

HUGO DIONNE

**STRATÉGIES D'ASSEMBLAGE ET DE
DÉSASSEMBLAGE DANS UN CONTEXTE DE
DÉVELOPPEMENT DURABLE**

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de maîtrise en génie mécanique
pour l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES ET DE GÉNIE
UNIVERSITÉ LAVAL
QUÉBEC

2006

Résumé

Ce mémoire traite de stratégies d'assemblage et de désassemblage dans un contexte de maintenance et de valorisation de produits en fin de cycle de vie. L'objectif visé est de proposer une stratégie de désassemblage permettant d'effectuer rapidement et économiquement les actions de maintenance ou de récupérer efficacement des composants d'un produit. Le mémoire analyse les principales stratégies de désassemblage proposées dans la littérature . Les méthodes de résolution du désassemblage sont étudiées en considérant, les besoins, les types de produits et le niveau de désassemblage. Parmi les méthodes examinées, on retrouve les réseaux de petri, les algorithmes génétiques et des algorithmes basés sur la géométrie des composants. Ces démarches se traduisent en outils d'aide à décision pour le développement et la mise en œuvre de stratégies de désassemblage. Finalement, ces différentes stratégies étudiées sont classées et comparées en fonction de leurs capacités, de leurs limites, et des différents objectifs.

Table des matières

1	<i>Introduction</i>	1
1.1	Problématique	1
1.2	1.2 Objectifs.....	3
1.3	Méthodologie	4
1.4	Structure de l'étude	5
2	<i>Problématique du désassemblage</i>	10
2.1	Pourquoi désassembler?.....	10
2.2	Qu'est-ce qu'on désassemble?.....	10
2.2.1	<i>Cycle de vie des produits</i>	11
2.2.2	<i>Le coût d'un cycle de vie de produit</i>	12
2.3	En quoi consiste le désassemblage?	14
3	<i>Procédure pré-désassemblage</i>	16
3.1	Analyse des besoins	16
3.2	Analyse de fin de vie du produit à désassembler.....	17
3.2.1	<i>Gestion de la récupération de produits</i>	17
3.2.2	<i>Quantification des niveaux de désassemblage</i>	20
3.3	Évaluation de la complexité du produit.....	25
3.3.1	<i>Évaluation de la complexité</i>	25
3.3.2	<i>Mesure de la complexité de la structure du produit</i>	27
3.4	Choix de la stratégie de désassemblage.....	31
3.5	Collecte d'information nécessaire	33
3.5.1	<i>L'outil de collecte de données</i>	33
3.5.2	<i>La carte de pointage DEI</i>	34
4	<i>Théorie et outils liés aux stratégies</i>	39
4.1	Représentations du produit	39
4.2	Outils de base pour l'utilisation des stratégies	40
4.2.1	<i>Matrice d'interférence et de désassemblage</i>	40
4.2.2	<i>Arbre de désassemblage</i>	49
5	<i>Stratégies de base</i>	52
5.1	Séquence de désassemblage avec ou sans composant défectueux	56

5.1.1	<i>Methodologie de la strategie</i>	56
5.1.2	<i>Exemple lie a la methode</i>	61
5.2	Les abstractions geometriques	63
5.3	L'apprentissage et le raisonnement geometrique	72
5.4	Les relations de priorite entre les composants	77
6	<i>Strategies avec les reseaux de Petri</i>	82
6.1	Methodes des reseaux de Petri.....	85
6.2	Reseau de Petri avec relations de priorite ET/OU complexes.....	87
6.3	Le reseau de Petri expert stochastique colore et augmente	99
6.3.1	<i>Description de la methode EECSPN</i>	99
6.3.2	<i>Exemple complet dans le cas d'une monture de lunette</i>	109
7	<i>Strategies basees sur l'aspect environnemental et economique</i>	127
7.1	Evaluation economique des operations de desassemblage.....	129
7.1.1	<i>Partie analyse environnementale</i>	129
7.1.2	<i>Partie optimisation de la sequence de desassemblage</i>	138
7.2	Reseau logique de desassemblage	149
7.3	Algorithme genetique avec aspect economique et environnemental.....	157
8	<i>Strategies pour l'entretien</i>	165
8.1	Algorithme genetique pour la maintenance	166
8.2	Methode du desassemblage selectif global.....	175
9	<i>Strategie dans le cas d'une ligne de desassemblage</i>	181
9.1	Approche d'optimisation avec presence d'echec de taches.....	182
9.2	Exemple d'application pour l'approche.....	196
10	<i>Lien entre la conception et le desassemblage de produit</i>	207
10.1	Analyse de la conception pour aider le desassemblage	207
10.1.1	<i>Analyse du desassemblage pour un retour sur la conception</i>	207
10.1.2	<i>Retour sur la conception pour le recyclage</i>	214
10.2	Conception environnementale	219
11	<i>Conclusion</i>	225
11.1	Resume de l'analyse comparative sur les strategies	226
11.2	Perspectives	229
12	<i>References</i>	231

