Int_i}$ is called enriched interaction. It is denoted as $\boldsymbol{Int'_i}$. Then, the relationship between the enriched interactions $\boldsymbol{Int'_i}$ and the entities of analysis $\boldsymbol{e_j}$ is given by the matrix C[c_{ij}], $i = 1, \dots, n$; $j = 1, \dots, m$. If the enriched interaction $\boldsymbol{Int'_i}$, contains the entities of analysis $\boldsymbol{e_j}$, then $C_{ij} = 1$, otherwise $C_{ij} = 0$ (Figure 3).

For example, a collaborative and distributed design is conducted within the GRACC[1]. Four research laboratories (CRAN-Nancy, IRCCyN-Nantes, M3M-Belfort, and 3S-Grenoble) have participated in this experience.

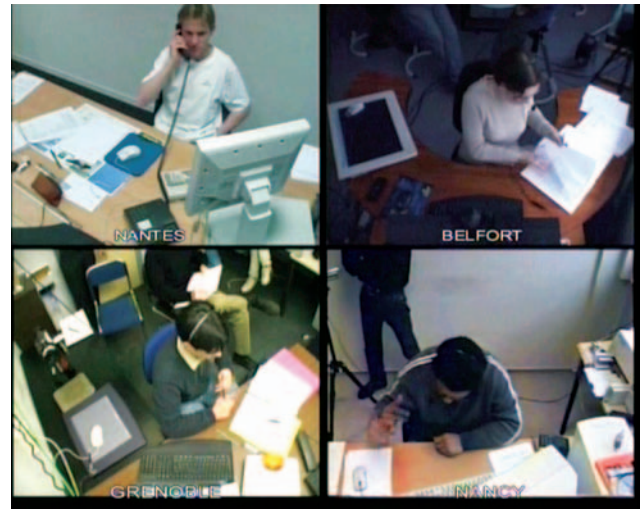

Figure 5. Collaborative and distributed design experience of GRACC.

The problem is the design of a children's trailer (Figure 4). A group of work with four actors was constituted with a project manager, a form designer, a frame designer, and a link designer. The design experiences that have been carried out are mainly remote design experience which implement computer means and software of the market (CAD, MS-Office, video conference, sharing of applications...) (Figure 5). There were four meetings overall. The total duration of each synchronous meeting is 2 h and their corresponding corpus contains ≈1000 interventions.

The first synchronous meeting of the experience of collaborative design (GRACC) was analyzed. For this design experience, the matrix C[cij] (Figure 6), corresponding to an extract of the corpus, represents the relation between the enriched interactions $\boldsymbol{Int'_i}$ and the entities of analysis $\boldsymbol{e_j}$.

2.2 Identification of micro state-problems

In the case when a family of enriched interactions corresponds to a family of entities of analysis, the matrix C[c_{ij}], $i = 1, \dots, n$; $j = 1, \dots, m$ takes a particular form (Figure 7). Here, the partition of the set of enriched interactions and the partition of the set of entities of analysis are carried out so that each part of the enriched interactions corresponds to a part of the entities of analysis. Ideally, the search of these correspondences allows the decomposition of the matrix C[c_{ij}], in sub-matrix C(Diagonal) and C(Out-Diagonal) where sub–matrices C(Diagonal) are filled only with '1', and sub-matrix C(Out-Diagonal) are filled with '0'. In practice, it is not always the case. During the interaction, it is necessary to have confrontation and the negotiation to arrive finally at a consensus. Mathematically, the search of the

