

Documentation générale



Préface

SYNAPXIS est un logiciel conçu pour piloter et gérer des machines et des processus robotisés. Ce document présente et explique les fonctionnalités du logiciel dans le cadre de son utilisation par l'utilisateur final. La documentation relative à la configuration et la programmation de **SYNAPXIS** (version 2.7.0), qui est destinée à l'*intégrateur* du logiciel n'est pas incluse dans ce document.

Le Locle, août 2013

HE-Arc/IHC/UR-LPR

Table des matières

1	Présentation	1
1.1	Fonctionnalités de base	1
1.1.1	Vue 3D	2
1.1.2	Etat du mode simulation	2
1.1.3	Panneau latéral	2
1.1.3.1	Général	2
1.1.3.2	Préhenseurs	2
1.1.3.3	Frame	2
1.2	Fonctionnalités avancées et configuration	2
2	Vue 3D	5
2.1	Vues enregistrées	5
2.2	Options de représentation	6
2.3	Modes de la souris	6
2.3.1	Souris + clavier	6
2.3.2	Souris	6
2.4	Touches de contrôle	7
3	Sécurité	9
3.1	Introduction	9
3.2	Niveaux d'accès	9
3.3	Points d'accès	9
4	Robot	11
4.1	Tool	11
4.2	Vitesses	11
4.3	Transitions	12
4.3.1	Test	12
4.3.2	Edition	12
4.4	Paramètres	13
4.5	Connexion	13
4.6	Simulation	13
4.7	Avancée	14

5	Références	15
5.1	Templates	15
5.1.1	Description	15
5.1.1.1	Paramètres	15
5.1.1.2	Configuration	15
5.1.2	Template de base	16
5.1.3	Héritage des templates et des références	16
5.2	Configuration	16
5.2.1	Description	16
5.2.2	Configuration au niveau du template	16
5.2.2.1	Robot	16
5.2.2.2	Préhenseur générique	17
5.2.2.3	Usinage	17
5.2.2.4	Simulation	17
5.2.2.5	Palettes	17
5.2.3	Configuration au niveau de la référence	18
5.2.3.1	Préhenseur	18
5.2.3.2	Frame	18
5.2.3.3	Palette	18
5.2.3.4	Simulation	19
5.3	Paramètres	20
5.3.1	Description	20
5.3.2	Édition des paramètres au niveau du template	21
5.3.3	Édition des paramètres au niveau de la référence	21
5.4	Dossiers de références	22
5.5	Création et ouverture d'une référence	23
5.6	Importation/Exportation	24
5.6.1	Données exportées	24
5.6.2	Importation	25
5.7	Fichiers associés	25
5.7.1	Description	25
5.7.2	Fichiers associés et références	26
5.7.2.1	Duplication de fichiers associés	26
5.7.2.2	Archivage de fichiers associés	26
6	Préhenseurs	29
6.1	Introduction	29
6.2	Composants	29
6.2.1	Transformée	29
6.2.2	Représentation 3D	30
6.2.3	Catégories	30
6.3	Définitions	30
6.3.1	Liste de composants	30
6.3.2	Transformée totale	30
6.3.3	Utilisation d'outils	30

6.3.4	Représentation 3D	30
6.4	Préhenseur courant du robot	31
7	Frames	33
7.1	Introduction	33
7.2	Généralités	33
7.2.1	Point de transition	33
7.2.2	Outils	33
7.2.3	Simulation - représentation 3D	34
7.2.4	Position	34
7.2.4.1	Frame parent	34
7.2.4.2	Décalage avant	34
7.2.4.3	Frame	34
7.2.4.4	Décalage après	34
7.2.4.5	Décalage utilisateur	35
7.3	Frame plan	35
7.4	Frame cylindrique	35
7.4.1	Position et orientation	35
7.4.1.1	Frame orienté selon Y - Orientation libre	35
7.4.1.2	Orientation standardisée	36
7.4.2	Diamètre et limites	36
7.5	Palette	36
7.5.1	Emplacements	36
7.5.2	Entraxes	36
7.5.3	Offset	37
7.5.4	Pick	37
7.5.5	Positions des emplacements quelconques	37
8	Outils	39
8.1	Introduction	39
8.2	Outils et références	39
8.2.1	Remise à zéro	39
8.2.2	Application de l'usure	40
8.3	Outils	40
8.3.1	Plan	40
8.3.2	Disque	40
8.3.3	Disque profilé	41
8.3.4	Cylindre	42
8.3.5	Sphère	42
8.3.6	Cône	42
8.4	Paramètres des outils	42
8.4.1	Alarmes	42
8.4.2	Limites	42
8.4.3	Vitesse de l'outil	42
8.4.4	Usure	43

CHAPITRE 1

Présentation

1.1 Fonctionnalités de base

L'interface de **SYNAPXIS** est conçue de manière à ce que l'utilisateur puisse accéder rapidement et de manière ergonomique aux fonctionnalités de base. Son layout est compatible avec une utilisation sur écran tactile, en environnement industriel (par exemple : utilisation de gants par l'opérateur en mode production).

La fenêtre principale est composée de 3 parties principales :

1. Vue 3D.
2. Etat du mode simulation.
3. Panneau latéral.

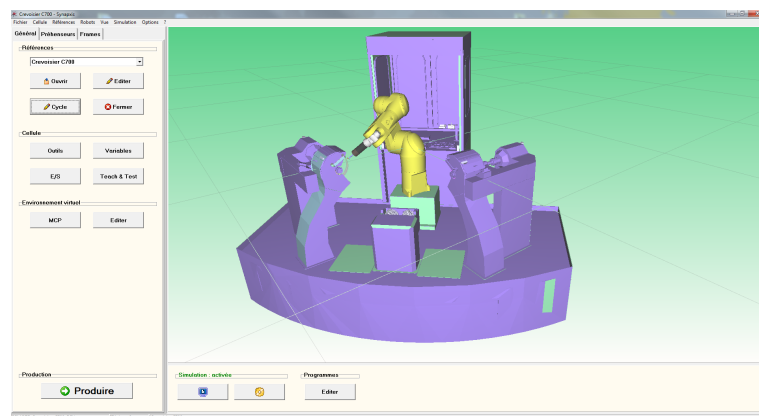


Figure 1.1: Fenêtre principale

1.1.1 Vue 3D

La vue 3D permet de représenter virtuellement les différents éléments de la machine. Elle est utile notamment pour simuler les mouvements du robot, vérifier les accessibilités et les éventuelles collisions du robot avec son environnement.

1.1.2 Etat du mode simulation

Cette partie de l'interface permet d'activer et désactiver le mode simulation de **SYNAPXIS**. En mode *simulation activée*, les mouvements sont dirigés vers le ou les robots simulés correspondants, et les actions sur les IOs sont sans effet (les entrées simulées valent systématiquement *false* (binaire) ou 0.0 (analogique)).

1.1.3 Panneau latéral

1.1.3.1 Général

L'onglet *général* est décomposé en quatre parties :

1. La partie *Références* est utilisée pour la sélection, l'édition et la modification de références. Pour plus de détail se référer au chapitre 5 de ce document.
2. La section *Cellule* donne accès au *gestionnaire d'outils*, aux *variables machines* ainsi qu'aux *IOs* de la machine.
3. L'*Environnement virtuel* permet d'accéder à la gestion de l'environnement virtuel (option) ainsi qu'au panneau de commande des robots virtuels.
4. Accès au module de *production*. Ce module permet de gérer le flux de production (lots), et de superviser et piloter le processus : affichage d'informations, interruption et reprise, etc.

1.1.3.2 Préhenseurs

L'onglet *Préhenseur* donne accès à l'édition et l'utilisation des préhenseurs. Ils sont classés en fonction du robot auquel ils sont associés.

1.1.3.3 Frame

L'onglet *Frame* donne accès à l'édition des frames. Les *frames* regroupent les frames plans, cylindriques (axe d'outils tournants, etc), les palettes et les axes externes. Ils sont classés en fonction du robot auquel ils sont associés. Le chapitre 7 détaille l'édition et l'utilisation des *frames*.

1.2 Fonctionnalités avancées et configuration

Le menu donne accès aux fonctionnalités avancées et à différents dialogues de configuration de **SYNAPXIS** :

Robots Ce menu donne accès à la configuration des différents robots de la machine (connexion, paramètres, transitions, etc).

Vue Ce menu permet d'accéder à la fenêtre de gestion de la vue 3D. Il permet également de sélectionner rapidement une position de la vue 3D. Les fonctionnalités sont expliquées au chapitre 2.

Options Le menu options donne accès aux fonctionnalités avancées de **SYNAPXIS** :

Répertoire de travail Gestion des *répertoires de travail* ; un répertoire de travail, ou *workspace*, est le dossier racine qui contient l'ensemble des données propres à une machine. Le changement de workspace courant de **SYNAPXIS** nécessite un redémarrage de l'application.

Floating Gestion des paramètres de la fenêtre *Floating*. Celle-ci permet d'accéder rapidement à diverse fonctionnalités de la machine, au travers de fonctions macros prédéfinies. La floating est affichée/masquée à l'aide de la touche *F5*.