Annexes

<i>Annexe A :</i>	<i>Algorithme de Dijkstra.....</i>	<i>237</i>
<i>Annexe B :</i>	<i>Document de collecte de données.....</i>	<i>239</i>
<i>Annexe C :</i>	<i>Nomenclature Réseau de Pétri.....</i>	<i>242</i>
<i>Annexe D :</i>	<i>Nomenclature Ligne de désassemblage.....</i>	<i>245</i>

Index des tableaux

<i>Tableau 3.1 :</i>	<i>Les options de récupération de produit.....</i>	<i>20</i>
<i>Tableau 3.2 :</i>	<i>Nombre total de séquences de désassemblage.....</i>	<i>26</i>
<i>Tableau 3.3 :</i>	<i>Stratégies en fonction des besoins et des paramètres du produit.....</i>	<i>32</i>
<i>Tableau 3.5 :</i>	<i>La carte de pointage DEI.....</i>	<i>38</i>
<i>Tableau 5.1 :</i>	<i>Sommaire de la collecte de données (IBM PS/2 Model 30).....</i>	<i>61</i>
<i>Tableau 5.2 :</i>	<i>Résultat de l'exemple (IBM PS/2 Model 30).....</i>	<i>62</i>
<i>Tableau 6.1:</i>	<i>Comparaison des méthodes du réseau de Petri.....</i>	<i>84</i>
<i>Tableau 6.2 :</i>	<i>Temps de désassemblage, outils, directions pour l'exemple.....</i>	<i>98</i>
<i>Tableau 6.3 :</i>	<i>Description d'un arbre ET/OU.....</i>	<i>110</i>
<i>Tableau 6.4 :</i>	<i>Description du réseau de Petri.....</i>	<i>110</i>
<i>Tableau 6.5 :</i>	<i>Évolution des places dans le modèle EECSPN.....</i>	<i>116</i>
<i>Tableau 6.6 :</i>	<i>Évolution des symboles dans le modèle EECSPN.....</i>	<i>117</i>
<i>Tableau 6.7 :</i>	<i>Étape initiale et son évolution.....</i>	<i>118</i>
<i>Tableau 6.8 :</i>	<i>Détermination du chemin optimal.....</i>	<i>119</i>
<i>Tableau 6.9 :</i>	<i>Description des places et des transitions pour le haut-niveau.....</i>	<i>121</i>
<i>Tableau 6.10 :</i>	<i>Description des places et des transitions pour le bas-niveau.....</i>	<i>125</i>
<i>Tableau 7.1 :</i>	<i>Table de décision pour la récupération et le rejet.....</i>	<i>138</i>
<i>Tableau 7.2 :</i>	<i>Structure de données pour un nœud et un arc.....</i>	<i>154</i>
<i>Tableau 7.3 :</i>	<i>Exemple de structure de magasin des nœuds.....</i>	<i>156</i>
<i>Tableau 8.1 :</i>	<i>État de départ étalant les séquences possibles.....</i>	<i>173</i>
<i>Tableau 8.2 :</i>	<i>Identification du sous-assemblage.....</i>	<i>173</i>
<i>Tableau 9.1 :</i>	<i>Base d'apprentissage du produit étudié.....</i>	<i>196</i>
<i>Tableau 9.2 :</i>	<i>États du ISN (et SN).....</i>	<i>197</i>
<i>Tableau 9.3 :</i>	<i>SDP du WSN.....</i>	<i>197</i>
<i>Tableau 9.4 :</i>	<i>Compilation des coûts de tous les SDP.....</i>	<i>205</i>
<i>Tableau 10.1 :</i>	<i>Guide de conception développé par la conception axiomatique.....</i>	<i>224</i>

Tableau 11.1 :	Caractéristiques particulières des stratégies.....	227
Tableau 11.2 :	Liens fonctionnels entre les stratégies et les différents éléments étudiés ..	228

