Methodological utilities for organizing cooperative work process according to ISO 9001 standard recommendations

Michael David, Zahra Idelmerfaa

HAL Id: hal-00112094
https://hal.science/hal-00112094

Submitted on 7 Nov 2006

Methodological utilities for organizing cooperative work process according to ISO 9001 standard recommendations

Michaël DAVID and Zahra IDELMERFAA

Centre de Recherche en Automatique de Nancy – CRAN CNRS UMR 7039
Université Henri Poincaré, Nancy I
BP 239, 54506 Vandoeuvre Lès Nancy cedex, France
michael.david@cran.uhp-nancy.fr
Phone: (33) 03 83 68 44 41 Fax: (33) 03 83 68 44 37

Abstract: Research for improving the product development process becomes crucial in new work organizations. Today, projects involve more and more activities and work groups which must be efficiently coordinated. In this paper, a method for structuring cooperative activities in order to improve its global performance is proposed. This global and methodological approach can be used to specify collaborative frameworks like Web technology environments. The study is based on the recommendations of ISO 9001 standard. Classical methods decompose a project into several work groups which are then scheduled. This first decomposition level is supplemented with a more detailed structuring level in order to control the size of the working teams. The steps of the process being defined and planned, the responsibilities inside the working groups are specified. Finaly, the method proposes to identify the couples of actors who generate the best interface for the effectiveness of the cooperative work.

Keywords: Concurrent Engineering, Project Management, Work Organization, Decomposition Method, Graph Theory.

1. INTRODUCTION

To improve the product development process (PDP), companies lean on cooperative work organisations, which enable collection of a great number of skills and thus of actors for the development project. But, these organisations require defining, planning and coordinating of the work groups and activities in the most effective possible way. Consequently, it is important to provide methods that enable to structure and to effectively control the PDP.

With the emergence of Concurrent Engineering, numerous research works and publications (Gebala and Eppinger, 1991; Kusiak, and al., 1994; Krishnan, and al., 1997) were developed for organization and scheduling of cooperative activities. The different approaches are based on tools to describe the design activities and their dependencies (Steward, 1981). They use partitioning methods to decompose a work organization into a set of work groups and to organize them according to their dependencies (Harary, 1962).

A refinement of the groups obtained by partitioning methods is proposed. The objective is to improve lead time and workload of these groups. The application of o spectral algorithm allows to identify independent task groups capable of working in parallel. This decompositoion allows to minimize reworking and repetition between tasks. The method proposes a set of decomposition criteria relative to the balancing of workload and a model for the estimation of development time and cost. The model allows to estimate and to compare various work organizations.

The method is derived from the analysis of a complex PDP in order to deduce its optimal organisation. For this, the recommendations of the ISO 9001 standard (ISO, 2000) are applied, which define different requirements to control and to plan the design and/or development of a product and/or service,:

- the steps of the PDP are defined and planned,
- the required activities for review, verification and validation are well identified,
- the responsibilities and authorities for the PDP are well identified,
- the interfaces between groups involved in the development process are managed.

For the analysis of the information flow between the actors of a PDP, different techniques based on the graph theory are proposed in order to help a quality manager to determine the fulfilment of the requirements. In section 2, a method for decomposing and planning the steps of the PDP is defined. The responsibilities of the obtained groups are specified in section 3. Section 4 describes the interfaces which must be managed to ensure both an efficient communication and to clarify the responsibilities.

2. DEFINING THE STEPS OF THE PDP

2.1 General Decomposition of the PDP

The first step of the proposed method aims at determining an initial organisation of the cooperative activities of a PDP. It enables to identify and schedule the activities such that they:

- can be executed only after they receive all the information required from their predecessors (Serial activities),

- do not depend on others tasks (Parallel activities),

- are interdependent and must be executed simultaneously (Coupled activities).

This step is based on the work relative to the management of complex systems (Steward, 1981). A product design project is modelled by means of a matrix called Design Structure Matrix (DSM).
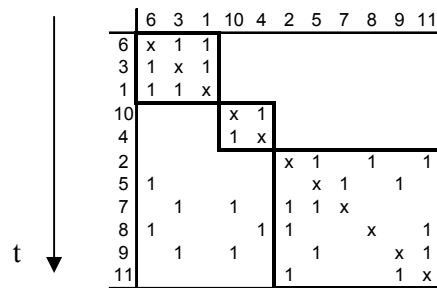

Fig. 1. Initial Design Structure Matrix. Represents the exchanges between eleven actors of a design project. The upper and lower triangle visualise unknown and known information, respectively.

In a DSM, an identically labelled row and column represent a design activity (Fig.1) has to be made clear in the text by defining and describing an example, which should be presented prior to the method and which should be used throughout the paper. The elements "1" within each row identify, which activities must contribute to information for the proper achievement of the PDP. The initial DSM depicted on figure 1 represents the exchanges between eleven actors of a process. The upper triangle visualises unknown information and the lower, known information. Thus, elements "1" in row a_3 indicate that the actor a_3 requires information produced by the actors a_1 and a_6. The information

to be transferred from the actor a_1 is known because he has finished his work. But the information transferred from the actor a_6 is unknown and must be estimated by the actor a_3 because a_6 has not yet started his activity. The order of the rows (or columns) indicates the chronology of activities, i.e. the sequence to start design activities. The matrix is initially not structured (unspecified chronology) and doesn't show any visible connected activity.

