

Sylvie BESSETTE

Etudiante en Génie cognitif

Jean VAUCHER

Professeur en intelligence artificielle
Laboratoire INCOGNITO,
Département d'informatique
de l'Université de Montréal

Marius FREGIER

Architecte, Informaticien
Chercheur au laboratoire GAMSAU
Ministère de l'Urbanisme
et du Logement du gouvernement
français
Professeur invité

Nicolas CHOUROT

Informaticien
Faculté de l'Aménagement
Université de Montréal

**Système expert pour la
composition de colonnades
d'ordres classiques selon
les règles de l'architecture
régulière formulées
par Andréa PALLADIO**

Summary:

This paper describes the transfer of an expertise on the Architectural rules of composition formulated by the Architect Andrea PALLADIO, during the XVI century.

The regular Architecture, in other words, the Architecture organized by rules, belongs François DE CHAMBRAY, constitute a traditional expertise, consequently few controversial.

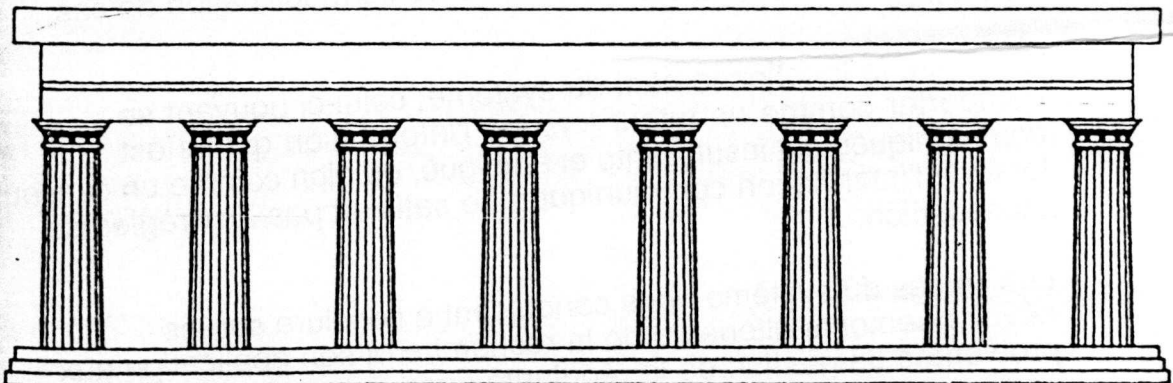
In this context, the subset quite well formulated and concise of the composition rules of colonnades is taken as a subject of expertise transfer.

The system resulting is based on the verification of the coherency of the composition rules, under the constraints, deduced from informations communicated by the user (variables linked by the composition rules).

The solutions produced by the expert system are communicated to a graphical kernel, under the form of parameters, this made possible by graphical means, both, the control of the results, and the handling of new constraints to modify these results.

Behavior of the system is described, it can perform as a tutor, in the case of insufficient or ambiguous information, or as a censor when the information doesn't apply to the composition rules.

The limits of the system brings us to conclude on further developments and their use to manage complexity by regularity, a problem which arises to carry-out an open building industry using compatible components.



ORDRE : DORIQUE

Mots-clés: Architecture régulière, base de donnée graphique, morphologie des données factuelles et cognitives, primitives graphiques de haut niveau, représentation des connaissances.

Resumé:

Cette communication décrit un transfert d'expertise portant sur les règles de composition de l'architecture régulière formulée au XVIIe siècle par l'architecte Andréa PALLADIO.

L'Architecture régulière autrement dit organisée par des règles selon François de CHAMBRAY, constitue une expertise traditionnelle qui de ce fait est peu discutée. Dans ce cadre, le sous ensemble, relativement bien formulé et concis, des règles de composition des colonnades est choisi comme sujet de transfert d'expertise.

Le système qui en résulte fonde son fonctionnement sur la vérification de la cohérence des "règles de composition" à l'égard de contraintes déduites des informations communiquées par l'utilisateur (variables liées par les règles de composition).

Les solutions fournies par le système expert sont communiquées sous forme de paramètres à un noyau graphique afin de permettre une vérification visuelle des résultats obtenus, ainsi que la saisie graphique de paramètres des contraintes de modification de ces solutions.

On décrit le fonctionnement du système; celui-ci pouvant se comporter comme un tuteur lorsque l'information qui lui est communiquée est insuffisante et ambiguë, ou bien comme un censeur lorsque l'information communiquée ne satisfait pas aux règles de composition.

Les limites du système nous conduisent à conclure sur les développements ultérieurs de la recherche et son application aux problèmes de gestion de la régularité dans la complexité, rencontrés en "industrialisation ouverte", et par composants compatibles.

1. Introduction au domaine d'expertise:

L'architecture, à travers les temps, a souvent suivi des lois bien particulières. Cette partie de l'architecture qui suit un ordre des choses peut être caractérisée d'architecture régulière. L'époque classique illustre très bien ce phénomène. Elle définit des règles qui ont été acceptées et suivies pendant une très longue période. Ces règles ont été enregistrées dans les Traités d'Architecture, sous forme mathématique et géométrique. La diffusion de ceux-ci a permis de faire connaître et de reproduire les principes classiques. Les traités ont été un outil théorique pour l'élaboration de méthodes et ont contribué à généraliser la pratique de l'architecture.