Sécurité Gestion des niveaux d'accès et des accès utilisateurs vers les différentes fonctionnalités de **SYNAPXIS**.

Backups Configuration des backups automatiques.

Import/Export Importation/Exportations de différentes données machines.

Interfaces Gestion et éditions des interfaces dédiées à la machine.

TCP/IP Manager Gestion des clients et serveurs TCP/IP de **SYNAPXIS** (optionnel).

Configuration système Configuration de générale de **SYNAPXIS**.

CHAPITRE 2

Vue 3D

2.1 Vues enregistrées

Plusieurs *vues enregistrées* peuvent être définies et utilisées pour repositionner rapidement la représentation 3D. Lors de sa création, une nouvelle *vue enregistrée* adopte la position courante de la vue 3D. Il est possible d'éditer la valeur de cette position ou de la redéfinir en fonction de la position courante de la vue 3D (*Ici*).

Remarques :

- L'ordre des vues peut être modifié (drag-and-drop).
- La *vue initiale* est la vue qui est instaurée au démarrage de **SYNAPXIS**.
- Le temps de déplacement entre la vue courante et la vue sélectionnée est constant. La vitesse de déplacement est adaptée en fonction de la distance entre les deux positions de vue.
- La fenêtre principale est toujours active à l'arrière plan de manière à pouvoir modifier la position de la vue 3D avec la souris alors que la fenêtre de configuration est ouverte au premier plan.



Figure 2.1: Edition des vues

2.2 Options de représentation

Les options suivantes permettent de personnaliser la représentation de la vue 3D :

Facteur d'échelle Le facteur d'échelle permet d'adapter l'ensemble des positions et des dimensions de l'environnement 3D en fonction de la grandeur générale de l'application représentée (par exemple : usine entière ou processus très compact).

Origine Représentation du système d'axes principal de la vue 3D. Les axes *X*, *Y*, *Z* correspondent aux couleurs *rouge*, *vert*, *bleu* (RVB/RGB), de la même manière que pour tous les systèmes d'axes représentés dans **SYNAPXIS**.

Grille Représentation du plan XY par une grille.

Accélération matérielle Utilisation des ressources de la carte graphique pour mémoriser et afficher la représentation des différents éléments représentés. Cela a pour effet de libérer des ressources du processeur.

Polygones pleins Remplissage des triangles et polygones qui forment une image 3D surfacique.

2.3 Modes de la souris

2.3.1 Souris + clavier

Ce mode permet de déplacer la vue à l'aide des boutons de la souris et de la touche clavier *Ctrl* :

Ctlr + bouton gauche Zoom en fonction du déplacement de la souris.

Ctlr + bouton milieu Rotation horizontale et verticale en fonction du déplacement et de la position de la souris.

Ctlr + bouton droit Translation en fonction du déplacement et de la position de la souris.

La rotation et la translation sont faites par rapport à l'endroit pointé par la souris si celui-ci correspond à une géométrie de l'environnement 3D. La molette de la souris peut aussi être utilisée pour réaliser le zoom en observant comme point de fuite la position du curseur de la souris.

2.3.2 Souris

Ce mode permet de déplacer la vue uniquement à l'aide des boutons de la souris :

Rotation molette Zoom avec comme point de fuite la position du curseur de la souris. La translation de la vue est réalisée par la combinaison d'un zoom arrière/avant et gestion du point de fuite.

Bouton milieu + bouton droit Rotation autour de l'axe normal de la vue en fonction du déplacement et de la position de la souris.

Bouton milieu Rotation horizontale et verticale en fonction du déplacement et de la position de la souris.

La rotation et la translation sont faites par rapport à l'endroit pointé par la souris si celui-ci correspond à une géométrie de l'environnement 3D. La translation avec *Ctrl + bouton droit* est également disponible dans ce mode.

2.4 Touches de contrôle

Les fonctions de déplacement de la vue peuvent être attribuées à des *touches de contrôle*, liées par exemple à un clavier virtuel (association du code ASCII de la touche à une action). Les fonctions disponibles sont les suivantes :

1. Zoom avant.
2. Zoom arrière.
3. Translation gauche.
4. Translation droite.
5. Translation haut.
6. Translation bas.
7. Rotation horizontale haut.
8. Rotation horizontale bas.
9. Rotation verticale gauche.
10. Rotation verticale droite.
11. Rotation normale anti-horaire.
12. Rotation normale horaire.
13. Vues - début de la plage 1-10.

La touche spécifiée pour *Vues - début de la plage 1-10* permet d'appliquer la première position de vue enregistrée. Les neuf touches dont les codes ASCII suivent celui de cette touche appliquent dans l'ordre les neuf vue enregistrées suivantes.

CHAPITRE 3

Sécurité

3.1 Introduction

Le module sécurité permet de gérer les différents points d'accès ainsi que les niveaux d'accès utilisateurs. Il est ainsi possible de gérer l'accès à certaines fonctionnalités en fonction de l'utilisateur.

3.2 Niveaux d'accès

Par défaut, deux niveaux d'accès sont définis : *root* et *constructor*. Ils ne peuvent pas être supprimés, et le mot de passe du niveau *constructor* n'est pas modifiable. Il est destiné à protéger les fonctionnalités liées aux options payantes du logiciel.

D'autres niveaux d'accès peuvent être créés, avec ou sans mot de passe (par exemple : régleur, opérateur, etc). La hiérarchie des niveaux d'accès est établie selon l'ordre de création, les niveaux *constructor* et *root* ayant les priorités les plus élevées. De cette manière, l'utilisateur *root* possède par exemple tous les droits des niveaux hiérarchiquement inférieurs.

Un historique permet de mémoriser les différents accès réalisés lorsqu'un utilisateur utilise son mot de passe.

3.3 Points d'accès

Les fonctionnalités suivantes sont notamment protégées par un point d'accès :

Security/Settings Gestion des niveaux d'accès et des points d'accès.

References/Settings Options du gestionnaire de références.

Production/Settings Interface de production.

IO/Settings Configuration et édition des sources d'entrées/sorties et de leurs composants.

Robot/Settings Configuration du robot.

Robot/Déplacement du robot vers un point de transition Validation du mouvement lorsque le robot est à côté d'un chemin de transition et qu'il doit être repositionné en vitesse lente sur le point de transition le plus proche.

CHAPITRE 4

Robot

L'interface de configuration du robot permet de spécifier les paramètres propres aux robots réel et virtuel.

4.1 Tool

Edition du *tool* courant. Le robot considéré (réel ou simulé) dépend du mode actif ou non de la simulation.

4.2 Vitesses

Edition des vitesses prédéfinies : *Default*, *Approch*, *Depart*, *Teach*, *Security*. Ces vitesses peuvent avoir une valeur différente en fonction du mode de production. Elles sont utilisées dans les différentes fonctions macro liées à la gestion du cycle de production, par le biais d'une instruction spécifique. Il est ainsi possible de changer les vitesses du processus sans apporter de modification du programme. Les valeurs sont exprimées en % de la vitesse maximale.

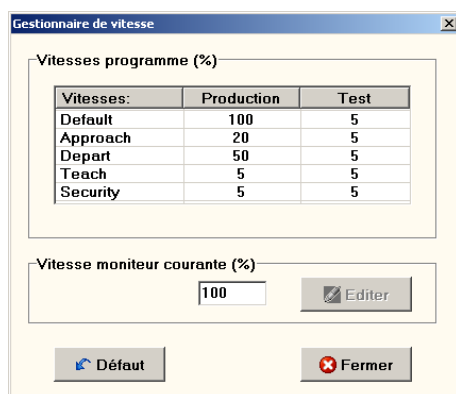


Figure 4.1: Vitesses du robot

4.3 Transitions

Un réseau de points de transition, ainsi que les chemins à suivre pour passer d'un de ces points à l'autre peut être défini pour chaque robot. Ces points de transition, ou de *sécurité* forment un *réseau de points*, reliés par les *chemins de transition*. Les points sont de type *joint* ou *articulaire*, ce qui définit à la fois la position et la configuration du robot.

Cette fonctionnalité permet d'amener le robot dans une zone donnée de son environnement et d'assurer que le chemin qu'il parcourt est connu, quelle que soit la position courante du robot (point de départ). Si celle-ci ne coïncide pas avec un des points de sécurité existants, le point le plus proche est déterminé ; un déplacement en vitesse *Security* est alors proposé à l'utilisateur, qui peut ou non valider le mouvement.

4.3.1 Test

Le mode test permet de sélectionner un point de transition et de lancer l'exécution du mouvement depuis la position actuelle du robot (simulé ou réel) vers cette destination. Il est possible de régler la vitesse d'exécution du mouvement.

4.3.2 Edition

Ce mode permet d'éditer les différentes positions de transition ainsi que les chemins qui les lient. Un point de transition peut être défini comme point de travail ou non. Lorsqu'un point de transition doit être sélectionné dans **SYNAPXIS**, seuls les points de transition définis comme des points de travail sont proposés à l'utilisateur. Les autres points permettent simplement de réaliser le mouvement désiré, sans que le robot ne s'y arrête.