The objective is to find a sequence of activities that enable the matrix to become lower triangle. The partitioning process consists in rearranging the initial matrix by interchanging rows and swapping the corresponding columns in order to achieve a more organised work sequence i.e. to enable the different actors to work on valid information. The method aims at identifying the strongly related components inside the matrix (Harary, 1962). The actors are then collected within a coupled task, i.e. a work group where the actors must closely cooperate. In the example, three strongly related components constituted of the actors $\{a_1, a_3, a_6\}$, $\{a_4, a_{10}\}$, and $\{a_2, a_5, a_7, a_8, a_9, a_{11}\}$ are identified.

```
         6  3  1  10 4  2  5  7  8  9  11
    6    x  1  1
    3    1  x  1
    1    1  1  x
   10             x  1
    4             1  x
    2                   x  1     1     1
    5    1                x  1      1
    7       1        1    1  1  x
    8    1              1 1        x     1
    9       1        1       1        x  1
   11                      1              1  x
```
t →

Fig. 2. Partitioned DSM. Three strongly related components called Coupled Tasks.

They are gathered in the matrix in three coupled tasks which are then scheduled in order to remove the "1" on the upper triangle of the matrix and to obtain a partial order (Fig. 2). The two-coupled tasks implying the actors $\{a_1, a_3, a_6\}$ and the actors $\{a_4, a_{10}\}$ can be executed in parallel because they do not have interaction. Then, as soon as these two tasks are performed, the third coupled task implying the actors $\{a_2, a_5, a_7, a_8, a_9, a_{11}\}$ will be able to work since it will have all information it requires, transferred from the first two tasks.

The major interest to define a work organization by partitioning a DSM is to decompose and to schedule the design project. However, the method does not enable to control the size of the groups, i.e. the number of actors within a team. Thus, the application of this method to a complex design

project can lead to the constitution of groups with various sizes. In the next level, the interest is related to large groups.

2.2 Detailed Decomposition of the PDP

The second step is concerned with the reduction of the number of participants inside groups identified in the first decomposition level. Indeed, collaboration in the same work session with to many people can slow down the work performance of this group significantly (Zhao and Liu, 2003).

2.2.1 Performance Estimation of an Organization

2.2.1.1 Work Transformation Matrix

The Work Transformation Matrix (WTM) model is used to estimate time and workload of a coupled task and so global duration and cost of a design project (Smith and Eppinger, 1997). It is an extension of the DSM, which integrates an assessment of activities in term of estimated time and rework rate (see Fig 3). The diagonal elements represent the time to complete each activity during the first iteration. The off-diagonal elements represent the strengths of dependence between activities, giving rise to the transfer of work, or rework, involved in the iterations. It is allowed that each activity creates a deterministic amount of rework for other activities.

Time matrix W: Diagonal elements mean that the activity 10 has an estimated initial time of 4 units and the activity 4 lasts 7 units of time.

Rework matrix A: The element ".2" on the first line means that 20 % of the activity 10 must be reworked after the realization of the activity 4. Finally the element ".4" on the second line means that 40 % of the activity 4 must be resumed with the results of the activity 10.
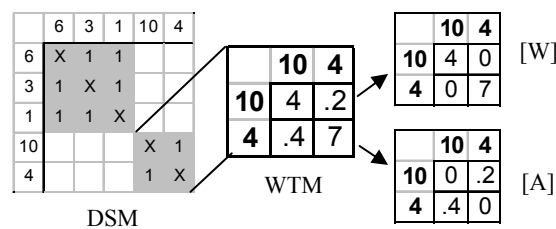

Fig. 3. WTM model (applied on the second Coupled Task of the example)

2.2.1.2 <u>Identifying performance indicators</u>

The working vector u_t represents the remaining work for every activity to the tth iteration. At first, it remains the totality of the work to be executed for each activity, so the initial vector is $u_0 = (1,1,\ldots,1)^n$ with n, number of activities. After iteration, the working vector is multiplied by the rework matrix:

$$u_{t+1} = A.u_t \text{ or } u_t = A^t.u_0$$

A total working vector can also be defined:

$$U = (\sum_{t=0}^{\infty} A^t).u_0 \qquad (1)$$

R is the vector of work times representing the workload of each activity by unit of time (for example men per week) and is expressed by:

$$R = WU = W.(\sum_{t=0}^{\infty} A^t).u_0 \qquad (2)$$

R integrates the time factor. This expression can be reduced. Indeed $\lim_{t \to \infty}(M^t) = (I-M)^{-1}$ if the maximum eigenvalue of a matrix M is lower than 1 (I is the unit matrix). This condition is always true for A, so: $\qquad\qquad\qquad R = W(I-A)^{-1}.u_0 \qquad (3)$

E is the total workload representing the cost of the coupled tasks set. E is the sum of all elements of R:

$$E = \sum_{i=0}^{n} R^{(i)} \qquad (4)$$

n represents the number of activities in the project and the notation $V^{(i)}$ means the ith element of V.

T is duration of a coupled task. T is obtained in adding the duration of the longest activity for each iteration: $\qquad\qquad\qquad T = \sum_{t=0}^{\infty} \max[W.u_t]^{(i)} \qquad (5)$

We apply the equations (3), (4), (5) to estimate the workload and the lead time of the second coupled task or TC2 (Fig. 4): At the first iteration, the execution of activities 10 and 4 start. So $u_0 = [1; 1]$

$R_2 = W . (I\text{-}A)^{-1} . u_0 = [5.22 ; 10.65]$

$E_2 = R^{(10)}+R^{(4)} = 15.87$ weeks

$T_2 = t(\text{iter. }1) + t(\text{iter. }2) + \ldots + t(\text{iter. }x) = \max(W.u_0) + \max(W.u_1) +\ldots+ \max(W.u_{x-1})$

$T_2 = 7 + 2.8 + 0.56 + 0.22 + 0.05 + 0.02 + \ldots = 10.65$ weeks

We note that the critical time delay for CT2 comes from the activity 10 during all iterations. Same manner we calculate the durations and workload for the two other coupled tasks with the WTM model of the global process (Fig. 4). We collected the results in Table 1.