Des traités d'architecture qui ont vus le jour depuis le 15^e siècle, les "Quatre livres de l'Architecture" de l'architecte Andrea Palladio, édités à Venise en 1570, en est un des plus connus. Ce traité présente un recueil des oeuvres de Palladio, dont les célèbres Villas Palladiennes, oeuvres associées à des relevés de bâtiments anciens. Il offre une sorte de catalogue qui illustre, à l'aide de planches, des édifices de toutes natures, anciens et modernes, qu'utiliseront les architectes pendant près de 3 siècles. Ces planches sont très précises et donnent, en projection géométrique, les plans, coupes et élévations des édifices. L'apport théorique de l'ouvrage est aussi très important. Il est alimenté de la pensée de Vitruve, architecte romain du 1^{er} siècle avant J-C à qui on doit le traité De Architectura faisant état de l'architecture à cette époque. Il fait aussi référence à Bramante et Alberti, 2 autres grands architectes. De plus il constitue le recueil de connaissances architecturales le plus complet et cohérent paru jusqu'alors. Il est basé sur le concept de proportions modulaires.

La grande innovation de Palladio a été d'utiliser les Ordres de l'architecture classique dans ses villas. Les Ordres sont définis comme étant la "combinaison des parties d'un édifice dans des proportions telles que leur ensemble soit harmonieux et régulier" (Larousse). Il y a trois Ordres grecs: l'Ordre Dorique, l'Ordre Ionique, l'Ordre Corinthien, et deux Ordres romains: l'Ordre Toscan et l'Ordre Composite.

Palladio fait appel à trois principes dans son oeuvre: la commodité, la durée et la beauté. La commodité indique la répartition

des entités (loggia, chambres, caves, greniers,...) selon leur mode d'utilisation, que se soit selon des critères fonctionnels ou pour refléter la richesse du propriétaire. La durée fait référence à la solidité qui est le résultat de la bonne construction et du respect des règles de la statique, propres à certains éléments (colonnes, ouvertures,...). La beauté fait appel aux critères esthétiques.

2. Détermination d'un sous-ensemble pour un système expert:

Le domaine de l'architecture régulière et plus particulièrement l'oeuvre de Palladio, riche en normes et informations précises, se prête très bien à l'implantation d'un système expert. Il est cependant très vaste et un sous-ensemble en sera extrait.

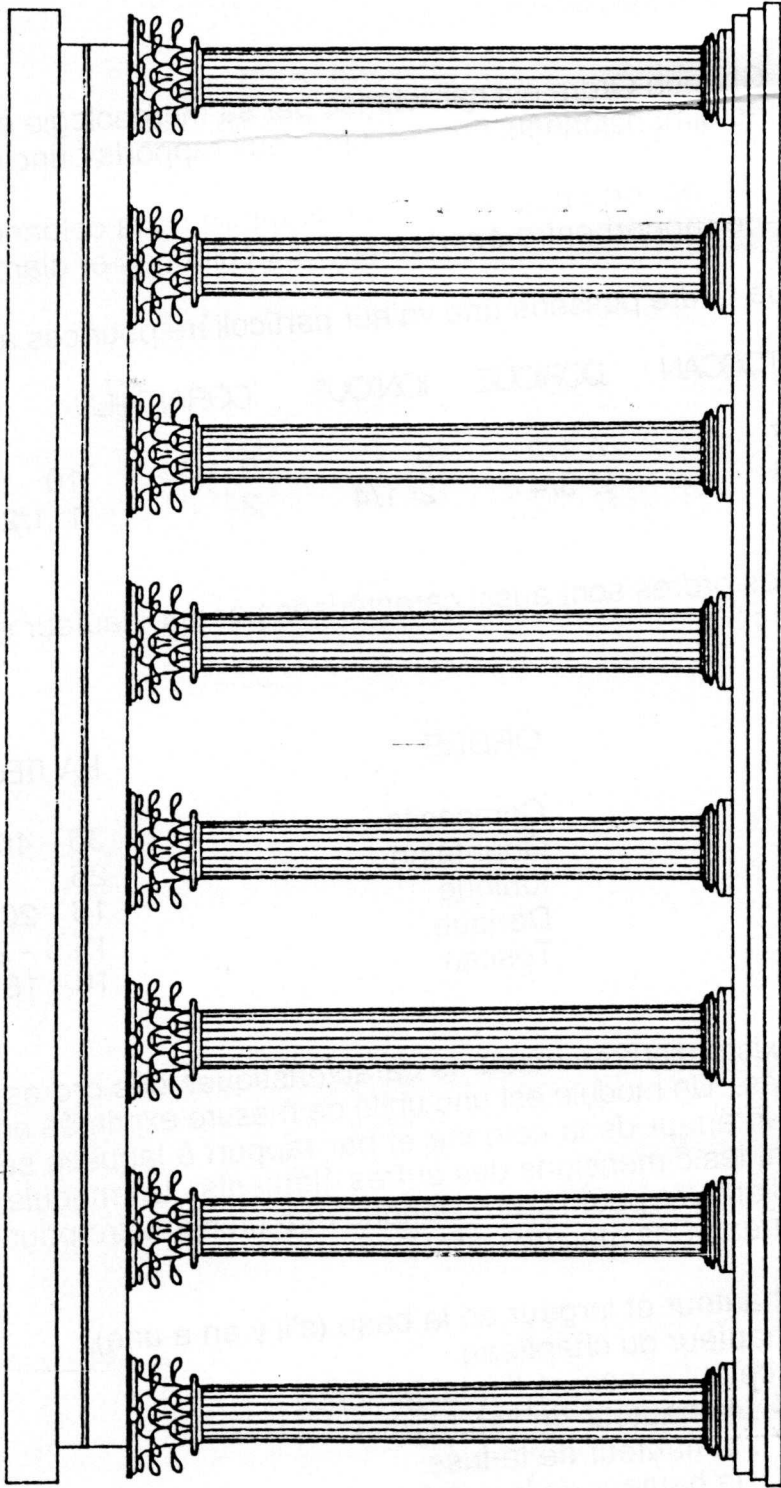
Le premier livre de l'oeuvre de Palladio définit des règles élémentaires pour trois catégories d'éléments de l'architecture des maisons. On retrouve d'abord les éléments matériels: fondations, murs, plafonds,...etc, puis les unités spatiales: chambres, portes, fenêtres...etc, puis les Ornaments, éléments des Ordres architecturaux, qui sont gérés par une ensemble de règles géométriques et proportionnelles particulièrement rigoureux. Les Ornaments ont été choisis comme sous-ensemble, à cause de leur bonne définition et à cause de leur lien avec les Ordres, très importants dans l'histoire. Les critères à caractère social ont été éliminés à cause de leur imprécision.