Un chemin est défini comme une suite de points de transition. Si plusieurs chemins sont définis, ils doivent tous avoir un seul et unique point commun avec au moins un autre chemin, de manière à être connecté au réseau. Il faut veiller à ne pas créer de boucle en liant directement ou indirectement, en plus d'un point, deux chemins. Plusieurs chemins peuvent être exécutés partiellement ou entièrement, dans un sens ou l'autre lors de l'exécution d'une transition.

Remarques :

- L'option *lissage* permet de générer un mouvement robot continu ; celui-ci ne marque pas d'arrêt aux différents points de transition avant d'être arrivé à son point de destination. Les points intermédiaires sont donc seulement approchés en fonction des paramètres de lissage ou *blending Leave et Reach*.
- Lors de la recherche du point le plus proche, une comparaison est faite avec la position actuelle de chaque axe du robot et les coordonnées des différents points de transition définis. Une pondération est effectuée en fonction du numéro de l'axe, de manière à ce qu'une différence donnée au niveau de l'axe 1 ait une incidence plus forte que la même différence au niveau de l'axe 4 par exemple. D'autre part,

il est possible d'ignorer le dernier axe du robot lors de cette recherche car celui-ci est généralement *multi-tours*.

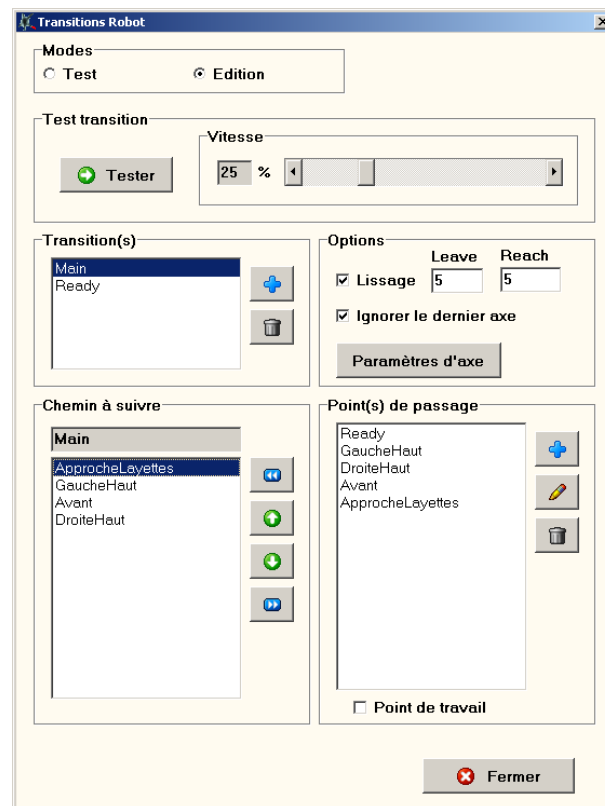


Figure 4.2: Edition des transitions

4.4 Paramètres

Les paramètres du robot permettent de configurer le dialogue entre **SYNAPXIS** et le contrôleur du robot. D'une manière générale, ils ne doivent pas être modifiés.

4.5 Connexion

La connexion de **SYNAPXIS** vers le robot requiert l'adresse TCP/IP du contrôleur ainsi que le numéro de port spécifique. Cette connexion peut être établie de manière automatique au lancement de l'application.

4.6 Simulation

Deux assemblages doivent être définis afin de pouvoir afficher automatiquement dans la représentation 3D les *tools* et les *frames* liés au robot.

4.7 Avancée

Options réservées à l'installation de l'application et au diagnostic des fonctionnalités robot.

CHAPITRE 5

Références

Le *Gestionnaire de Références* permet de gérer facilement les données nécessaires au fonctionnement d'une application (processus) destinée à plusieurs références, celles-ci pouvant être assimilées à différents "types de pièce". Chaque référence possède des *paramètres* qui lui sont propres ainsi que un ou plusieurs états de *configuration*.

5.1 Templates

5.1.1 Description

Chaque référence dépend (hérite) d'un template, qui représente la "famille" à laquelle elle appartient. Les données utiles au fonctionnement de l'application sont déclarées au niveau du template, et peuvent prendre une valeur spécifique pour chaque référence.

SYNAPXIS permet de gérer plusieurs templates, desquels peuvent être issues autant de références que nécessaires.

5.1.1.1 Paramètres

Les paramètres sont utilisés par les programmes de la machine et définissent les valeurs qui sont propres à chaque référence (distance d'approche, position de charge, décalage RZ, etc.), à distinguer des valeurs propres à la machine.

5.1.1.2 Configuration

La configuration fixe les différentes relations matérielles entre la référence et la machine : le robot utilisé (dans le cas d'une application multi-robots), son positionnement par rapport à celui-ci (pièce portée/fixe) ainsi que les *tools*, *frames* et *palettes* utilisés. La configuration permet également de définir la représentation 3D de la référence. Ces données sont nécessaires à l'édition du cycle d'usinage (suivi de trajectoire) ainsi que pour la simulation de l'application.

5.1.2 Template de base

Dans le cas d'une application complexe, il peut être nécessaire d'avoir plusieurs templates. Le template de base peut alors être utilisé pour créer des paramètres communs entre tous les templates, ceux-ci héritant directement du template de base.

5.1.3 Héritage des templates et des références

La figure ci-dessous illustre l'héritage des templates et des références :

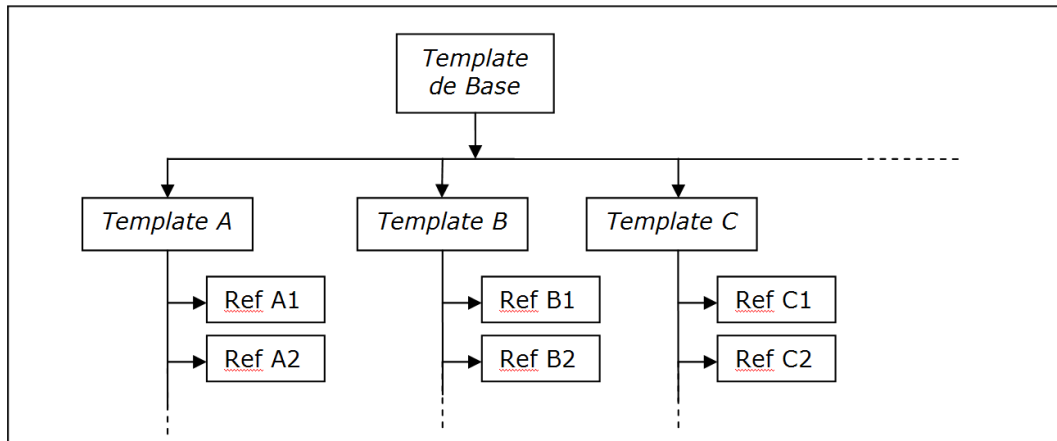


Figure 5.1: Héritage des templates et références

5.2 Configuration

5.2.1 Description

Dans le cas d'une application simple (un seul robot, un seul type d'opération), un état de configuration unique permet de décrire tous les liens de la référence avec l'environnement robotique de la cellule. S'il y a plusieurs robots ou plusieurs types d'opérations (par exemple ébavurage en pièce fixe, polissage en pièce portée), il est nécessaire de définir plusieurs états de configuration.

5.2.2 Configuration au niveau du template

Les différents états de configuration sont définis au niveau du template et identifiés par leur nom. Ils renseignent les informations suivantes :

5.2.2.1 Robot

Le robot concerné par l'*état de configuration*, ainsi que la configuration dans laquelle il est utilisé : en mode *pièce portée* ou *pièce fixe* par rapport au robot.

5.2.2.2 Préhenseur générique

Cette option permet de définir un préhenseur unique au niveau du template pour la préhension des pièces (pièce portée uniquement). Le préhenseur sélectionné doit comporter 2 composants *modifiables* :

fixation Composant spécifique à la référence qui assure la fixation mécanique de la pièce sur le préhenseur (pince, chuck) et qui est amené à changer d'une référence à l'autre, en terme de sa transformée ou de représentation 3D.

offset Composant permettant d'ajouter une correction à la transformée du composant de fixation. Il n'a pas de représentation 3D.

Chaque référence définit ensuite la transformée et la représentation 3D du composant de *fixation*, ainsi que la transformée du composant *offset*. Ces valeurs sont appliquées au *Préhenseur générique* lorsque la configuration de la référence est appliquée (ouverture manuelle, chargement de références en mode automatique). Cette option permet de ne pas multiplier le nombre de préhenseurs existants en fonction du nombre de références. Si elle n'est pas activée, le choix du préhenseur dédié à la référence est fait au niveau de la référence.

5.2.2.3 Usinage

L'utilisation d'outils machine pour l'usinage (pièce portée ou fixe).

5.2.2.4 Simulation

L'état initial de la simulation, qui permet de spécifier quel *état de simulation* appliquer lorsqu'une référence est ouverte depuis l'interface.

5.2.2.5 Palettes

Les différents types de palette (palette de charge, décharge, etc).