Fig. 4. WTM model of the example

Criteria (week)	CT 1	CT 2	CT 3
Workload E	36.54	15.87	70.48
Lead Time T	15.85	10.65	18.73

Table 1. Estimated workload and lead time of the three coupled tasks

The WTM model allows estimating the performance of a cooperative work. The optimization criteria only depend on working vector R. Indeed R allows to obtain cost E and duration T for a cooperative process. Then specific models will be introduced for the coupled task decomposition, i.e. the realization in series or in parallel for a set of workgroups.

2.2.2 The Decomposition of a Coupled Task

2.2.2.1 Cooperation graph

The objective is to reduce the complexity of a work task by separating it in sub-tasks. The goal is to find a compromise for structuring and to make collaborating these sub-tasks. For this, the concept of cooperation graphs (CG) is used to study the information flow between the actors and then to deduce the positions of each one (Fig. 5). Edge from agent Aj to Ai means that Aj needs information transfer from Ai. Agent Ai cooperates with agent Aj if Ai gives or shares some of its information with Aj.

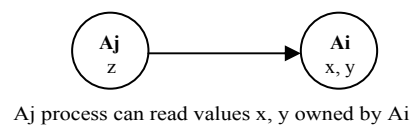Aj process can read values x, y owned by Ai

Fig. 5. Cooperation graph.

The concept of CG has been defined by (Diaz, and al., 1996) who applied it to specify network services and protocols for managing agents in team work. The approach proposes to apply it to cooperative PDP. A cooperative group is composed of agents who are organised according to the relationships between the actors. Vertices represent the agents; edges represent the relations between them. CG provides a formal representation that enables to analyse the kind of cooperation between the actors according to their relationships (Rondeau, and al., 1999):

- if a relation is symmetrical then the cooperation is bilateral,

- if a relation is transitive then the cooperation is hierarchical,

- if a relation is both symmetrical and transitive then the cooperation is total.

2.2.2.2 Model for the Sequential Decomposition of a Coupled Task

We introduce specific models for the decomposition of a group, i.e. the realization in series and in parallel for a set of workgroups. We choose to arbitrarily study a coupled task containing 4 strongly connected activities. By identification, we define a model for the sequential execution of a group after a bi-partitioning, i.e. a dichotomy of the set of activities.

On the example of the figure 6, the sequential decomposition of the coupled task initializes a first subgroup composed by activities 1 and 2, then in a second phase, a second subgroup (activities 3 and 4). The first phase of the sequential decomposition consisted in activities whose go to their term. Then in the second phase, we initialize remaining activities.

$$R = W(I - A)^{-1}.u_0$$
Concurrent Work

Phase 1

$$K \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

$$R_1 = W(I - KAK)^{-1}.Ku_0$$

Phase 2

$$R_2 = W(I - A)^{-1}(I - K).u_0$$
Work in series
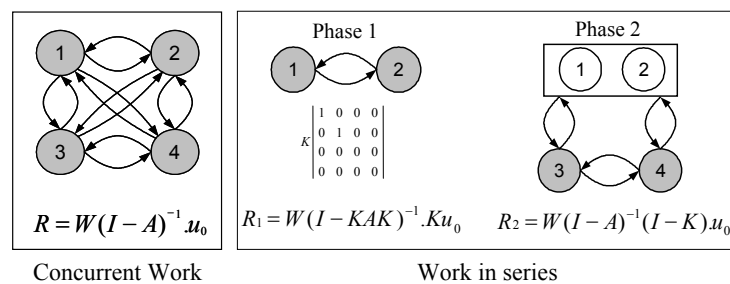
Fig. 6. Sequential decomposition of a coupled task

According to their results, we resume the first activities with the preliminary hypothesis that there are no variations of parameters between two iterations. For the calculation of the responsibility R, a factor K, called cutting matrix, is introduced. It allows to take into account only activities executed during the first phase (R_1). I represents the unit matrix. During the second phase, there is initial work to be completed only on the activities that were not done in the first phase. Iterative rework may need to be done on any activity, whether it is a first or second phase task (R_2).

Model for the sequential decomposition is described by the following equations:

<div style="border:1px solid #000; background:#ccc; padding:10px;">

Model for the sequential decomposition

In the first phase:
$$R_1 = W(I - KAK)^{-1}K.u_0$$

In the second phase:
$$R_2 = W(I - A)^{-1}(I - K).u_0$$

Total Workload:
$$E = \sum_{i=0}^{n}\left(R_1^{(i)} + R_2^{(i)}\right)$$

Total lead-time:
$$T = \sum_{t=0}^{\infty}\max[W.KA^tK u_0]^{(i)} + \sum_{t=0}^{\infty}\max[W.A^t(I - K)u_0]^{(i)}$$

</div>

This method allows to estimate the performance of a sequential decomposition. However we notice two problems: on the one hand, there is no successful method to find an optimal decomposition in the series (we can only simulate exhaustively the various organizations). On the other hand, a complete serial decomposition does not optimize the development time. It can drive to a repetition phase (phase 2) as long and complex as the process before its decomposition. In the next step, the objective is to define an organization, which proposes an overlapping of the subgroups.