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}	e_{17}	e_{18}	e_{19}	e_{20}	e_{21}	e_{22}	e_{23}	e_{24}	e_{25}	e_{26}	e_{27}	e_{28}	e_{29}	e_{30}	e_{31}	e_{32}	e_{33}	e_{34}	e_{35}	e_{36}	e_{37}	e_{38}	e_{39}	e_{40}	e_{41}	e_{42}	e_{43}	e_{44}	e_{45}	e_{46}	e_{47}	e_{48}	e_{49}	e_{50}	e_{51}	e_{52}
Int'$_{17}$		1					1																																													
Int'$_{19}$																						1	1															1							1							
Int'$_{23}$																																																			1	
Int'$_{26}$				1	1											1																																				
Int'$_{28}$													1														1					1																1				
Int'$_{30}$																																	1																				
Int'$_{31}$				1	1																																																
Int'$_{34}$			1			1																																															
Int'$_{35}$													1																																							
Int'$_{38}$		1			1																																																
Int'$_{42}$			1																																																		
Int'$_{44}$						1										1																																					
Int'$_{45}$												1																																								
Int'$_{49}$			1																																																		
Int'$_{50}$												1		1																																							
Int'$_{57}$																																													1								
Int'$_{66}$		1					1																																														
Int'$_{67}$				1					1																																												
Int'$_{69}$									1																																												
Int'$_{74}$																					1																																
Int'$_{76}$																																																					1
Int'$_{80}$																											1																										
Int'$_{83}$																				1										1							1																
Int'$_{86}$	1																																																				
Int'$_{87}$	1																																																				
Int'$_{90}$	1																																																				
Int'$_{91}$	1																																																				
Int'$_{93}$								1		1	1			1	1		1	1						1								1				1		1		1									1			
Int'$_{95}$		1								1	1				1		1	1																								1	1				1						
Int'$_{97}$								1											1						1										1										1					1		
Int'$_{98}$																										1																1										
Int'$_{100}$																													1																								
Int'$_{101}$																															1																						

Figure 6. Matrix C[c_{ij}].

	e_2	e_7	e_4	e_5	e_{13}	e_{27}	e_{33}	e_{48}	e_{12}	e_9	e_1	e_3
Int' $_{1}$	1	1										
Int' $_{10}$	1			1								
Int' $_{17}$	1	1										
Int' $_{4}$			1	1								
Int' $_{7}$			1	1								
Int' $_{5}$					1	1	1	1				
Int' $_{13}$									1			
Int' $_{15}$									1			
Int' $_{18}$			1							1		
Int' $_{19}$										1		
Int' $_{24}$											1	
Int' $_{25}$											1	
Int' $_{26}$											1	
Int' $_{27}$											1	
Int' $_{8}$												1
Int' $_{11}$												1
Int' $_{14}$												1

Figure 7. Partition of matrix C[c_{ij}].

1	e_2	faire,.V
	e_7	bilan,.N
2	e_4	mettre,.V
	e_5	partage,.N
3	e_{13}	orga,.N
	e_{27}	document,.N
	e_{33}	envoyer,.V
	e_{48}	s'appeler,.V
4	e_{12}	numéro,.N
5	e_9	écran,.N
6	e_1	entendre,.V
7	e_3	connecter,.V

Figure 8. Entities of analysis families.

families of enriched interactions and the families of entities of analysis is a search problem of simultaneous partitions in the two sets, the enriched interactions and the entities of analysis, in correspondences or quasi-correspondences, class of partition to class of partition. Then, the basic idea consists in carrying out permutations of lines and columns of the matrix C[c_{ij}] such as to find the structure of the correspondence on the crossing of these two sets. This clustering technique is called block-seriation [12].

The structure of the matrix C[c_{ij}] makes it possible to identify the families of the entities of analysis (Figure 8).