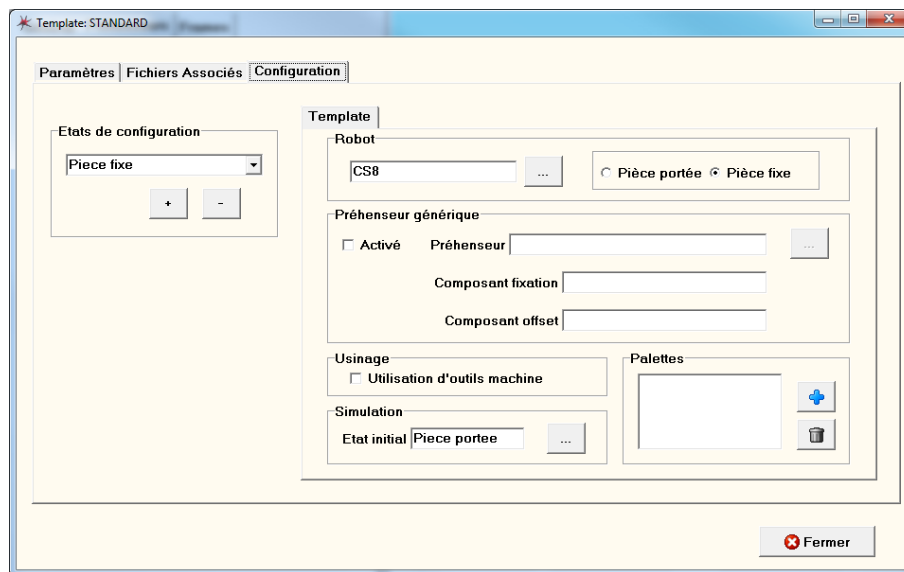


Figure 5.2: Configuration du template

5.2.3 Configuration au niveau de la référence

Au niveau de la référence, les états de configuration contiennent les informations suivantes :

5.2.3.1 Préhenseur

Le nom du *préhenseur* utilisé (préhenseur en pièce portée, broche en pièce fixe). Il doit au préalable être créé dans le gestionnaire de *préhenseur*. Le choix du préhenseur n'est possible que si l'option *Préhenseur générique* n'est pas activé au niveau du template (pièce portée uniquement). Si l'option *Préhenseur générique* est activée, la transformée du composant de fixation doit être spécifiée (ne peut pas être nulle), ainsi que sa représentation 3D. La transformée de l'offset doit être spécifiée, et peut être nulle.

5.2.3.2 Frame

Le nom du *frame* (uniquement un mode pièce fixe ; en *pièce portée*, plusieurs frames peuvent être utilisés et sont définis dans le cycle d'usinage). Il doit au préalable être créé dans le gestionnaire de *frames*.

5.2.3.3 Palette

Le nom de la palette utilisée pour chaque type de palette.

5.2.3.4 Simulation

SYNAPXIS comporte une *bibliothèque d'éléments simulés* 3D qui regroupe notamment les références, les composants de préhenseurs et les *frames*. Les éléments de la bibliothèque sont classés selon des *familles*, des *groupes* et des *composants*.

Les différents états des références simulées appartiennent à la *famille* "Références". Le nom de la référence indique le *groupe*, et le nom de l'état de configuration indique le *composant* lui-même. La représentation 3D d'une référence est donc spécifique à chaque *état de configuration*.

Les templates n'ont pas de représentation 3D. Pour éditer les propriétés du modèle 3D, double-cliquer sur le composant dans la liste ou utiliser le bouton *Editer*. Il est également possible d'accéder directement aux dimensions de l'objet (*Tool simplifié* et *Cylindre* uniquement).

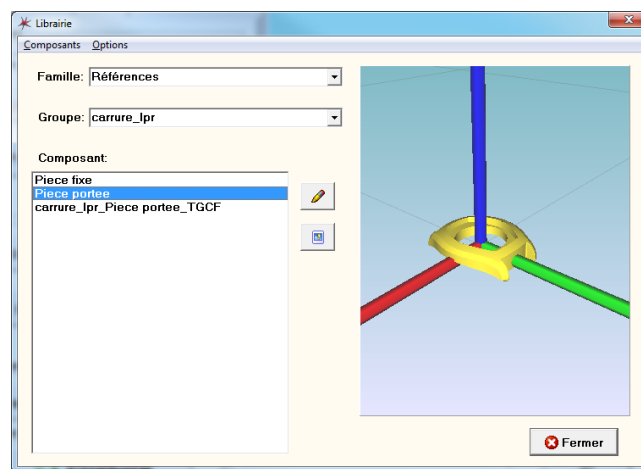


Figure 5.3: Bibliothèque des éléments simulés

Les propriétés qui doivent être éditées pour le modèle 3D de la référence sont les suivantes :

Fichier Modèle Le modèle CAO de la référence doit être enregistré au format VRML ou STL pour pouvoir être importé. Il est judicieux de placer l'origine de la pièce de manière à ce qu'il coïncide avec l'origine de la fixation de la pièce (extrémité du TOOL en pièce portée, origine du FRAME en pièce fixe).

SYNAPXIS utilise comme unité dimensionnelle le millimètre. Si le fichier VRML est généré en mètre, il faut spécifier un facteur d'échelle de 1000. S'il est généré en millimètre, le Facteur d'échelle vaut 1.

Le fichier 3D doit être placé et sélectionné dans le *répertoire de travail* de la simulation (*SimData*).

Offset L'offset permet de modifier l'origine de la pièce importée (origine de la représentation, dépend de l'origine utilisé dans la CAO lors de l'exportation), pour si

nécessaire, le faire coïncider avec son origine réel.

Couleur Il est possible de sélectionner une couleur différente de la couleur par défaut pour la représentation 3D. Toutefois, si un modèle VRML comporte plusieurs couleurs, elles seront remplacées par une seule et unique couleur.

Inversion des vecteurs normaux Si la pièce apparaît sans illumination (noire et sans distinction des arrêtes), il se peut que ses vecteurs normaux soient orientés vers l'intérieur. Il est alors possible d'inverser les vecteurs normaux. Dans l'état actuel, il faut redémarrer l'application pour pouvoir visualiser le changement.

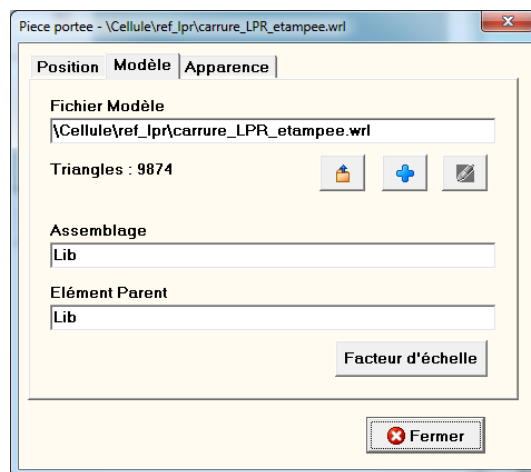


Figure 5.4: Propriétés de l'élément simulé

5.3 Paramètres

5.3.1 Description

Les paramètres sont identifiés par leur nom, qui doit être unique. Leur type peut être booléen, entier signé, réel, string, location cartésienne, location joint, frame ou tableau.

Pour chaque paramètre, il est possible d'utiliser la valeur par défaut, ou "parent", (la valeur du template), ou une valeur propre à la référence.

La valeur définie au niveau du template correspond à la valeur par défaut. Lorsqu'une référence est créée, le template dont elle découle doit être sélectionné (s'il y a un seul template, la sélection est automatique). Cette nouvelle référence comporte alors exactement tous les paramètres du template. Les paramètres sont alors une copie de la valeur par défaut.

Lorsqu'un paramètre est ajouté ou supprimé au niveau du template, la modification est reportée automatiquement sur les références lors de leur ouverture. Il n'est pas possible d'ajouter ou de supprimer un paramètre au niveau d'une référence.

Les paramètres peuvent être édités à 2 niveaux : au niveau du template et au niveau de la référence.