2.2.2.3 <u>Model for the Parallel Decomposition of a Coupled Task</u>

By identification with equation (3), a model for the parallel execution of a coupled task after a bi-partitioning is defined, i.e. a dichotomy of the set of activities (Fig. 7). Phase 1 consists in the independent ly work of two subgroups. The factors K_1 and K_2 allow to define the composition of each subgroup and are additional.
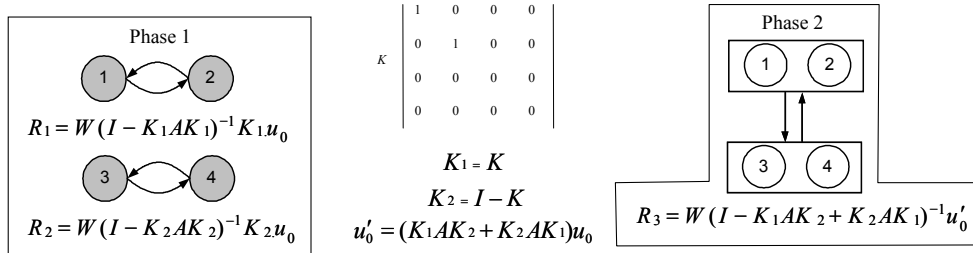

Fig. 7. Decomposition and realization in parallel of a coupled task.

The term u'_0 defines the information flows to be taken into account in the second phase, i.e. only inter-group exchanges. The total lead-time T is the sum of the delay's phases. So T= max $[t(R_1); t(R_2)] + t(R_3)$.

The parameters of the WTM determine the values of vectors R corresponding in a parallel decomposition. Phase 2 is called re-connexion phase. The duration of this stage is going to highly influence the total delay. Instinctively, this approach has to bring a decrease of the total duration of cooperative work because all activities are initialized from the first phase. In the second one, it has to make the working subgroups collaborate efficiently together.

<div style="border:1px solid black; padding:1em;">

Model for the parallel decomposition

In the first phase:
$$R_1 = W(I - K_1 A K_1)^{-1} K_1 . u_0$$
$$R_2 = W(I - K_2 A K_2)^{-1} K_2 . u_0$$

In the second phase:
$$R_3 = W(I - K_3)^{-1} K_3 . u'_0$$

$$E = \sum_{i=0}^{n} \left(R_1^{(i)} + R_2^{(i)} + R_3^{(i)} \right)$$

$$T = \max\left(\sum_{t=0}^{\infty} \max[W.K_1 A^t K_1 u_0]^{(i)} ; \sum_{t=0}^{\infty} \max[W.K_2 A^t K_2 u_0]^{(i)} \right) + \sum_{t=0}^{\infty} \max[W.K_3 A^t K_3 u_0]^{(i)}$$

</div>

This model enables the comparison of different choices in the project organization. The approach gives the means to estimate in a quantitative and relevant way the quality of the organization of PDP during decomposition as far as the border line is choose. So it remains to define the way of decomposing coupled tasks.

2.2.2.4 <u>Separation of the Working Groups</u>

In order to minimize reviews and work repetitions in the second phase of the parallel decomposition, the identified subgroups have to be relatively independent (determination of the cutting matrix K). That is the first criterion for a partitioning, which will be applied. The second criterion of decomposition is to balance the workload of each group. The analysis of these criteria conforms to the problems defined in the researches relative to the graph partitioning. The general problem of the group decomposition can be expressed as follow.

Given is a graph $G = (V, E)$ with V the set of nodes, E the set of edges and $|V| = n$. Decomposition or partitioning of this graph corresponds to the decomposition of the set of nodes V in k sub-groups V_1, $V_2...,V_k$ such as:

- $$\sum_{i=1}^{k} V_i = V \quad and \quad V_i \cap V_j = \phi \quad with \ i \neq j$$

- The number of edges connecting its groups is minimal (minimizing inter-group streams)

- The weight of the nodes of every group is appreciably equal (balancing workload)

To solve this problem, several types of algorithms were developed such as spectral algorithms (Pothen et al., 1990), multi-levels (Karypis and Kumar, 1998) or genetic (Todd and Sen, 1999). A spectral algorithm is applied in this context. The interest of this algorithm is to be able to arrange nodes on a scale characterizing the force of dependencies of nodes couples. So the first decomposition criterion is satisfied. Then it remains to choose at which place the cut will be applied. If all nodes have identical weights, the median shall be taken as caesura to respect the criterion of balancing workload. The spectral algorithm allows to obtain the cutting matrix K, which gives the optimal decomposition of a coupled task (David et al., 2004).

The spectral algorithm uses the following procedure to partition a graph into two subsets:

- **Step 1**: Building the Laplacian matrix L of the graph, $L = D - A$

  where A is the matrix such as: $A=[a_{ij}]$ and,

  $$a_{ij} = \begin{bmatrix} w\ (v_i,\ v_j) & if\ (v_i,\ v_j) \in E_m \\ 0 & otherwise \end{bmatrix}$$

  (w represents the weight of the edges) and D is a diagonal matrix such as $D =[d_{ij}]$ and

  $$d_{ij} = \begin{bmatrix} \sum w\ (v_i,\ v_j) & if\ i = j \\ 0 & otherwise \end{bmatrix}$$

- **Step 2**: Calculating the eigenvalues of the Laplacian matrix L;

- **Step 3**: Identifying the second smaller eigenvalue λ_2 and finding the eigenvector y corresponding to this eigenvalue;

- **Step 4**: Calculating the median M for the values yi;

- **Step 5**: Dividing the set of nodes V into two subsets P_1 and P_2 with the following criteria:

  if $yi \geq M$ then $V_i \in P_1$ and if $yi < M$ then $V_i \in P_2$.