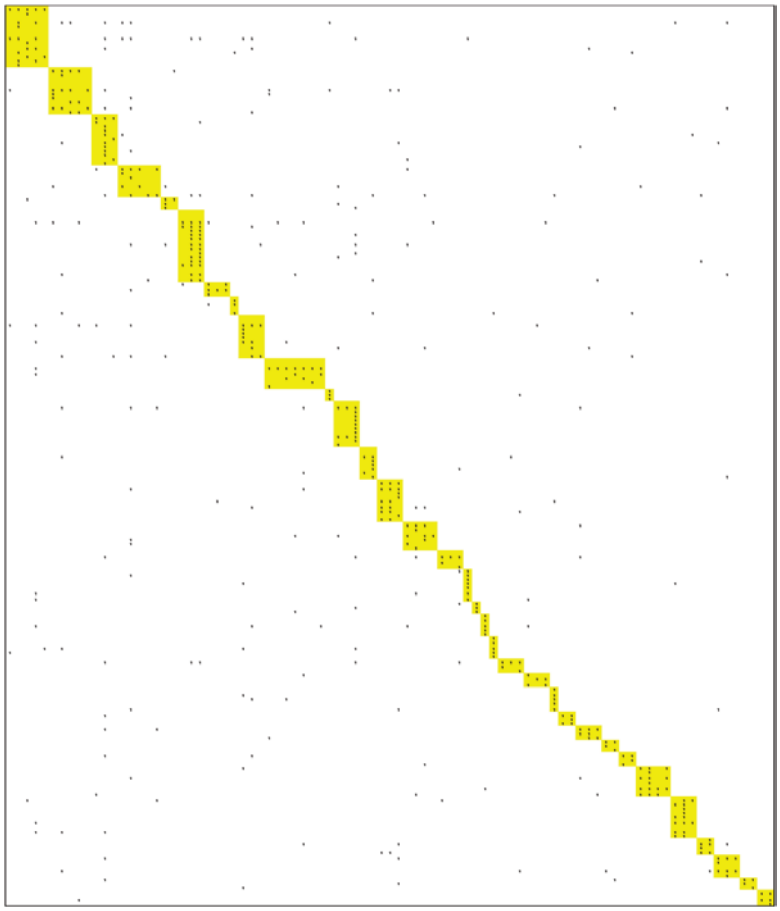

Figure 9. Partition of matrix C[c_{ij}] for a meeting.

The sequence of these entities of analysis in a family is called a concept. Emergence here means that a concept, as a group of entities of analysis, was not previously represented, but can be represented because it has been constructed now [9]. Moreover, the correspondences per block permit the characterization of a family of interventions by the corresponding concept(s). Each concept offers a fair idea to a state of the design process and this related problem, which is called micro state-problem. Thus, a family of enriched interactions allows identification of the state-problem of the collaborative and distributed design process.

For example, Figure 9 shows the partition of enriched interactions-entities of analysis matrix C[c_{ij}] for the considered design experience. It shows the identification of the 33 families of entities of analysis corresponding to 33 state-problems. The recognition of the state-problems in the corpus is carried out with the help of the software '*Intex*[2]' (Figure 10).

2.3 Dynamics of micro state-problems

The relationship between the interventions and state-problems permit to note their continued evolution. For instance, Figure 11 represents the evolution of state-problems found in the partition of matrix C[c_{ij}] (Figure 7). Then, the design process, as a dynamics system, is represented like a change of qualitative states. A state can be considered as attractor in a dynamics system. It is represented then, as a state which drew up the other neighboring states.

The representation of the state-problems in time allows to note the dynamics of the problems in relation to the design process organization, as well as in relation to the evolution and/or the emergence of the solutions. For example, the dynamics of 33 state-problems (Figure 9) is represented in Figure 12 for a synchronous meeting of 500 interventions (2 h). The analysis of these state-problems shows that these dynamics are characterized rather by many irregular leaps. These leaps, probably unforeseeable, show that the design process, on a micro scale, is far from being harmonious. In fact, these leaps depend strongly on the structural causes, such as the auto-organization (or self-organizing) inside of the team [18], the human action, such as the creative characteristic of the design [13,14].