5.3.2 Édition des paramètres au niveau du template

L'édition des paramètres au niveau du template permet de :

- ajouter et supprimer des paramètres au template, et donc à toutes les références qui en sont issues (références enfants).
- éditer la valeur du paramètre au niveau du template. Il s'agit de la valeur par défaut pour les références enfants.
- si ce paramètre existe au niveau du template de base, il est possible d'utiliser sa valeur au niveau du template en spécifiant l'utilisation de la valeur parent. [Options]
- il est possible de forcer l'application de la valeur parent lors de l'importation d'une référence d'une application **SYNAPXIS** vers une autre. Ceci peut être utilisé notamment pour éviter que des paramètres appris et testés sur une machine soient utilisés automatiquement sur une autre machine. [Options]
- afficher ou cacher les paramètres au niveau des références. Pour des raisons de sécurité, il peut être utile d'empêcher l'édition de certains paramètres au niveau des références. [Options]

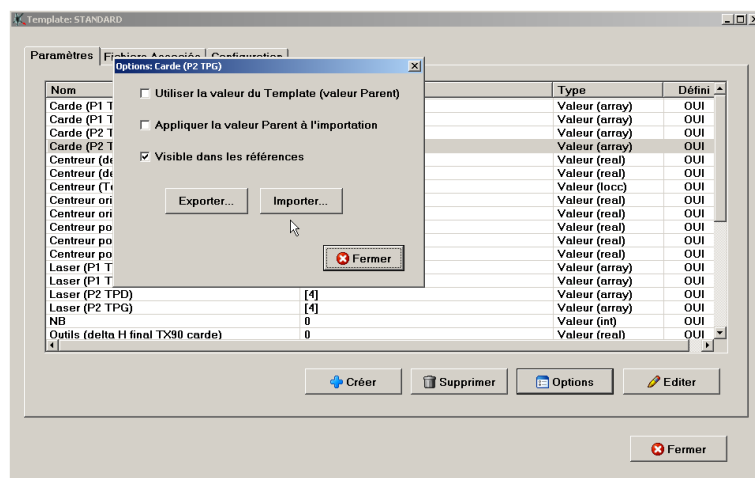


Figure 5.5: Édition des paramètres au niveau du *template*

5.3.3 Édition des paramètres au niveau de la référence

L'édition des paramètres au niveau de la référence permet de :

- éditer la valeur du paramètre au niveau de la référence
- spécifier l'utilisation de la valeur du template, ou valeur parent [Options]

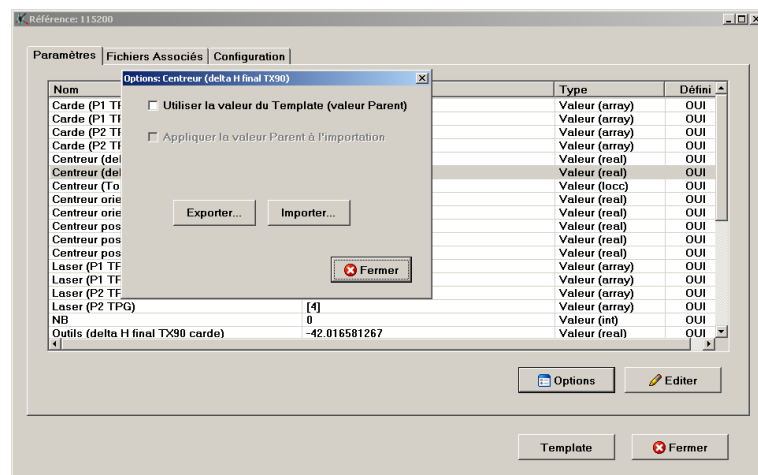


Figure 5.6: Édition des paramètres au niveau de la *référence*

Remarques :

- Pour chaque paramètre, l'éditeur de référence affiche la notion "Défini", qui peut avoir les valeurs suivantes : OUI, NON, (OUI), (NON). "OUI" signifie que la valeur existe. Les parenthèses "()" signifie que c'est la valeur parent qui est utilisée.
- Selon les applications, les paramètres de la référence peuvent également être modifiés via un programme d'apprentissage de la machine.

5.4 Dossiers de références

Lors de la création d'une référence, un dossier dont le nom est identique à celui de la référence est créé. C'est dans ce dernier que le fichier de référence (*.rip), le fichier du cycle d'usinage (*.cip) et les autres fichiers attachés éventuels (comportant également le même nom que la référence, avec une extension différente) sont enregistrés. La gestion de ce dossier et de ses fichiers est faite par **SYNAPXIS**.

Les dossiers de références sont édités depuis les options du gestionnaire de références.

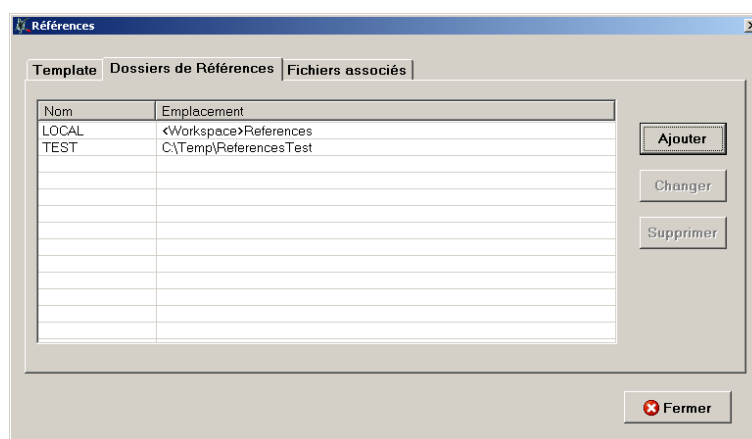


Figure 5.7: Édition des *dossiers de références*

Remarques :

- Le dossier de référence par défaut est le dossier *LOCAL* ; il est créé automatiquement dans le *workspace* courant.
- Un *dossier de référence* ne doit contenir aucun fichier ni dossier qui n'ait été créé par **SYNAPXIS**.
- Les *dossiers de références* peuvent être définis sur plusieurs lecteurs différents.
- Le nom d'une référence doit être unique, et ce sur l'ensemble des *dossiers de références*.

5.5 Création et ouverture d'une référence

Pour créer ou ouvrir une référence, il faut utiliser le dialogue "Ouvrir" depuis la fenêtre principale de **SYNAPXIS**. Celui-ci permet de sélectionner le *dossier de référence*, et affiche les références existantes dans ce dossier.

A la création d'une nouvelle référence, il faut indiquer de quel template elle hérite. Si un seul template existe, l'affectation est faite automatiquement.

Il est possible également de dupliquer une référence, en spécifiant un nouveau nom, ou encore de déplacer une référence d'un *dossier de référence* vers un autre.

Lors de la suppression d'une référence, toutes ses données ainsi que son dossier attribué sont définitivement effacés.

Lorsque la référence est ouverte, elle est ajoutée aux *références courantes*.

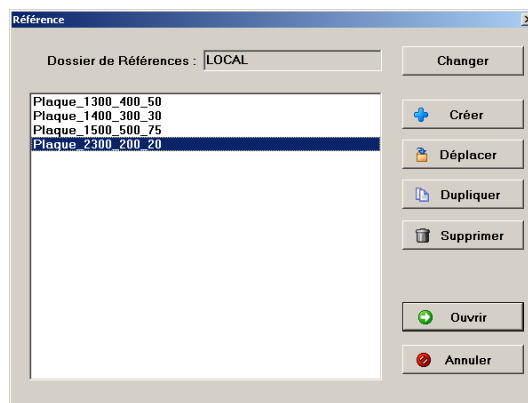


Figure 5.8: Création et ouverture d'une référence

Remarques :

- Lors de sa création, la référence n'existe qu'en termes de dossier et fichiers. C'est seulement lors de son ouverture que ses données sont chargées en mémoire. La référence devient une référence courante. (Une référence n'est utilisable par **SYNAPXIS** que si elle est chargée en mémoire).
- Il n'est pas possible de dupliquer une référence "manuellement" depuis l'explorateur de fichier.
- Le nom d'une référence doit être unique, sur l'ensemble des *dossiers de références*.

5.6 Importation/Exportation

Une référence peut être exportée et importée. Cette fonctionnalité permet de déployer rapidement sur plusieurs machines une référence ainsi que ses différentes données de configuration. Les données de configurations exportées sont celles qui sont liées directement à la référence, et non à la machine. Dans ce sens, pour une configuration en *pièce fixe*, le *tool* et le *frame* sont considérés comme liés à la machine, et ne sont donc pas exportés. Dans le cas d'une configuration en *pièce portée*, le *tool* est directement lié à la référence, tout comme les différentes *palettes*.

5.6.1 Données exportées

Les données suivantes sont exportées dans un fichier d'import/export unique :

- Nom de la référence : celui-ci reste inchangé lors de l'opération d'exportation/importation.
- Le nom du template duquel est dérivée la référence.
- Les paramètres de la références, avec leur valeur et options.
- Les fichiers associés de la référence, avec, le cas échéant, leurs archives existantes.
- Les différentes configurations avec pour chacune d'elle :

- Les données de représentation 3D de la référence : offset, fichier du modèle 3D correspondant, couleur, etc.
- Le *tool* utilisé : nom, ses composants et leur transformée, ainsi que leurs données de représentation 3D (offset, fichier du modèle 3D correspondant, couleur, etc). Uniquement dans le cas d'une configuration en *pièce portée*.
- Les palettes utilisées, avec leur données (positions, entraxes, etc) ainsi que leurs données de représentation 3D. La palette doit être positionnée de manière *relative* par rapport à un frame *parent*. Dans le cas contraire, la palette n'est pas incluse lors de l'exportation. Uniquement dans le cas d'une configuration en *pièce portée*.

5.6.2 Importation

Lors de l'importation d'une référence sur une machine à partir d'un fichier d'import/export, il faut tenir compte des considérations suivantes :

- Le nom d'une référence étant unique, la machine sur laquelle est importée la référence ne doit pas comporter de référence ayant un nom identique.
- Le *template* de la référence considéré lors de l'exportation doit exister (nom identique). La référence est re-synchronisée avec le template. Si des paramètres diffèrent entre les deux (en terme de nom), ce sont ceux du template qui sont imposés à la référence.
- Si le *tool* d'une configuration existe déjà, il est possible de le ré-importer à nouveau ou d'utiliser le tool existant.
- Si le frame *parent* d'une palette n'existe pas (en terme de nom), un autre frame parent doit être sélectionné.