For example, the third coupled task (CT3) obtained after the preliminary partitioning phase (Fig. 2) is considered. The spectral algorithm is applied to the set of activities: {a2, a5, a7, a8, a9, a11}. The first step consists in building the laplacian matrix. For that purpose, the cooperation graph of CT3 is

symmetrized (Fig. 8). Note that we used here the binary parameters (activities dependencies) instead of the review rates w of WTM. Nevertheless the resulting partition is the same one.
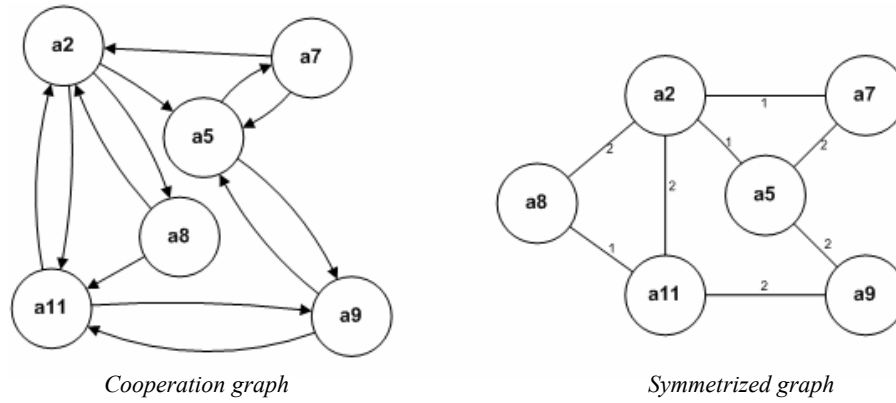

| Cooperation graph | Symmetrized graph |

Fig. 8. Cooperation graph symmetrization

Laplacian matrix of the symmetrized graph is:
$$L = \begin{bmatrix} 6 & -1 & -1 & -2 & 0 & -2 \\ -1 & 5 & -2 & 0 & -2 & 0 \\ -1 & -2 & 3 & 0 & 0 & 0 \\ -2 & 0 & 0 & 3 & 0 & -1 \\ 0 & -2 & 0 & 0 & 4 & -2 \\ -2 & 0 & 0 & -1 & -2 & 5 \end{bmatrix}$$

Its eigenvalues λ_i are:

λ_2	λ_5	λ_7	λ_8	λ_9	λ_{11}
-0.000	1.9619	2.8985	5.5910	6.8767	8.6718

The second smaller eigenvalue of the laplacian matrix L is λ_5. To define the sub-groups, we use the corresponding eigenvector, which is:

	a2	a5	a7	a8	a9	a11
ev5	0.2039	-0.3935	-0.5617	0.6364	-0.1380	0.2529

The activities are grouped according to values of the components of the eigenvector 5: the 3 largest values in a group and 3 smaller in the other. So the application of the spectral algorithm enables to decompose this group into two sub-groups: {a2, a8, a11} and {a5, a7, a9} (Fig. 9).
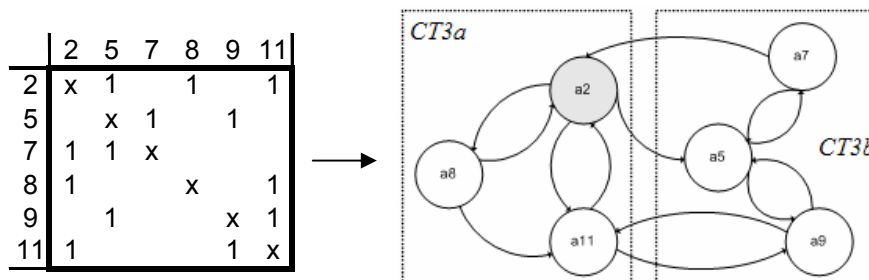

Fig. 9. Decomposition of the Coupled Task 3 with the Spectral Algorithm

The analysis of this decomposition shows that the obtained organization leads to an optimal solution:

- the number of actors in the two groups is balanced (3 in each group),

- the interactions between the two groups are minimized (4 exchanges)

- the number of interactions inside the groups are homogeneous (5 exchanges in the first group and 4 in the second)

We collected in Table 2 the estimated results for the decomposition of CT3 according to the spectral algorithm. Note that the serial decomposition lets to a choice between two possibilities according to the order of execution for the subgroups. The decomposition compared to a concurrent realization of CT3 (see Table 1) allows to reduce the workload and so to improve the resources management. For the serial decomposition, the execution of CT3b before CT3a leads to the equilibration of the workload during the two phases and increases the total duration of the process. The parallel decomposition still brings a reduction to the total workload while increasing the lead-time. In this case, main efforts are concentrated in the first phase.

Execution CT3a / CT3b		In series (a < b)	In series (b < a)	Parallel
E	Phase 1	13.62	25.41	39.04
	Phase 2	43.50	26.99	8.09
	Total	57.12	52.40	47.13
T	Phase 1	11.16	14.92	10.74
	Phase 2	14.92	11.63	10.96
	Total	26.08	26.56	21.70

Table 2. Estimated workload and lead-time for the decomposition of CT3

There is not an optimal configuration here. The choice must be guided by the availabilities of the resources and the urgent character and not by the finalization of the project. These models allow to compare different choices in the organization of a process. The automated calculation of criteria gives a good help to decide if manager have to decompose a task or not. This approach gives the means to estimate in a quantitative and relevant way the quality of the organization of cooperative processes during decomposition as far as we choose the border line.