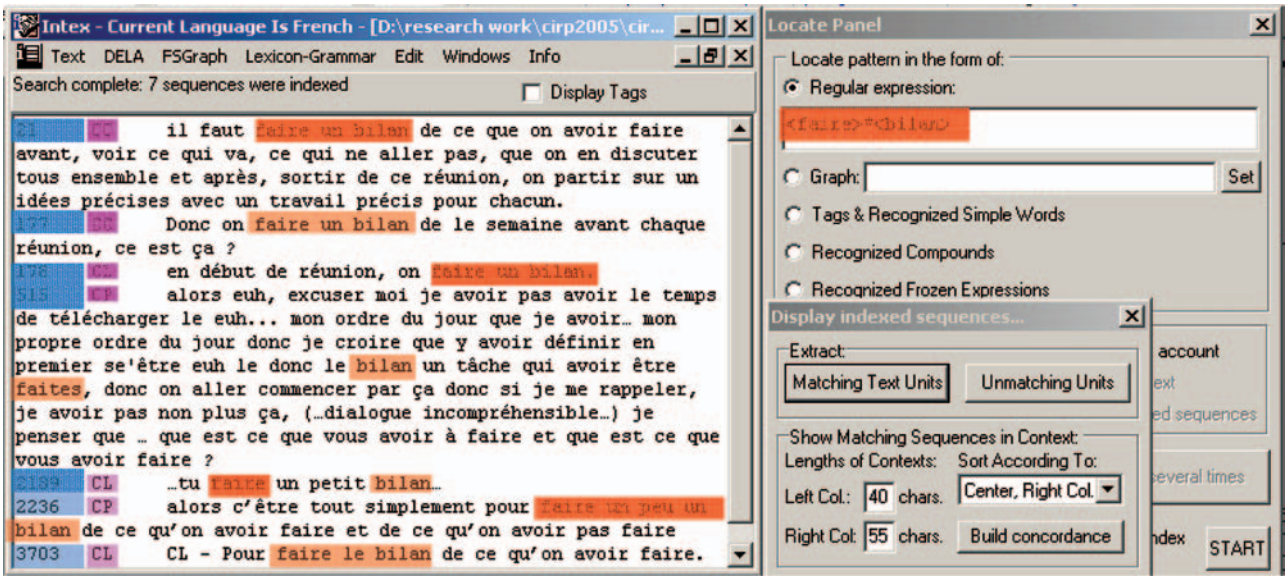

Figure 10. Researching of entities of analysis of the first family.

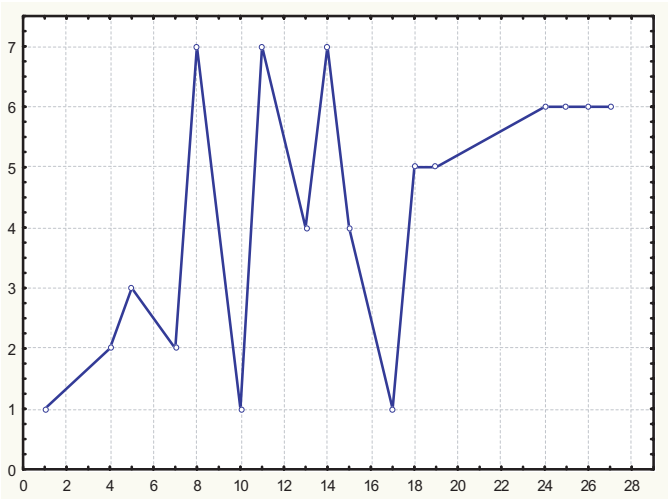

Figure 11. Dynamics of state-problems for the extract of 27 interventions.

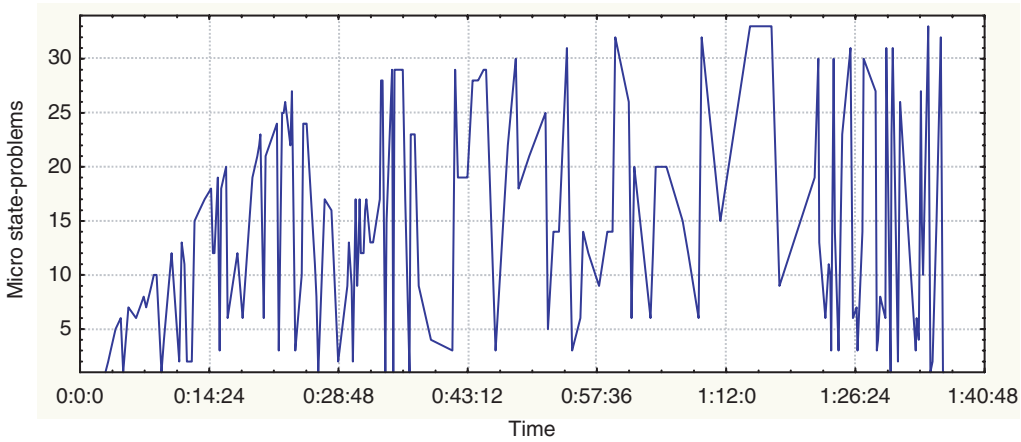

Figure 12. Dynamics of micro state-problem in a synchronous meeting.