3. Description du problème:

Un Ordre Architectural se définit comme la superposition verticale d'un certain nombre d'éléments architecturaux ayant une fonction ornementale et structurelle:

BASE + COLONNE + CHAPITEAU + ENTABLEMENT.

L'entablement est composé de l'architrave, de la frise et de la corniche. La base est caractérisée par une hauteur et une saillie (partie dépassant le diamètre de la colonne). Ces éléments ne se retrouvent pas tous dans chaque Ordre. Les figures 1 à 5, extraites des planches de Palladio, illustrent les composantes pour chaque style.



ORDRE : CORINTHIEN

Chaque Ordre est caractérisé par sa morphologie et par des rapports dimensionnels précis. Les deux rapports principaux sont :

- le rapport entre hauteur et diamètre de la colonne (H/d).
- le rapport entre écartement des colonnes et diamètre (E/d).

Chaque Ordre possède une valeur particulière pour ces attributs :

	TOSCAN	DORIQUE	IONIQUE	CORINTHIEN	COMPOSITE
H/d	7	7 1/2	9	9 1/2	10
E/d	3	2 3/4	2 1/4	2	1 1/2

Les ordres sont aussi caractérisés par une hauteur en termes absolus :

ORDRE	HAUTEUR COLONNE
Composite	35 - 40 pieds
Corinthien	25 pieds
Ionique	18 - 20.5 pieds
Dorique	15.5 - 20 pieds
Toscan	10 - 18 pieds

Il existe d'autres mesures caractéristiques des ordres, données en modules. Un module est une unité de mesure exprimée en terme du diamètre inférieur de la colonne et par rapport à laquelle se définissent les dimensions des autres éléments. Un module est égal à un demi diamètre pour l'ordre Dorique et à un diamètre pour les autres ordres. Ces mesures sont :

- la hauteur et largeur de la base (s'il y en a une).
- la hauteur du chapiteau.
- en ce qui concerne l'entablement :
 - . la hauteur de l'architrave.
 - . la hauteur de la frise.
 - . la hauteur de la corniche.

Ces valeurs constituent les mesures de base de chaque ordre.

Les Ordres peuvent se retrouver sous forme de colonnade simple (une colonnade est un ensemble de colonnes). Une colonnade a les caractéristiques suivantes :

- les colonnes reposent sur le sol par leur base (sans piedroit).
- les colonnes sont reliées uniquement par l'entablement.
- l'espace entre les deux colonnes centrales, appelé entrecolonnement central, doit être plus large que les autres (entrecolonnements complémentaires). Cette différence est d'environ 10 à 20% et peut être pris par rapport aux bases ou aux fûts.
- elle a une portée (longueur).
- elle a un nombre pair de colonnes.

4. Fonctions du système expert:

La fonction du système expert sera de fournir une aide à la conception d'une colonnade dans un des 5 Ordres architecturaux. Le concepteur construit sa colonnade en fournissant au système certains paramètres qui seront déterminants dans la situation où il se trouve. Par exemple si seul la hauteur importe, il peut demander au système de calculer le reste. Si l'aspect esthétique l'intéresse, il peut indiquer le nombre de colonnes désiré ou encore l'Ordre. Il peut aussi avoir des contraintes de longueur de bâtiment, d'entrecolonnement central (nombre de personnes déterminé devant pouvoir y passer, etc...).

Les fonctions du système sont donc :

- Si l'utilisateur est inexpérimenté, le système fonctionne comme un tuteur en lui posant des questions ou en répondant à ses questions grâce à un système d'aide.
- Si l'utilisateur fournit trop peu de paramètres, le système fonctionne comme un guide, en donnant des solutions possibles ou encore en lui indiquant les paramètres qu'il doit fournir pour déterminer une solution.
- Le système peut aussi agir en censeur, en exprimant si une description de colonnade suit les règles ou s'il y a contradiction et pourquoi.

Les problèmes principaux à résoudre sont :

- 1) A partir de la longueur de colonnade approximative et d'un entrecolonnement central minimum, déterminer les ordres acceptables, le nombre de colonnes et l'entrecolonnement complémentaire.
- 2) A partir de la longueur de la colonnade et du style, donner la hauteur de la colonnade.
- 3) A partir de la hauteur et de l'entrecolonnement central, donner les ordres compatibles et la longueur minimale de la colonnade.
- 4) A partir de la hauteur et du style, donner l'entrecolonnement minimum.
- 5) A partir de la hauteur, du style et de la portée, donner l'entrecolonnement.