5.7 Fichiers associés

5.7.1 Description

Il est possible de définir plusieurs types de fichiers qui seront associés aux références. Cela permet par exemple d'attacher un fichier d'usinage à une référence pour le polissage, un fichier 3D pour effectuer des calculs d'orientation, etc.

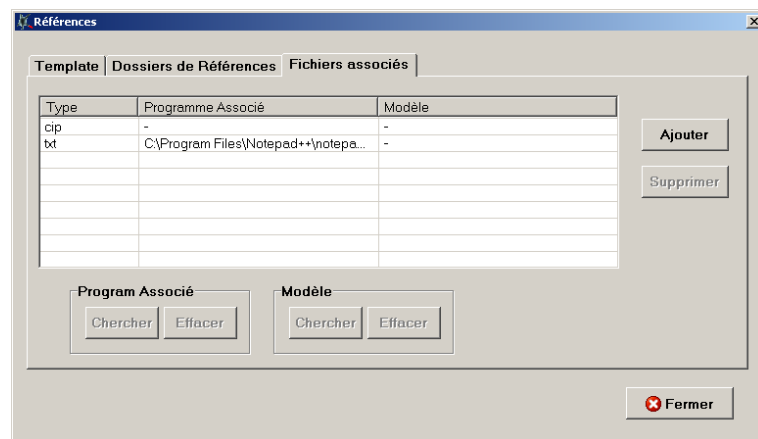


Figure 5.9: Types de fichiers associés aux références

A chaque type de fichier associé peut être attaché un programme associé et un modèle. Par exemple, il est possible de définir un type de fichier **.txt* afin de créer des notes qui accompagnent chaque référence. Si un programme Windows (Notepad.exe, etc) est associé à ce type de fichier, l'ouverture des notes peut être faite automatiquement depuis **SYNAPXIS**. Si un modèle est défini, lors de la première ouverture d'un fichier associé, il est proposé à l'utilisateur de créer ce fichier à partir de ce modèle (formulaire pré-établi, etc).

5.7.2 Fichiers associés et références

L'association d'un type de fichier à une référence est faite au niveau du template. Lorsqu'un type de fichier est associé à un template, il faut indiquer si ce type de fichier peut être dupliqué et archivé.

5.7.2.1 Duplication de fichiers associés

Il est possible d'empêcher la duplication de fichiers associés lorsqu'une référence est dupliquée.

5.7.2.2 Archivage de fichiers associés

Il est possible de réaliser plusieurs archives d'un fichier associé si celui-ci évolue dans le temps.

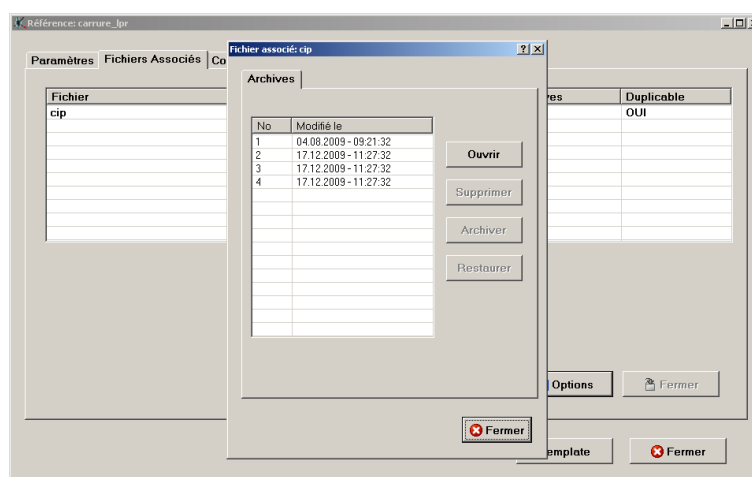


Figure 5.10: Archivage d'un fichier associé à une référence

Lorsque **SYNAPXIS** utilise un fichier associé et que celui-ci est archivé, c'est toujours la dernière archive qui est utilisée. Une archive ancienne peut être restaurée et il est ainsi possible de réutiliser d'anciennes données.

CHAPITRE 6

Préhenseurs

6.1 Introduction

Le terme préhenseur désigne le dispositif qui est monté sur l'axe terminal du robot. Dans le cas d'une configuration en *pièce portée*, le préhenseur est l'élément qui maintient la pièce (ou *référence*). Dans le cas d'une configuration en *pièce fixe*, le préhenseur est l'élément qui maintient l'outil d'usinage.

Dans le cas d'une application multi-robots, chaque *préhenseur* doit être associé à un robot.

6.2 Composants

Un préhenseur est défini par les différents éléments mécaniques qui le composent (les *composants*). En définissant une représentation 3D et une transformée pour chacun des composants, il est possible de représenter leur assemblage formant le *préhenseur* dans la vue 3D et de calculer sa transformée totale.

Un composant ne peut pas être supprimé s'il est utilisé dans la composition d'un préhenseur existant. Il peut par contre être renommé, la modification est alors répercutée au niveau des préhenseurs qui l'utilisent.

6.2.1 Transformée

La transformée d'un composant correspond aux translations (x, y, z) et aux rotations (rx, ry, rz) entre son *origine* et son *extrémité*. Son *origine* coïncide avec l'*extrémité* du composant parent. Dans le cas où il n'y a pas de composant parent, c'est le flasque robot qui est considéré.

La transformée est nécessaire pour le calcul de la transformée totale du préhenseur, ainsi que pour le positionnement de l'élément enfant du composant dans la vue 3D.

Sil la transformée est *modifiable*, celle-ci peut être modifiée depuis un programme (instruction *machineToolSetPartTrans*) ou par le gestionnaire de référence dans le cas de l'utilisation d'un préhenseur *générique*.

6.2.2 Représentation 3D

Afin de pouvoir représenter l'ensemble du préhenseur dans la vue 3D, chaque composant doit être défini dans la *bibliothèque d'éléments simulés* (voir chapitre 5.2.3.4). La *famille* correspond à *tools*, le *groupe* est donné par la *catégorie* du composant et le *composant* par son nom. L'*offset* doit être indiqué de manière à faire coïncider l'*origine* du modèle 3D avec celui du composant.

6.2.3 Catégories

Les composants peuvent être classés selon des *catégories de composants* afin d'améliorer leur classification.

6.3 Définitions

6.3.1 Liste de composants

La composition du préhenseur est établie par la liste de ses *composants*. L'ordre de la liste est déterminant, le premier composant étant celui dont le parent est le flasque du robot, et le dernier celui dont l'*extrémité* coïncide avec l'extrémité utile du préhenseur.

6.3.2 Transformée totale

La transformée totale du préhenseur est l'addition matricielle des transformées de ses composants, dans l'ordre de la liste.

Si la transformée du préhenseur est amenée à varier au cours du processus (mesure laser, etc), il est possible de rajouter un composant dont la transformée est mise à jour au travers du programme de mesure (instruction *machineToolSetPartTrans*, le composant en question doit alors être *modifiable*).

6.3.3 Utilisation d'outils

En pièce fixe, des outils peuvent être appliqués aux préhenseurs (usinage avec broche, soudure, découpe, etc.). Il faut pour cela spécifier l'utilisation d'outils pour le préhenseur voulu. Il est alors considéré comme une *unité* à laquelle des outils peuvent être appliqués (voir chapitre 8).

6.3.4 Représentation 3D

Lorsqu'un préhenseur est appliqué au robot simulé par le biais de l'interface des préhenseurs, sa représentation 3D est ajoutée à l'environnement courant, dans un assemblage spécifique lié au flasque du robot. Chaque composant est positionné en fonction de ses paramètres de représentation 3D (bibliothèque) et des transformées des composants parents.

Un système de coordonnées est dessiné à l'extrémité du dernier composant, de manière à représenter l'*extrémité* finale et exacte du préhenseur, qui correspond au

repère *tool* courant du robot. Dans le cas d'une configuration en pièce portée, celui-ci coïncide avec l'*origine* de la référence.

6.4 Préhenseur courant du robot

Le préhenseur sélectionné peut être appliqué au robot, simulé ou réel, en fonction du mode courant.

La transformée du préhenseur courant du robot peut être modifiée. Le préhenseur courant est mis à jour périodiquement (il peut être modifié lors de l'exécution d'un programme). Dans le cas où sa transformée correspond à un des préhenseurs existants, c'est son nom qui est indiqué. Dans le cas contraire, c'est la transformée qui est indiquée. Dans le cas où plusieurs préhenseurs ont des transformées identiques, le nom n'est pas affiché afin de ne pas induire l'opérateur en erreur.

CHAPITRE 7

Frames

7.1 Introduction

Les *frames* sont des *positions cartésiennes* dont la translation et l'orientation sont définies sur la base de plusieurs points. Ceux-ci doivent être appris dans l'environnement *réel*.