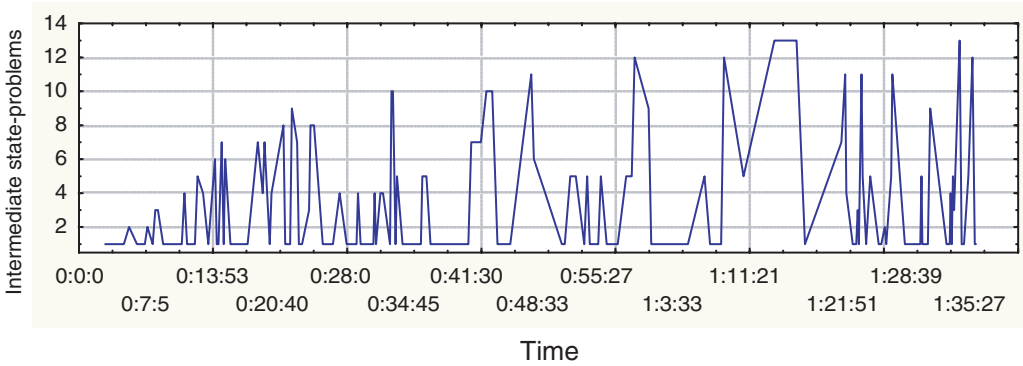

Figure 13. First aggregation of state-problems.

3. Discovering macro state-problems

The state-problems are semantically interpretable. Consequently, these state-problems corresponding to the design activities can be combined in hierarchical aggregation form. Thus, the state-problems of inferior levels are included in the state-problems of superior levels, until this recursive process can reach the last level, called macro state-problem. Then, the discovery of the macro state-problems is based on the aggregation principles. The goal of the recursive aggregation of the (micro and/or intermediate) state-problems is to carry out a hierarchical representation of families of the state-problems and families of the associated entities of analysis. The result of the hierarchical aggregation enables the concerned agent to understand the causality of variation of the discussions. Indeed, there is a relationship between a series of intermediate state-problems and the changing of the discussions.

The partition of matrix $C[c_{ij}]$ is considered here for example (Figure 9). Using the technique presented in [15], the first aggregation, based on the relationship of the sub-matrices C(Diagonal), gives as a result 13 intermediate problems (Figure 13). Following up the process of aggregation, the number of state-problems is reduced to six (Figure 14). This is the last possible aggregation. Therefore corresponding states are the macro state-problems. The dynamics of the state-problems is shown in Figure 15.

The dynamics of the state-problems implies that a hierarchical network of the concepts related to a state-problem can be created. Consequently, this dynamics of the state-problems enables one, on the one hand, to identify the borders of the discussions, and on the other hand, to understand the semantics of these discussions. Here, the discussions are strongly dependent on the tackled state-problems.

4. Conclusions

In this study, a framework is proposed for understanding the dynamics of state-problems in the collaborative and distributed design. This approach consists in distinguishing, from the real interactions and the emergence of the concepts, the different state-problems and their dynamics characterizing the collaborative and distributed design process.

The analysis of the state-problems in a real experience of the collaborative and distributed design shows that their dynamics is characterized rather by many irregular leaps. These leaps, probably unforeseeable, show that the design process on a micro scale is far from being harmonious. Indeed, these leaps depend strongly on structural causes, such as the auto-organization inside the team, as well as the human action, such as the creative characteristic of the design process.

The dynamics of the state-problems permit one to identify the approximate boundaries of the discussions and to decompose a corpus of the collaborative and distributed design process.

Based on the proposed approach, the authors are currently working on the development of a multi-agents system for discovering state-problems in the collaborative and distributed design process. In addition to developing a multi-agents system for discovering state-problems, the experiences with real industrial problems is considering to see the application of the proposed approach.

Further research is required to develop domain ontology for improving the ability to recognize and identify relevant state-problems. Modeling and analysis of collaborative and distributed design processes based on the theoretical background of dynamic systems seems to be a relevant vector to explain phenomena, such as auto-organization and attractor states.