The proposed approach to decompose a complex cooperative process into several load balanced workgroups and to schedule them onto several development process steps, is based on two phases. First, the analysis of information flows with the DSM allows a general decomposition of the process

in coupled tasks. In a second phase, these tasks are separated with a spectral algorithm in relatively independent groups to be able to work in parallel. These two levels define the steps of the PDP and a basic structure for the organization.

3. RESPONSABILITIES FOR THE DEVELOPMENT PROCESS

To determine the main responsibilities of the development process, two kinds of actors can be identified by means of the CG analysis:

- the validating members,

- the task managers.

3.1 Identification of the validating members

Validating members are actors who are authorised to validate or to reject propositions on models, components, values … These actors are able to react on a proposition expressed by another actor. That means the kind of interactions in a coupled task is either:

- a bilateral cooperation (interdependency between two actors),

- or a total cooperation (all the links between actors are used and the matrix is full),

- or actors implied in a circuit and not in a hierarchical cooperation.

In a general manner, the following definition (definition 1) is proposed to identify a validating member inside a work group.

Definition 1: The actor is employed in strongly related components of a cooperation graph, which constitutes the set of validating members.

3.2 Identification of the Task managers

Task managers are actors who are able to observe the whole work process and to immediately react by sending new orders or updating information to the other actors. For this, a task manager must have a central position in a work group. This position and the information exchange with the other actors result from the expected skills of a task manager. Thus, the survey to specify this actor is to analyze

the different distances between each actor. Concepts of eccentricity and anti eccentricity defined in the graph theory are used.

Definition of the eccentricity: *let G be a graph and v be a vertex of G, the eccentricity of the vertex v is the maximum distance from v to any vertex.*

That is $e^+(v)=max\{d(v,w) : w$ in $V(G)\}$;

Definition of the anti eccentricity: *let G be a graph and v a vertex of G, the anti eccentricity of the vertex v is the maximum distance from any vertex to v.*

That is $e^-(v)=max\{d(w,v) : w$ in $V(G)\}$.

In the context of a cooperative work organization, the eccentricity represents the ability of an actor to capture information inside the group with a minimum of mediators and the anti eccentricity represents the ability of an actor to propagate information inside the group with a minimum of mediators. Thus, to identify the actor who has a central position in a PDP, i.e. a task manager, the sum of eccentricity and anti eccentricity for each actor is calculated. The actor with the minimum value is the task manager. When two or more actors have the same result, the actor who has the least number of elements corresponding to the max $d(a_j, a_i)$ and the max $d(a_i, a_j)$ is the task manager. Thus, the definition that is applied to identify the task manager is:

Definition 2: The task manager in a CG is identified by the actor a_i with: **min$\{e^+(a_i)+e^-(a_i) : a_i$ in V(CG)$\}$**. When two or more actors have the same value, the task manager is the actor a_i with:

$min|\{max\{d(a_i, a_j)\}\cup\{max \{ d(a_j, a_i) \} : a_i, a_j$ in $V(CG)|$

The groups 1 and 2 (TC1 and TC2) in the example are composed by activities engaged in a total cooperation. Inside these groups, each actor has the same distance with any other one. So these workgroups have no task manager and the actors employed in this cooperation have the same level of responsibilities (validating members).

	a_2	a_5	a_7	a_8	a_9	a_{11}	$e^+(a_i)$	$e^+(a_i) + e^-(a_i)$
a_2	0	1	2	1	2	1	2	**4**
a_5	2	0	1	4	1	2	4	6
a_7	1	1	0	2	2	2	2	5
a_8	1	2	3	0	2	1	3	7
a_9	2	1	2	3	0	1	3	5
a_{11}	1	2	3	2	1	0	3	5
$e^-(a_i)$	2	2	3	4	2	2		

Table 3. Eccentricity and Anti-eccentricity of activities in Coupled Task 3

Applied to the third coupled task depicted in the Fig. 2, we identify that this group collects (Table 3):

- a set of validating members: the whole actors involved in the coupled task;

- a task manager: the actor 2, who minimizes the sum of eccentricity and anti-eccentricity.

The application of these two definitions used to analyze dependencies in a work group enables to check if the group can efficiently work in a cooperative way, i.e. it collects the right skills, otherwise there is a need to restructure it.

4. INTERFACES BETWEEN WORK GROUPS

4.1 Objectives

One of the major difficulties to manage a development project is to ensure the information consistency between the different work groups. The objective required in the PDP as defined in the ISO 9001 is to both identify:

- a way for communication, which ensures an efficient transfer of the information between groups,

- the responsibilities for information consistency within each work group.

Firstly, optimizing the information transfer needs to globally minimize the distance between the producer and the consumer of this information. That means it is important to avoid mediators for transfer of information from an actor to another one. The objective is to reduce semantic losses on the information. Secondly, the information consistency must be the responsibility of specific actors named key-members. The key-members are couples of actors who have to control the interface between groups. The key-members between two groups (for example group α and group β) involve two actors:

- one actor of the group α is responsible of the information consistency inside his group;

- one actor of the group β is responsible of the information exchange with the group α.