Ces problèmes peuvent être résolus à l'aide des informations décrites précédemment.

Les connaissances du système sont composées de faits et de règles représentant les dimensions, rapports et liens des paramètres des colonnes et colonnades pour chaque Ordre architectural (décrits dans la section précédente).

L'ignorance de certains paramètres, lors de la conception, est contournée par la détermination arbitraire de ceux-ci par le système. Cette méthode est un bonne façon de guider l'utilisateur qui comprend mieux ainsi les effets d'un paramètre sur les autres.

Une aide a été introduite dans le système. Il est possible par exemple demander la description des éléments d'une colonne, les caractéristiques des Ordres, les paramètres permettant de construire une colonnade, les paramètres influençant une information.

Les éléments d'une colonne sont :

- le fût
- la base
- le chapiteau
- l'entablement composé de l'architrave, la frise et la corniche.

Les paramètres d'une colonnade sont :

- la portée
- le nombre de colonnes
- l'entrecolonnement central (par rapport aux bases ou aux fûts)
- l'entrecolonnement complémentaire (par rapport aux bases ou aux fûts)
- la colonne employée
- l'ordre

Les paramètres d'une colonne sont :

- la hauteur de la colonne
- le diamètre
- la base (présence, hauteur et saillie)
- la hauteur du chapiteau
- l'entablement (présence, hauteur de l'architrave, frise et corniche).

Les caractéristiques des Ordres sont :

- les dimensions absolues de hauteur des colonnes
- l'intervalle E/d accepté
- l'intervalle H/d accepté
- la description de la colonne type :
 - . hauteur de la colonne
 - . base (hauteur, présence et saillie)
 - . hauteur du chapiteau
 - . entablement (présence, hauteur de l'architrave, frise et corniche).

La stratégie utilisée pour résoudre les problèmes est celle de propagation de contraintes avec l'aide du backtracking.

Lors de la conception d'une colonnade dans un ordre architectural, plusieurs paramètres inter-reliés doivent être déterminés. Chacun de ces paramètres exerce une contrainte sur les autres et par conséquent, chaque facteur peut difficilement être considéré indépendamment.

Le système présenté ici profite de ces relations (contraintes) pour résoudre le problème. Il fait ceci en limitant, lorsqu'un

paramètre est fixé, l'intervalle des valeurs acceptées des paramètres qui en dépendent. Ainsi, lorsqu'on détermine une valeur, on précise les autres en restreignant les possibilités et on diminue, par le fait même le risque d'un mauvais choix. La méthode de propagation de contraintes avec backtracking est utilisée.

4. Représentation des connaissances.

Les facteurs de conception et leurs liens sont définis par rapport à l'Ordre architectural. En effet, les paramètres sont bornés par des valeurs différentes dans chaque Ordre et les relations entre paramètres y sont décrits par des rapports différents.

La structure de données utilisée pour représenter les connaissances est centrée sur les Ordres. Les données sont réparties dans 3 structures principales :

- 1) Structure des valeurs types et intervalles pour chaque Ordre.
- 2) Structure des contraintes.
- 3) Structure des liens entre paramètres permettant l'inférence et la propagation des contraintes.

4.1. Structure des valeurs types et intervalles pour chaque Ordre.

Cette structure est codée sous forme de propriétés qu'on affecte à chaque Ordre. Elle contient les valeurs caractéristiques décrivant une colonnade dans un Ordre donné. Ces paramètres sont :

- l'entrecolonnement complémentaire par rapport aux fûts (propriété ECO_F).
- la hauteur de passage (propriété H_PAS).
- la hauteur totale (propriété H_TOT).

- colonne : base:

.présence	(P_BASE)
.hauteur	(H_BASE)
.saillie	(S_BASE)

- hauteur du chapiteau (H_CHA)
- hauteur du fût (H_FUT)

entablement :

.présence	(P_ENT)
.hauteur entablement	(H_ENT)
.hauteur architrave	(H_ARCHI)
.hauteur frise	(H_FRISE)
.hauteur corniche	(H_COR)

N.B. Les valeurs ci-dessus sont toutes en terme du diamètre de la colonne.

On ajoute à cette structure la propriété "bornes" qui contient une liste donnant les bornes de l'intervalle accepté pour les paramètres suivants :

- H_PAS : hauteur de passage.
- D : diamètre de la colonne.
- ECO_F : entrecolonnement complémentaire par rapport aux fûts.
- ECE_F : entrecolonnement central par rapport aux fûts.
- H_BASE : hauteur de la base.
- H_CHA : hauteur du chapiteau.
- H_ARCHI : hauteur de l'architrave.
- H_FRISE : hauteur de la frise.
- H_COR : hauteur de la corniche.
- H_TOT : hauteur totale de la colonnade.
- S_BASE : saillie de la base.
- H_FUT : hauteur du fût.
- H_ENT : hauteur de l'entablement.
- DIM_D : diminution du diamètre de la colonne.

4.2. Structure des contraintes.

Cette structure contient les informations décrivant les rapports entre les paramètres devant être respectés quel que soit l'Ordre. Elle est représentée à l'aide de deux listes : `contraintes` et `bornes_a_verifier` qui définissent respectivement des relations sous forme d'égalité et d'intervalles. Dans les deux liste, chaque contrainte est associée à un message qui sera utilisé pour informer l'utilisateur des contradictions (données erronées venant de l'utilisateur) qui surviennent. Ces messages aideront l'utilisateur à comprendre pourquoi la solution qu'il cherche n'est pas compatible.