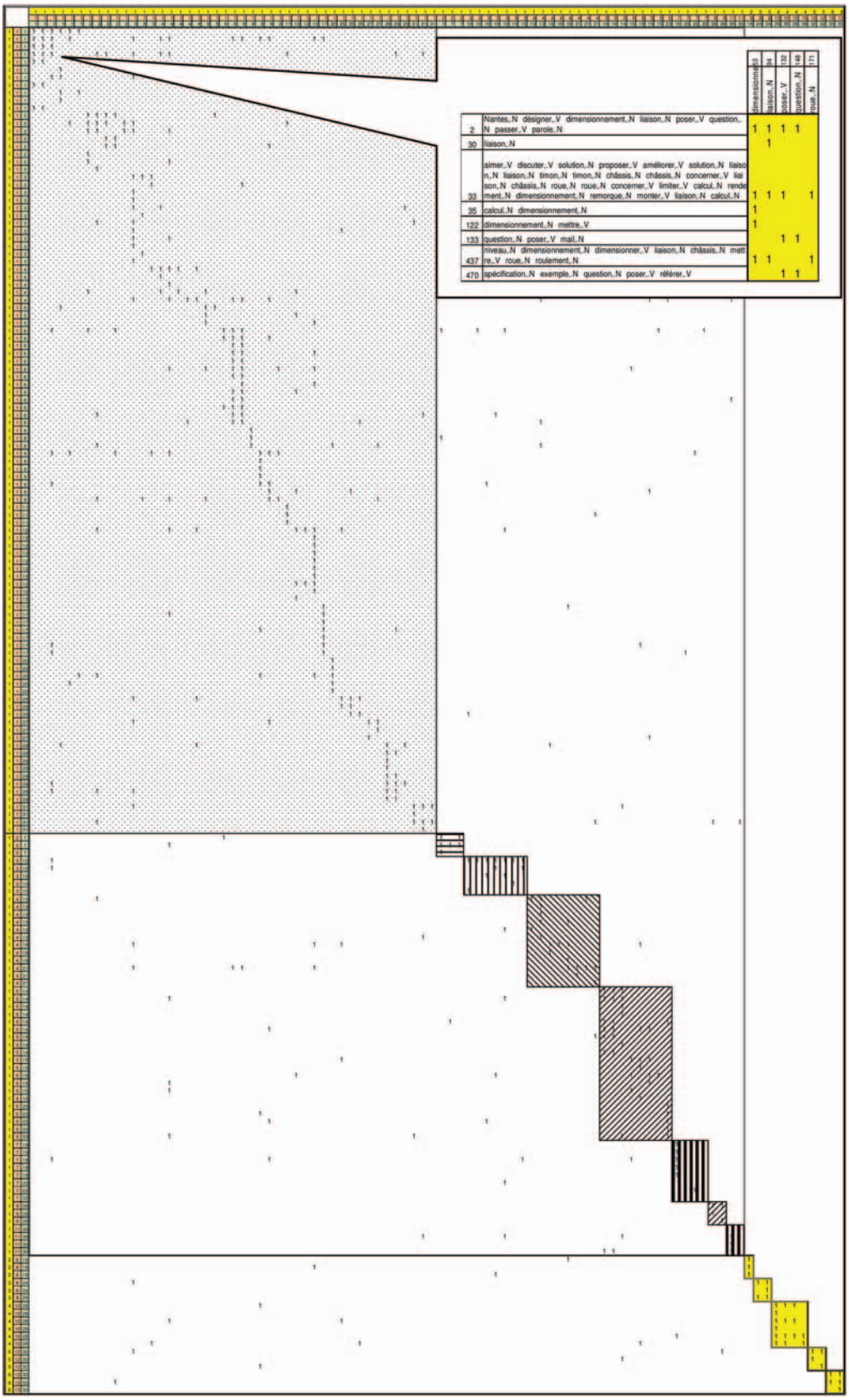

Figure 14. Last aggregated matrix with three aggregations.

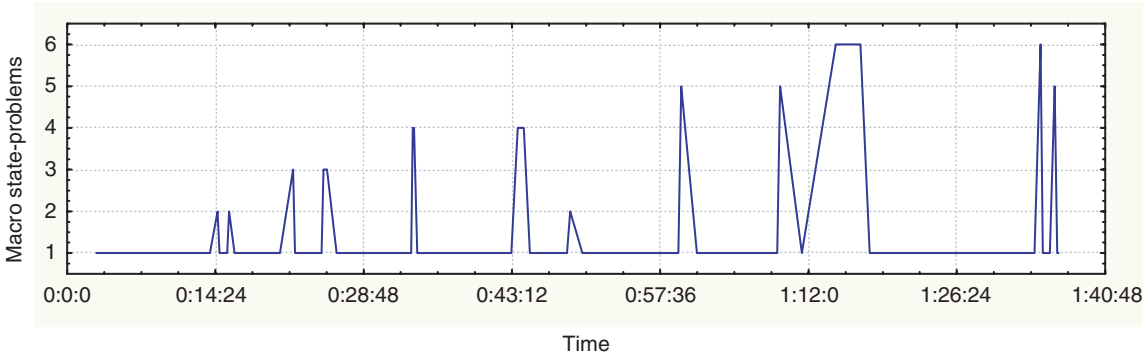

Figure 15. Dynamics of macro state-problems.