Ils sont destinés à positionner des éléments péri-robotiques de manière à pouvoir calculer des positions qui leur sont relatives : palettisation, usinage (*pièce fixe* et *pièce portée*), etc.

Dans le cas d'une application multi-robots, chaque *frame* doit être associé à un robot.

7.2 Généralités

7.2.1 Point de transition

Un point de transition peut être associé au frame, de manière à pouvoir automatiser son approche dans les *cycles d'usinage*.

7.2.2 Outils

Dans une configuration en pièce portée, un frame peut être défini comme *support d'outils*. Cela indique qu'il est l'élément mécanique par rapport auquel sont fixés des outils d'usinage : tampon de polissage, bande de meulage, etc.

Un outil est par définition une pièce d'usure, qui peut être renouvelée (dimensions nominales), et dont les dimensions peuvent évoluer au cours de son utilisation (usure). Il est positionné de manière répétitive par rapport au *frame*.

L'affectation d'un outil à un *frame* est déterminée par la référence (voir chapitre 8). Ainsi, d'une référence à l'autre, un même outil peut être monté sur différents *frames*. Il est nécessaire, d'un point de vue utilisation et représentation, de spécifier de quel côté du frame l'outil est monté par rapport à la position zéro de l'axe *z*. L'axe *z* du frame correspond toujours à la normal (*frame plan* et *palette*) ou à l'axe de rotation (*frame cylindrique*).

7.2.3 Simulation - représentation 3D

Un frame peut être représenté dans l'environnement 3D, dans un assemblage spécifique lié à la base du robot. Il est positionné en fonction de sa *position résultante* et de ses paramètres de représentation 3D. Il est possible d'utiliser la représentation par défaut (plan pour un frame plan, cylindre pour un frame cylindrique, palette pour un frame palette) ou de définir une représentation spécifique (bibliothèque : objet simplifié ou fichier *.wrl, *.stl).

Dans le cas où un frame *enfant* à une représentation par défaut et que son *parent* utilise une représentation spécifique, c'est cette dernière qui est affichée.

7.2.4 Position

La position cartésienne d'un *frame* est donnée par rapport au repère de base du robot (repère *world*), qui est le repère absolu. La *position résultante* du *frame* correspond à l'addition matricielle des éléments décrits ci-dessous :

7.2.4.1 Frame parent

Un frame peut être positionné de manière relative à un autre frame, le *frame parent*. En fonction des assemblages mécaniques de la machine (éléments goupillés, etc), cette option permet de minimiser le nombre de frames appris avec le robot : si un frame parent est déplacé (démontage, etc), il suffit de le ré-apprendre lui seul pour que tous les frames qui en dépendent (*frames enfants*) soient à nouveau définis correctement.

Les frames définis comme *supports d'outils* ainsi que les *frames palette* ne peuvent pas être utilisés en tant que *parent* pour un autre frame.

7.2.4.2 Décalage avant

Un décalage peut être spécifié entre le repère parent (repère *world* si aucun parent n'est défini) et le frame lui-même.

7.2.4.3 Frame

Le *frame* est la position calculée à partir des différents points appris avec le robot. Ceux-ci doivent obligatoirement être définis, sans quoi la position du frame est *invalidée*.

Si toutefois un frame doit être défini théoriquement et sans points appris (simulation pure, etc), les valeurs du frame peuvent être implémentées de manière à résulter en un frame *nul* ; la position du frame est alors simplement renseignée dans un des décalages.

7.2.4.4 Décalage après

Un décalage peut être ajouté après le frame. Cela est par exemple nécessaire lorsqu'un *outil de calibration* est positionné sur le frame lors de son apprentissage et qu'il est nécessaire d'annuler le décalage qu'il induit lui-même par rapport à l'origine mécanique du frame.

7.2.4.5 Décalage utilisateur

Un décalage *utilisateur* peut finalement être additionné à la valeur résultante du frame. La valeur de ce décalage peut notamment être définie par une instruction macro. Ceci permet par exemple de décaler un frame pendant son utilisation via l'interface de production.

7.3 Frame plan

Un frame plan est utilisé pour définir une position cartésienne à partir d'une surface plane. Les informations suivantes permettent de définir les 6 coordonnées du frame :

Origine Position d'origine du plan XOY

Point X Position X du plan XOY

Point Y Position Y du plan XOY

Position Position optionnelle, pouvant se trouver en dehors du plan XOY .

Direction principale Flag indiquant quel vecteur OX ou OY est utilisé pour orienter le frame.

Le calcul de la valeur du frame est réalisé comme suit : les coordonnées X, Y, Z du frame sont définies par les coordonnées X, Y, Z de *Origine*, ou de *Position* si cette valeur est utilisée. Les rotations RX, RY sont définies par le produit vectoriel $OX \otimes OY$; le sens du vecteur normal respecte ainsi la règle *de la main droite*. Finalement, la rotation RZ est fixée, en fonction du flag *Direction principale*, de manière à aligner soit le vecteur X du frame parallèlement au vecteur OX , soit le vecteur Y du frame parallèlement au vecteur OY .

7.4 Frame cylindrique

Un frame cylindrique est utilisé pour définir une position cartésienne à partir du périmètre et d'une face de cylindre. Le repère d'un frame cylindrique est positionné sur l'axe de rotation, le vecteur Z étant toujours confondu avec l'axe de rotation.

7.4.1 Position et orientation

3 points A, B, C situés sur la face du cylindre permettent de définir un vecteur normal à celle-ci, parallèle à l'axe de rotation du cylindre, fixant les rotations RX et RY .

3 points A, B, C sur le périmètres, projetés sur la face selon l'axe du cylindre, permettent de déterminer la position de son centre, fixant les translations X, Y, Z .

7.4.1.1 Frame orienté selon Y - Orientation libre

Dans le cas où le frame est *orienté selon y*, le sens du vecteur normal dépend du sens d'apprentissage des 3 points sur la face du cylindre (sens horaire ou anti-horaire). La rotation finale RZ est fixée de manière à ce que le vecteur Y du frame soit orienté en direction du point d'orientation optionnel.

7.4.1.2 Orientation standardisée

Un frame *standardisé* est orienté selon la règle suivante : le produit vectoriel de l'axe du cylindre Z et de l'axe Z du repère *world* donne le vecteur Y du repère ; si nécessaire, son sens est modifié de manière à être toujours orienté vers le centre du repère *world*, ce qui implique que le vecteur Z est également modifié de manière à toujours être en *rotation* autour de l'axe Z du repère *world*, dans le sens trigonométrique. Le vecteur X est par conséquent orienté vers le bas.

L'orientation standardisée n'est pas valable pour des frames cylindriques dont l'axe est parallèle à celui du repère *world*, qui implique un produit vectoriel nul.

7.4.2 Diamètre et limites

Le diamètre du cylindre *mesuré* par l'intermédiaire des 3 points du périmètre est indiqué. Sa comparaison avec la valeur effective du cylindre utilisé pour l'apprentissage des points donne une indication sur la précision des mesures. Le diamètre du cylindre qui doit être considéré peut être édité s'il diffère de celui de l'outil d'apprentissage.

Des points *limites* peuvent être utilisés pour déterminer les plages de *largeur* et d'*élévation* du frame cylindrique.

7.5 Palette

Un frame *palette* est un frame plan qui permet de définir un nombre donné d'emplacements arrangés en lignes et colonnes, respectant des entraxes réguliers. La position de l'origine d'un frame *palette* est identique à celle d'un frame *plan*.

Généralement, la position d'un frame *palette* est définie de manière à ce que le vecteur Z pointe vers l'intérieur de la palette.

7.5.1 Emplacements

Le nombre d'emplacements est donné par le produit du nombre de lignes et du nombre de colonnes. Les colonnes sont parallèles à l'axe Y du frame, et les lignes parallèles à l'axe X du frame. Le premier emplacement se situe à l'origine, les suivants étant répartis en lignes.

7.5.2 Entraxes

Les différentes positions sont calculées en fonction du numéro de la pièce, qui indique le numéro de la ligne et de la colonne correspondante, ainsi que des entraxes lignes (dx) et colonnes (dy).

Si les points X, O, Y du frame plan correspondent aux emplacements extrêmes de la palette, ils peuvent être utilisés pour calculer la valeur des entraxes lignes et colonnes.

7.5.3 Offset

Si les emplacements de la palette sont arrangés en *quinconce*, un offset peut être défini selon les lignes ou les colonnes.

7.5.4 Pick

Il est possible de définir un offset qui est ajouté sur la position de l'emplacement, par exemple dans le cas où l'origine des pièces palettisées n'est pas orientée selon le *frame palette*.

7.5.5 Positions des emplacements quelconques

Dans le cas où une palette ne peut être définie sur la base d'un calcul de lignes/colonnes et entraxes, les positions de chacun de ses emplacements doivent être définies dans un tableau, relativement à l'origine du *frame palette*.

CHAPITRE 8

Outils

8.1 Introduction

Les outils sont les pièces d'usure utilisées dans le cycle d'usinage, et destinées à l'exécution de l'usinage (polissage, fraisage, meulage, etc). Pour une configuration en *pièce portée*, les outils sont positionnés sur les *frames* ; pour une configuration en *pièce fixe*, les outils sont positionnés sur le préhenseur.