4.3. Structure des liens entre paramètres.

Cette structure contient 2 types d'informations :

- les paramètres suffisants donnent pour un paramètre donné les paramètres qui suffisent pour le déterminer et la fonction pour le faire. Cette structure est utilisée lors de l'étape d'inférence qui tente d'évaluer les paramètres dont la valeur n'a pas encore été choisie. Cette structure permettra de ne parcourir que les règles associées au paramètre qui nous intéresse plutôt que toutes les règles.

Cette information est codée sous forme de propriété de la façon suivante :

```
(PUTPROP 'paramètre
pour      '(( (liste d'ensembles de paramètres suffisants
           déterminer le paramètre)
           (fonction d'évaluation)
           )
pour      ( (liste d'ensembles de paramètres suffisants
           déterminer le paramètre)
           (fonction d'évaluation)
           )
           )
           'param_suff
)
```

- les paramètres liés donnent pour un paramètre donné la liste des paramètres dont il influence la valeur et la façon dont ils sont liés, représentée sous forme de fonction donnant la borne minimum et maximum.

Cette information est codée similairement à la première structure de la manière suivante :

```
(PUTPROP 'paramètre
      (( paramètres qui doivent être connus pour pouvoir
        propager l'information )
        ( paramètres liés)
        ( fonction d'évaluation pour la borne minimum)
        ( fonction d'évaluation pour la borne maximum)
      )
      'param_liés
    )
```

Cette structure est utilisée pour propager les effets de la détermination d'un paramètre sur les paramètres qui sont liés à lui. Ceci est fait à l'étape de "propagation de contraintes".

5. Autres structures de données utilisées dans le programme.

- Structure pour représenter la valeur des paramètres de conception lors de la résolution du problème.

Chaque paramètre est un atome auquel on associe la propriété bornes contenant les bornes minimum et maximum pour la valeur du paramètre dans la solution actuelle. La valeur du paramètre est le contenu de l'atome. Ces paramètres sont les mêmes que ceux permis en entrée et sont décrits en détail dans la section Entrées du Guide de l'utilisateur.

- Liste des solutions (Solutions) : contient la description de l'espace solution qu'on va parcourir. Sa forme est :

```
((paramètre (solutions possibles non parcourues))
  (paramètre (solutions possibles non parcourues))
```

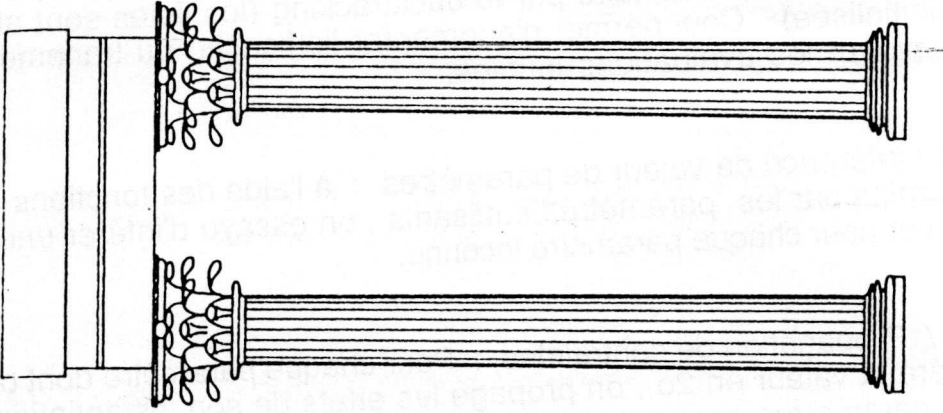
- ...)
- Liste des paramètres connus (par_connus) : contient le nom des paramètres dont on connaît la valeur (celle-ci a soit été donnée par l'utilisateur ou a été inférée).
 - Liste des paramètres inconnus (par_inconnus) : contient le nom des paramètres dont on ne connaît pas encore la valeur.
 - Liste des paramètres inférés (par_inférés) : contient le nom des paramètres inférés depuis la dernière propagation.
 - Liste des propagations (propagations) : contient, pour chaque paramètre déterminé à l'aide de l'heuristique, la liste de paramètres auxquels on a propagé des valeurs.
 - Liste-propage : contient la liste des paramètres affectés par la propagation depuis le dernier choix (heuristique). Cette liste fournit les éléments de la liste précédente.

Et les listes suivantes :

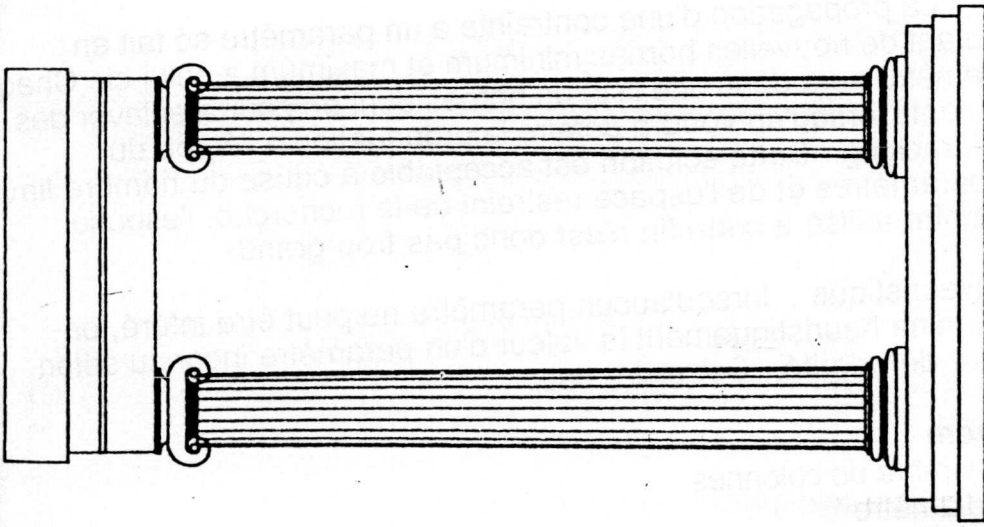
- paramètres : nom des paramètres permis.
- opérateurs : nom des opérations permises.
- liste_aide : nom des sujets sur lesquels on a de l'information (HELP).
- ordres : nom des ordres possibles.
- priorité : liste donnant le nom des paramètres par ordre de priorité dans lequel on veut les déterminer par l'heuristique (si nécessaire).
- messages : messages d'erreur.
- résultats : solutions au problème.