References

1. Akman, V., ten Hagen, P.J.W. and Tomiyama, T. (1994). Desirable Functionalities of Intelligent CAD Systems, In: Dagli, C.H. and Kusiak, A. (eds), *Intelligent Systems in Design and Manufacturing*, p. 119, ASME Press, New York.
2. Baya, V. and Leifer, L. (1995). Understanding Design Information Handling Behavior using Time and Information Measure, In: Ward, A.C. (ed.), *Proceedings of the 1995 Design Engineering Technical Conferences*, ASME DE-83, Vol. 2, pp. 555–562.
3. Chiu, M. (2002). An Organizational View of Design Communication in Design Collaboration, *Design Studies*, **23**(2): 187–210.
4. Coiera, E. (2002). Interaction Design Theory, *International Journal of Medical Information*, 1–18.
5. Dong, A., Hill, A. and Agogino, A.M. (2004). A Document Analysis Method for Characterizing Design Team Performance, *Journal of Mechanical Design, Transactions of the ASME*, **126**(3): 378–385.
6. Dong, A. (2004). Quantifying Coherent Thinking in Design: A Computational Linguistics Approach, *Design Computing and Cognition'04*, pp. 521–540, MIT, USA.
7. Dong, A. and Agogino, A.M. (1997). Text Analysis for Constructing Design Representations, *Artificial Intelligence in Engineering*, **11**(2): 65–75.
8. Garro, O., Salau, I. and Martin, P. (1995). Distributed Design Theory and Methodology, *International Journal of Concurrent Engineering: Research and Application* (*CERA*), **3**(3): 43–54.
9. Gero, J. (1998). Concept Formation in Design, *Knowledge-based Systems*, **11**: 429–435.
10. Hill, A., Song, S., Dong, A. and Agogino, A.M. (2001). Identifying Shared Understanding in Design using Document Analysis, In: *Proceedings of the ASME Design Engineering Technical Conference*, Vol. 4, pp. 309–315.
11. Lang, S., Dickinson, J. and Buchal, R.O. (2002). Cognitive Factors in Distributed Design, *Computers in Industry*, (48): 89–98.
12. Marcotorchino, F. (1987). A Unified Approach of the Block-Seriation Problems, *Journal of Applied Stochastic Models and Data Analysis*, **3**(2): 1–13.
13. Movahed-Khah, R., Ostrosi, E. and Garro, O. (2004). An Approach Based on the Communicative Traces for the Collaborative Design Process Analysis, *Scenario-Based Design of Cooperative Systems, COOP'04* (*6th International Conference on the Design of Cooperative System*), pp. 31–39, May 11–14, Hyères, France.
14. Movahed-Khah, R., Ostrosi, E. and Garro, O. (2005). A Cluster Based Approach for Collaborative Design Process Analysis, In: ElMaraghy, H.A. and Elmaraghy, W.H. (eds), *Advances in Design*, Springer-Verlag, UK, ISBN: 1846280044.
15. Ostrosi, E., Ferney, M. and Garro, O. (2003). A Fractal Approach for Concurrent Engineering, *International Journal of Concurrent Engineering: Research and Application* (*CERA*), **11**(4): 249–265.
16. Ostergaard, D. and Summers, D. (2003). A Taxonomic Classification of Collaborative Design, *ICED03*, Stockholm, August 19–21.
17. Ostrosi, E. and Ferney, M. (2004). Collaborative and Distributed Design Process Analysis for Intelligent Agents' Implementation, In: *Proceedings of the 11th ISPE International Conference on Concurrent Engineering* (*CE2004*), 26–30 July 2004, Beijing, P.R. China, *Concurrent Engineering – The Worldwide Engineering Grid*, Sobolewski, M. and Cha, J. (eds), ISBN-7-302-08802-0, pp. 215–222, Tsinghua University Press and Springer-Verlag.
18. Stempfle, J. and Badke-Schaub, P. (2002). Thinking in Design Teams- An Analysis of Team Communication, *Design Studies*, **23**: 473–496.