4.2 Identification of the key-members

To identify these actors, the definition 2 for a set of two groups, which must be interfaced is applied. Four cases can be analysed (Fig. 10):
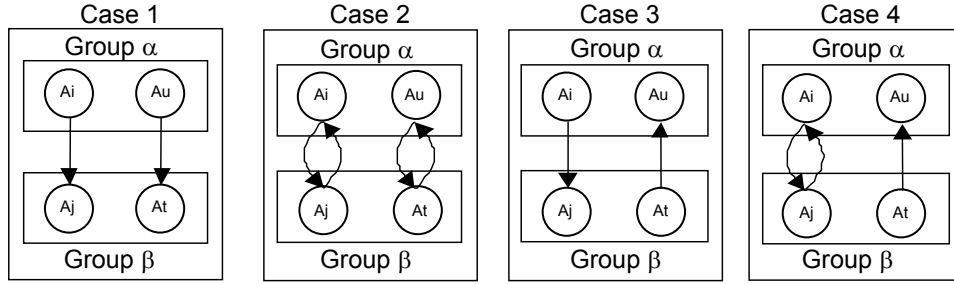

Fig. 10. Cases for two group's interfaces

Case 1: There is only a hierarchical cooperation between the two groups. In this case, the objective is to determine only one couple of actors to interface the groups. Let's consider two groups α and β. For each couple of actors (a_i, a_j), with a_i belongs to the group α and a_j belongs to the group β, expressions $e^+(a_i)$, $e^-(a_j)$ and, $e^+(a_i) + e^-(a_j)$ are calculated. $e^+(a_i)$ represents the ability of the actor a_i to capture information in the group α with a minimum of mediators and $e^-(a_j)$ represents the ability of the actor a_j to propagate information in the group β with a minimum of mediators. The key-members of this cooperation are the couple of actors (a_i, a_j) with:

$$\text{Min } \{(e^+(a_i)+e^-(a_j))\}$$

Case 2: The couples of actors competent to interface two groups are involved in a bilateral cooperation. That means that the abilities of the actors to transfer information in two directions must be taken into consideration. In this case, one couple of actors responsible to manage the communication involving two groups is identified. The same properties defined in the case 1 are used but applied in the two directions. Thus, the key-members of this cooperation are the couple of actors (a_i, a_j) with a_i belongs to the group α and a_j belongs to the group β and:

$$\text{Min } \{ (e^+(a_i) + e^-(a_i) + e^+(a_j) + e^-(a_j)) \}$$

Case 3: The communication between the two groups is managed by couples of actors in hierarchical cooperation. But, at least one couple of actors transfers information only in one direction and at least one couple transfers information in the other direction. That means that two couples of actors responsible to manage the group interface must be selected. To identify the key-members, the properties defined in the case 1 are applied, firstly to identify the couple of actors responsible to transfer information from the group α to the groupβ, and secondly to identify the couple of actors responsible to transfer information from the group β to group α.

Case 4: This case is a hybrid solution between the case 2 and the case 3. Some couples of actors are involved in a bilateral cooperation and others in a hierarchical cooperation. In this case, the couples involved in a hierarchical cooperation are not taken into account and the properties defined in the case 2 are consequently applied. The reason of this choice is that it is better to have only one couple of actors in a bilateral cooperation, who manages all the communications between two groups than two couples of actors in a hierarchical cooperation.

Remark: When actor couples have the same position in cooperation, the definition 2 described in the previous section is applied. The objective is to decide between the set of all possible couples.

Let a_i, a_α belong to group α and a_j, a_β belong to the group β, the expression to be evaluated for each couple is:

$$\text{Case 1\&3:} \quad \mathbf{Min|\{max\{d(a_i, a_\alpha)\}\} \cup \{max\{d(a_\beta, a_j)\}\}|}$$

$$\text{Case 2\&4:} \quad \mathbf{Min |\{max \{d(a_i,a_\alpha)\}\} \cup \{max \{d(a_\alpha,a_i)\}\} \cup \{max \{d(a_j, a_\beta,)\}\} \cup \{max \{d(a_\beta, a_j)\}\}|}$$

To illustrate this study, the example of the interface analysis connecting the second and the third coupled task (group 2 and 3) depicted in the Fig. 2 is used. The analysis is applied to three couples: (a_{10}, a_9); (a_{10}, a_7); (a_4, a_8). The cooperation is hierarchical (case 1), thus only one couple have to be choose. The results are collected in Table 4. Activities in group 2 are engaged in a total cooperation, so $e^+(a_i)$ is equal to 1 for activities 4 and 10. $e^-(a_j)$ for activities in group 3 result from Table 3.

Couple (a_i, a_j)	$e^+(a_i)$	$e^-(a_j)$	$e^+(a_i) + e^-(a_j)$
(a_{10}, a_9)	1	2	3
(a_{10}, a_7)	1	3	4
(a_4, a_8)	1	4	5

Table 4. Key members for interface between second and third group

Thus, the actors 9 and 10 have the smallest value and constitute the couple, which will be responsible for the interface between these two groups. Same manner key members for the interface cooperation between group 1 and group 3 are actors 5 and 6.
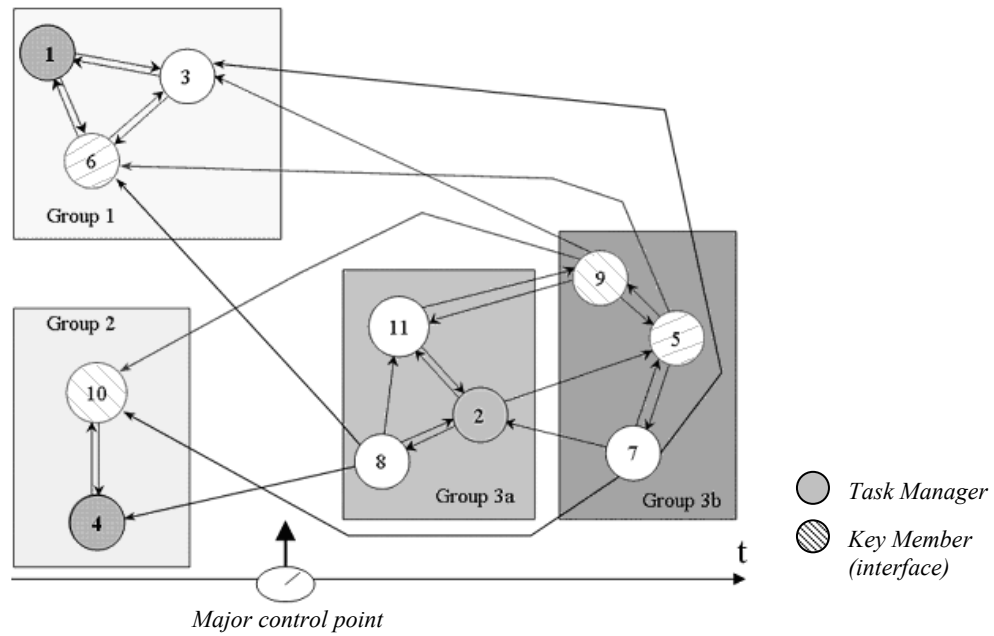