6. Méthode utilisée. Le système procède en 5 étapes :

1o / Vérification des bornes et contraintes : le système vérifie si le contexte actuel de la solution satisfait les contraintes et bornes de conception.



ORDRE I CORINTHIEN



ORDRE I IONIQUE

A mesure qu'une contrainte ou borne est vérifiée (tous les paramètres impliqués sont alors connus), elle est éliminée de la liste des contraintes ou bornes à vérifier, celle-ci restant vraie tant que la solution n'est pas défaite par le backtracking (les listes sont alors réinitialisées). Ceci permet d'augmenter l'efficacité du traitement en éliminant les évaluations inutiles.

2o / Inférence de valeur de paramètres : à l'aide des fonctions fournies par les paramètres suffisants, on essaye d'inférer une valeur pour chaque paramètre inconnu.

3o / Propagation de contraintes : Pour chaque paramètre dont on a inféré la valeur en 2o, on propage les effets de son instantiation sur les paramètres encore inconnus qui lui sont liés. Ceci se fait à l'aide des paramètres liés. De plus pour chacun de ceux-ci, si leurs bornes ont été modifiées alors on propage cette modification de la même manière sur leurs propres paramètres liés... Cette étape permet de restreindre l'intervalle des valeurs possibles pour les paramètres inconnus, et donc de faciliter leur détermination à l'aide d'heuristique si on ne peut le faire par inférence.

La propagation d'une contrainte à un paramètre se fait en ajoutant de nouvelles bornes minimum et maximum à celui-ci. Chaque paramètre aura donc une liste de bornes dont on pourra enlever des éléments lorsqu'on voudra défaire la propagation à l'étape du backtracking. Cette solution est acceptable à cause du nombre limité de paramètres et de l'espace restreint de la recherche, l'espace mémoire utilisé à cette fin n'est donc pas trop grand.

4o/ Heuristique : lorsqu'aucun paramètre ne peut être inféré, on détermine heuristiquement la valeur d'un paramètre inconnu selon l'ordre de priorité. Cet ordre est :

- . L'ordre
- . Le nombre de colonnes
- . Le diamètre
- . L'entrecolonnement complémentaire /fûts
- . La hauteur de passage
- . L'entrecolonnement central /fûts
- . La portée
- . La présence de la base

- . La présence de l'entablement
 - . La hauteur totale
 - . La hauteur de la base
 - . La saillie de la base
 - . La hauteur du fût
 - . La hauteur du chapiteau
 - . La hauteur de l'entablement
 - . La hauteur de l'architrave
 - . La hauteur de la frise
 - . La hauteur de la corniche
 - . L'entrecolonnement complémentaire /bases
 - . L'entrecolonnement central /bases.
- Pour le nombre de colonnes on distingue 2 fonctions heuristiques dépendant de la connaissance ou non de la valeur de la portée. Un ensemble de solutions possibles est généré, celui du premier cas sera plus restreint que le deuxième. La première valeur générée est affectée à NBCOL, les autres sont conservées dans la liste Solutions comme alternatives pour l'étape de backtracking, si la solution ne marche pas.
- Pour les autres paramètres, on affecte le point milieu entre la borne minimum et la borne maximum du paramètre.
- Les ordres sont tous parcourus si aucune précision n'est donnée par l'utilisateur.

50/ Backtracking : lorsqu'on arrive à une contradiction, il faut revenir au dernier choix fait à l'aide de l'heuristique et affecter une autre solution, s'il y en a. Les paramètres inférés et la propagation faite depuis ce point doivent être défaits.

Ceci est fait en partant du dernier choix, en mettant à NIL les paramètres inférés à la suite de celui-ci, et en enlevant des paramètres affectés, les éléments de leur liste de bornes celles associées à cette propagation.

Le parcours de l'espace solution se fait de la manière suivante :

- On choisit un ordre (si l'utilisateur ne l'a pas donné).
- On détermine un nombre de colonnes quelconque.
- On tente de déterminer le reste à l'aide des étapes précédentes.

La résolution se fait en choisissant d'abord un Ordre. Ceci est dû au fait que les connaissances dépendent de l'Ordre et qu'on ne peut rien faire si on ne détermine pas celui-ci. Tous les Ordres seront parcourus un à un (à moins que l'utilisateur n'est précisé une valeur pour l'Ordre) afin d'essayer de trouver une solution pour chacun de ceux-ci.