Les frames où sont placés les outils doivent être définis en tant que *support d'outils*, et sont également appelés *unités*.

8.2 Outils et références

La *répartition des outils utilisés* sur les unités est mémorisée par la référence. Le *taux d'usure* est également mémorisé par référence et pour chaque outils.

Les autres paramètres affichés dans la liste sont propres aux outils et indépendants des références :

Usure L'usure correspond à la variation de rayon entre le rayon nominal et le rayon courant (outil cylindrique). Elle est calculée sur la base du nombre de pièces usinées par l'outil (progression linéaire ou par tables de valeurs).

Offset usure Variation additionnelle du rayon, déterminée par l'opérateur afin de compenser une éventuelle dérive de l'usure calculée.

Diamètre actuel Diamètre courant de l'outil, calculé en fonction de son diamètre nominal et de l'*usure* + *offset usure* (usure totale).

Diamètre nominal Diamètre de l'outil lorsque l'usure totale vaut zéro.

Pièces Nombre total de pièces usinées par l'outil.

8.2.1 Remise à zéro

Lorsqu'un outil est remplacé dans la machine, ses paramètres d'usure doivent être remis à zéro : nombre total de pièces, usure et offset usure.

8.2.2 Application de l'usure

L'usure est appliquée automatiquement à chaque pièce lors de l'exécution du programme de production. Il est également possible de l'appliquer manuellement, pour l'outil sélectionné ou pour tous les outils, en fonction de la référence sélectionnée.

8.3 Outils

8.3.1 Plan

Un outil plan est défini simplement par sa hauteur. L'usure et l'offset usure se soustraient directement à cette grandeur. Les valeurs de *largeur* et d'*offset* ne sont actuellement pas considérées.

8.3.2 Disque

Un outil disque est un cylindre défini par son diamètre nominal, ainsi que ses largeurs ; celles-ci sont considérées selon l'axe Z du frame :

Largeur gauche Largeur utile de l'outil à gauche ($+Z$) de son origine.

Largeur droite Largeur utile de l'outil à droite ($-Z$) de son origine.

Offset gauche Largeur inutilisée de l'outil à gauche ($+Z$) de son origine.

Offset droite Largeur inutilisée de l'outil à droite ($-Z$) de son origine.

Ces différentes largeurs permettent de positionner un outil sur une unité dont l'utilisation d'outil est définie pour le côté $+Z$ ou le côté $-Z$. L'offset représente l'offset mécanique entre le $Z0$ du frame et le bord de l'outil. La largeur permet de déplacer l'origine de l'outil vers son centre.

Le sens *gauche/droite* correspond à un outil monté sur un frame cylindrique orienté de manière *standard*, vu par le robot. Un outil disque est utilisé radialement, sur son périmètre (Z), l'usure se portant au rayon.

8.3.3 Disque profilé

L'outil *disque profilé* correspond à la révolution d'une trajectoire de profile autour de l'axe de l'outil (*frame cylindrique*).

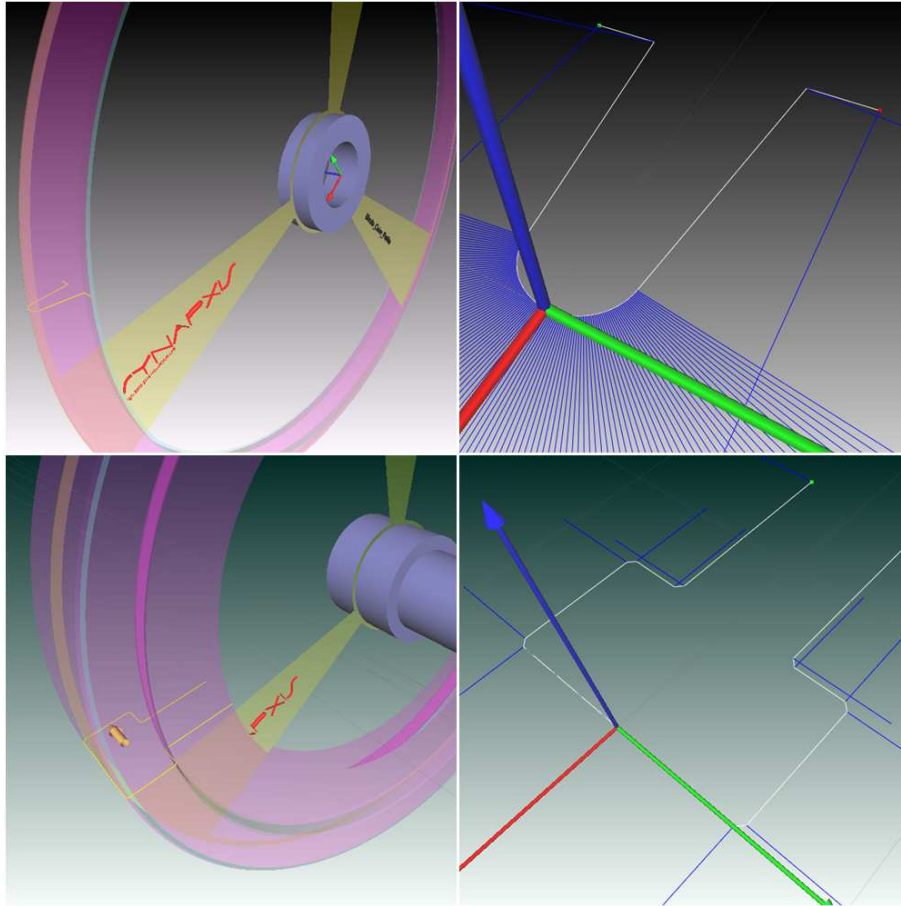


Figure 8.1: Outils profilés

- Le profile de l'outil est édité dans le plan $X-Y$. La coordonnée Z des points du profile doit être nulle.
- Le sens de la trajectoire de profile va de $-Y$ vers $+Y$.
- Les vecteurs Z doivent être normaux au profile, et confinés dans le plans $X-Y$.
- L'origine du profile correspond au point où $Y = 0$. Si le profile coupe l'axe X plusieurs fois, c'est la première intersection, en partant depuis $-Y$, qui est considérée.
- La coordonnée X est additionnée au rayon de l'outil. Il est judicieux de décaler le profil de manière à ce que $X = Y = 0$, de manière à avoir le diamètre *nominal* à l'origine.
- La cote minimale du profile selon l'axe $-X$ ne doit pas être plus petite que le

rayon minimal de l'outil.

- Lors de l'utilisation d'un outil profilé, le paramètre *Frame/X* correspond à la position linéaire le long de la trajectoire du profile. Si le paramètre $X = 0$, le point de contact est à l'origine du profile.
- Les largeurs de *limites* et d'*offsets*, ainsi que l'usure fonctionnent comme pour un outil cylindrique (*Disque*).

8.3.4 Cylindre

Un outil cylindre représente une fraise d'usinage cylindrique ou tout autre outil dont la forme se rapporte à un cylindre, et qui est utilisé sur sa hauteur (Z). L'usure est considérée sur la hauteur.

8.3.5 Sphère

Identique à l'outil cylindre. Sa représentation se termine par une demi-sphère.

8.3.6 Cône

Identique à l'outil cylindre. Sa représentation se termine par un cône.

8.4 Paramètres des outils

8.4.1 Alarmes

Des seuils d'alarmes peuvent être définis pour chaque outil en terme d'usure maximale et en terme de nombre de pièces usinées maximal. Lors de l'exécution du programme de production, ces seuils d'alarme peuvent être testés, et le cas échéant, une information peut être transmise à l'opérateur.

8.4.2 Limites

Des seuils limites doivent être définis pour chaque outil en terme d'usure maximale et en terme de nombre de pièces usinées maximal ; ils correspondent à l'usure maximale acceptable pour l'outil et son unité. Ils doivent permettre d'éviter des collisions entre le robot et le frame. Lors de l'exécution du programme de production, ces seuils sont testés, et le cas échéant, la production est stoppée.

8.4.3 Vitesse de l'outil

Afin de garder une efficacité nominale en fonction de l'usure, une compensation de vitesse de coupe peut être définie pour un outil.

8.4.4 Usure

Afin de garder une efficacité nominale en fonction de l'usure, une compensation du rayon (ou de la hauteur) sur l'outil peut être définie pour un outil. Cela permet de conserver un *appui* constant.

Table des figures

1.1	Fenêtre principale	1
2.1	Edition des vues	5
4.1	Vitesses du robot	11
4.2	Edition des transitions	13
5.1	Héritage des templates et références	16
5.2	Configuration du template	18
5.3	Bibliothèque des éléments simulés	19
5.4	Propriétés de l'élément simulé	20
5.5	Édition des paramètres au niveau du <i>template</i>	21
5.6	Édition des paramètres au niveau de la <i>référence</i>	22
5.7	Édition des <i>dossiers de références</i>	23
5.8	Création et ouverture d'une référence	24
5.9	Types de fichiers associés aux références	26
5.10	Archivage d'un fichier associé à une référence	27
8.1	Outils profilés	41