Major control point

Fig. 11. Structural and strategic organization of the cooperative work example

The figure 11 resumes the main results of the proposed method applied on the cooperative work example. In order to equilibrate the workload and the role repartition, task manager for group 1 and 2 are chosen in minimizing the extra-group exchanges number. A principal control point is defined for review and validation of the project after realization of group 1 and 2. Other control points can be defined after work iteration of group 3.

5. CONCLUSION

In this article, an organizing method for complex cooperative development process is proposed. For this, the recommendations of the ISO 9001 standard are applied. The approach offers modeling and structuring elements to methodologically improve the collaborative work. The first point to study is the specification of the process steps. After a first decomposition based on the information flows analysis, several coupled tasks are obtained. If these tasks consist of to many actors, a second

decomposition level, which allows to reduce the size of the groups, is defined. This decomposition is based on a spectral algorithm, which takes into account the workload balancing and the minimization of the number of exchanges between groups.

The method needs to evaluate the duration of each step in order to determine the optimal organization. For that purpose, on the basis of the works of Smith and Eppinger, (1997) relative to concurrent design systems, a model for parallel approach is used. This model enables to compare different organizations according to criteria like duration and workload. The proposed decomposition allows to find a compromise which optimizes both delays and costs on the basis of the volume of exchange information.

The definition of the responsibilities among the work groups is required in the PDP of ISO 9001 standard. In a complex process involving a lot of exchanges, it is important to clearly define the actors who must ensure the consistency of information produced and consumed by the different groups. The approach proposes the identification of these actors. This technique takes into account criteria that minimize the distance of the information linking the producer and the consumer. The objective is to minimize the semantic losses during the information routing.

This method can give the means of defining the network needs and using the Web technology as well as possible. The main objective is to link the methodological and the technological parts of the cooperative work. The proposed algorithms can be used to both configure collaborative environment (software tools for cooperation, network architecture …) and to define the work organization management (structural and temporal decomposition, responsibilities ...)

REFERENCES

David M., Z. Idelmerfaa and T. Divoux (2004). Organizing Cooperative Work for the Product Development Process, *IFAC Symposium on INformation COntrol problems in Manufacturing, INCOM 04*, Salvador-Bahia, Brazil.

Diaz M., F. Vernadat, Villemur (1996). Spécification et Réalisation Formelles de Systèmes Coopératifs. *Ingénierie des protocoles, CFIP 96*, pp. 357-376.

Gebala D.A. and S.D. Eppinger (1991). Methods for Analyzing Design Procedures. *ASME International Conference on Design Theory and Methodology*, pp. 227-233.

Harary F. (1962). A Graph Theoric Approach to Matrix Inversion by Partitioning. *Journal of numerical mathematics*.

ISO/CD2 9001 (2000). Quality management systems. Requirements. ISO Standard.

Karypis G. and V. Kumar (1998). A Parallel Algorithm for Multilevel Graph Partitioning and Sparse Matrix Ordering. *Journal of Parallel and Distributed Computing*, **Vol. 48**, pp. 71-95.

Krishnan V., Eppinger S.D., Whitney D.E.(1997). A model-based framework to overlap product development activities, *Management Science*, **Vol. 43, n°4**: p. 437-451.

Kusiak A., Larson and J. Wang (1994). Reengineering of Design and Manufacturing Processes. *Computers and Industrial Engineering*, **Vol. 26, n°3**, pp. 521-536.

Pothen A., H.D. Simon and K.P. Liou (1990). Partitioning Sparse Matrices with Eigenvectors of Graphs. *SIAM Journal of Matrix Analysis and Applications*, **Vol. 11, n°3**, pp. 430-452.

Rondeau E., Z. Idelmerfaa and J. Richard (1999). Identification of Group Organization during a Design Process by Using Cooperation Graphs. *Concurrent Engineering Research Application*, ISSN 1063-293 X, **Vol. 7**, pp. 191-199.

Smith R.P. and S.D. Eppinger (1997). A Predictive Model of Sequential Iteration in Engineering Design. *Management Science*, **Vol. 43**, n°8, pp. 1104-1120.

Steward D.V. (1981). The Design Structure System: A Method for Managing the Design of Complex Systems. *IEEE Transactions on Engineering Management*, **Vol. EM-28**, n°3, pp. 71-74.

Todd D. and P. Sen (1999). Distributed Task Scheduling and Allocation using Genetic Algorithms. *Computers and Industrial Engineering* **n°37**, pp.47-50.

Zhao J.M. and Z.J. Liu (2003). Team-size Constraint based moderate Decoupling Algorithm for large Coupled Task Sets. *10th International Conference on Concurrent Engineering Research and Applications,* ISBN 90-5809-622-X, pp 123-128.