Le deuxième problème dans la recherche d'une solution est associé au paramètre du nombre de colonnes et vient du fait que l'espace solution de celui-ci est discret. Il est particulièrement difficile à calculer si le diamètre et l'entrecolonnement ne sont pas connus, à cause des liens entre ces derniers, la portée et le nombre de colonnes. Le problème a été réglé ici en associant un espace solution à ce paramètre (assez restreint) et en essayant chaque valeur jusqu'à ce qu'une solution soit trouvée ou qu'il n'y ait plus d'alternatives (l'Ordre est alors incompatible). Le changement de valeur se fait à l'étape du backtracking.

7. l'Interaction graphique:

L'édition graphique des solutions est obtenue en lançant sous VMS l'exécution d'un module spécifique du logiciel graphique GRAPH-EDIT-2D écrit en Pascal.

Le logiciel GRAPH-EDIT-2D est composé de quatre modules:

- . Le module de dessin (GRAPH 2D un noyau graphique)
- . Le module EDIT-LIB qui permet l'implantation d'applications graphiques interactives.
- . Le module d'édition graphiques de figures géométriques (GRAPH-EDIT).

Le module d'édition de mosaïque pour le pavage du plan (MOSEDIT) et GRAPH-EDIT sont utilisés.

-GRAPH-2D

Ce module est un ensemble de procédures Pascal qui permettent la programmation d'applications graphiques en deux dimensions.

Ces primitives constituent un sous-ensemble proche des primitives du standard américain "CORE", cela pour des raisons de transportabilité, bien qu'un très grand nombre d'appareils de sortie graphique puissent être installés, trois interfaces graphiques, seulement, ont été implantées: l'une pour le terminal graphique visual 550 compatible au Tektronix 4014 (avec effacement sélectif) , pour l'imprimante à points Printronix LXY-11 compatible au protocole CALCOMP et le traceur de courbe HP7475A.

Pendant les structures des données de cette librairie de primitives graphiques sont, grâce aux utilitaires du langage Pascal, basées sur des structures de listes plutôt que sur des tableaux pour des raisons d'efficacité et de flexibilité de manipulation.

-GRAPH-EDIT:

Ce module permet l'édition interactive de figures en deux dimension à l'aide d'appareil de saisie graphique tels qu' une table à digitaliser. Ce module utilise les modules GRAPH-EDIT et EDIT-LIB pour effectuer toutes les sorties graphiques nécessaires.

Ce module très utiles pour tout genre d'édition graphique 2D (élaboration de plans d'Architecte, pièces industrielles, diagrammes explicatifs ...), mais n'est aucunement spécialisé pour l'une d'entre elles, c'est simplement un outil de dessin assisté par ordinateur.

Quatre principales classes de fonctions se retrouvent dans cet éditeur:

- 1/ Le choix de la figure à saisir (segment, suite de segments, arc d'ellipse, bspline, etc ...),
- 2/ Le type de manipulation à appliquer sur les figures déjà saisies translation, rotation, symétrie,, sélection, copie de figures à transformer, etc ...).
- 3/ Le mode de saisie des figures (sous la contrainte ou non d'une grille, orthogonalité des tracés, etc ...).
- 4/ La gestion sur fichiers de figures (sauvegarde de la session, catalogue des fichiers, lecture d'un fichier de figures, etc ...)

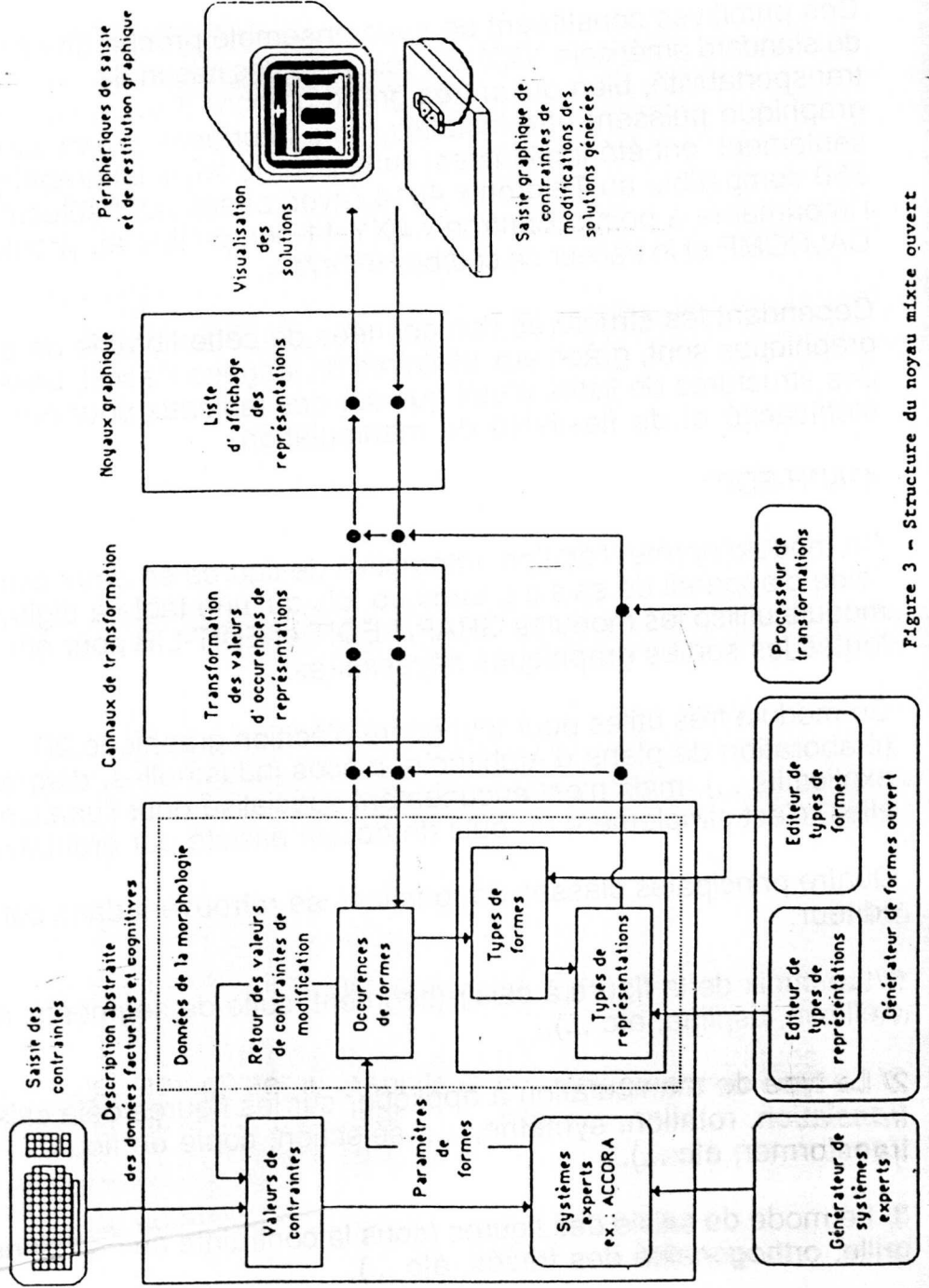


Figure 3 - Structure du noyau mixte ouvert

Ce module a permis d'éditer les représentation graphique des composants figures de base tels les chapiteaux, pour lesquels une représentation programme aurait été extrêmement fastidieuse. La représentation d'une colonnade est obtenue par l'exécution des tracés des représentations de ces composants aux proportions déterminées par les paramètres des solutions par le système expert.

-EDIT-LIB:

C'est un ensemble d'utilitaires d'interaction graphique telle que gestion graphique de l'écran, menu, ICON, suivi des coordonnées de la tablette graphique.

Ce module est très utile pour le développement d'éditeurs graphiques ainsi que de toute application nécessitant une interaction graphique.

Conclusion:

Par le choix de ce transfert d'expertise, dans le domaine bien défini de l'Architecture régulière, nous avons illustré un certain nombre de mécanismes visant développer les bases d'une génération de systèmes de CAO "intelligents".

Les travaux à venir devraient généraliser la méthode afin de disposer à terme d'un noyau mixte ouvert (fig. 3) de système de CAO, c'est à dire un système, qui assiste le concepteur dans la formulation des règles de la conception, et permette, à celui-ci de formuler la morphologie des objets et les représentations graphiques qui leurs sont associées, et, de manipuler les données de la conception, à l'aide de ces représentations graphiques, sous la contrainte des données cognitives.

Bibliographie:

Les sources suivantes ont servi de base d'information :

- Une analyse du contenu théorique et méthodologique des "Quatre livres de l'architecture" de Andrea Palladio, effectuée par le groupe d'études pour l'application des méthodes scientifiques à l'architecture et l'urbanisme (GAMSAU) de Marseille-luminy:

Principes, normes et modèles théoriques d'un traité
d'architecture du XVI^e siècle,
I Quattro libri Dell'Architettura de Andrea Palladio,
Paolo Donati et Paola Jervis, GAMSAU, Juin 1979.

Elle constitue le rapport de synthèse d'une recherche visant à
contrôler la valeur du concept de "grammaire normative" en
architecture, en analysant et mesurant sur un exemple, qui est
l'oeuvre de Palladio, la portée, les limites et les conséquences d'une
"théorie universelle".

- Les Quatre livres de l'architecture de Palladio, traduit de l'italien
en version intégrale par Fréart de Chambray (1650) et réédité en
1980.
- Palladio delineated and explained, William Pain, Gregg international
Publishers, 1972.
Livre expliquant l'oeuvre de Palladio.
- Principle of Artificial Intelligence, Tiogo Publishing Company,
Palo Alto, Ca., Nilson, N.J., 1980.

MAKECHAP (CHAP, ORDRE, HAUT, LARG, PTATT);



MAKEFUT (FUT, ORDRE, HAUT, LARG, DIMD, PTATT);



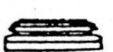
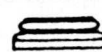
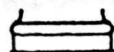
MAKEBASE (BASE, HAUT, LARG, PTATT);



UNIRFIG (COL, BASE);

UNIRFIG (COL, FUT);

UNIRFIG (COL, CHAP);



ACORA: Un système expert pour la composition de colonnades d'ordres classiques selon les règles de l'Architecture régulière formulées par Andréa PALLADIO

ACORA: An expert system for the composition of classic styles colonnades, belongs the rules of the regular architecture formulated by Andréa PALADIO.

Noms des auteurs: Nicolas CHOUROT, (faculté de l'Aménagement).
Sylvie BESSETTE, Jean VAUCHER (Département
d'Informatique et de Recherche Opérationnelle).
Université de Montréal.
Marius FREGIER(GAMSAU).

Adresses: Université de Montréal, C.P. 6128, succursale A, Montréal
(Québec)
H3C3J7, Canada. Téléphone 514 343 60 01
GAMSAU/EAM - 70, Route Léon Lachamp 13288
MARSEILLE CEDEX 9 FRANCE - Téléphone (33) 91 41 11 85