Index des figures

Figure 1.1 :	Structure hiérarchique utilisée pour le document.....	7
Figure 1.2 :	Positionnement de génération de séquences de désassemblage.....	8
Figure 2.1 :	Cycle de vie d'un produit	11
Figure 2.2 :	Coûts du cycle de vie d'un produit	12
Figure 2.3 :	Coûts cumulés d'un cycle de vie.....	13
Figure 2.4 :	Représentation et actions de désassemblage.....	15
Figure 3.1 :	Chaîne de montage intégrée.....	18
Figure 3.2 :	Contrôleur de décision pour un système de désassemblage	22
Figure 3.3 :	Comparaison de la complexité de trois types de produits.....	30
Figure 4.1 :	Ajout d'un sous nœud.....	43
Figure 4.2 :	Génération de sous nœuds avec contrainte.....	44
Figure 4.3 :	Génération d'un sous nœud avec contrainte	45
Figure 4.4 :	Algorithme de construction des matrices.....	46
Figure 4.5 :	Montant de dos télescopiques du fauteuil Prima d'Orthofab.....	47
Figure 4.6 :	Représentation : nœuds et arc	49
Figure 4.7 :	Exemple d'arbre de désassemblage d'un produit	50
Figure 5.1 :	Génération d'une séquence de désassemblage	59
Figure 5.2 :	Étape 3 avec présence de composants défectueux	60
Figure 5.3 :	Algorithme pour construire le graphique topologique d'assemblage.....	64
Figure 5.4 :	Algorithme pour calculer la facilité de séparation	65
Figure 5.5 :	Classification des composants limites	66
Figure 5.6 :	Algorithme de détermination des composants limites.....	67
Figure 5.7 :	Algorithme de détermination des composants limites.....	68
Figure 5.8 :	Algorithme de détermination des arêtes et faces personnelles.....	69
Figure 5.9 :	Faces personnelles et générées.....	69
Figure 5.10 :	Algorithme pour calculer $K_{i,A}$	70
Figure 5.11 :	Algorithme pour l'identification de la validité d'assemblage	71
Figure 5.12 :	Procédure de planification du désassemblage	73

<i>Figure 5.13 : Planification du désassemblage du composant.....</i>	<i>75</i>
<i>Figure 5.14 : Procédure de désassemblage partiel d'un composant.....</i>	<i>76</i>
<i>Figure 5.15 : Reconstitution d'un sous-assemblage</i>	<i>78</i>
<i>Figure 5.16 : Schéma et graphique d'une télécommande.....</i>	<i>80</i>
<i>Figure 5.17 : Actions pour chacun des sous-ensembles</i>	<i>81</i>
<i>Figure 6.1 : Diagramme pour la génération de la séquence de désassemblage.....</i>	<i>88</i>
<i>Figure 6.2 : Exemple de produit</i>	<i>88</i>
<i>Figure 6.3(a) : Algorithme pour générer la matrice de priorité de désassemblage.....</i>	<i>90</i>
<i>Figure 6.3(b) : Algorithme pour générer la matrice de priorité de désassemblage.....</i>	<i>91</i>
<i>Figure 6.4(a) : Algorithme pour générer réseau de Petri de désassemblage.....</i>	<i>92</i>
<i>Figure 6.4(b) : Algorithme pour générer réseau de Petri de désassemblage.....</i>	<i>93</i>
<i>Figure 6.5 : Désassemblage avec réseau de Petri (DPN) du produit</i>	<i>94</i>
<i>Figure 6.6(a) : Algorithme de l'arbre réduit d'accessibilité.....</i>	<i>96</i>
<i>Figure 6.6(b) : Algorithme de l'arbre réduit d'accessibilité.....</i>	<i>97</i>
<i>Figure 6.7 : Graphique ET/OU pour l'assemblage d'une monture de lunette.....</i>	<i>109</i>
<i>Figure 6.8 : Procédure d'assemblage/désassemblage avec un réseau de Petri.....</i>	<i>111</i>
<i>Figure 6.9 : Modèle EECSPN pour l'assemblage d'une monture de lunette.....</i>	<i>112</i>
<i>Figure 6.10 : Position des pièces dans le modèle EECSPN à la première étape.....</i>	<i>113</i>
<i>Figure 6.11 : Position des pièces dans le modèle EECSPN à la deuxième étape.....</i>	<i>114</i>
<i>Figure 6.12 : Position des pièces dans le modèle EECSPN à la troisième étape.....</i>	<i>115</i>
<i>Figure 6.13 : EECSPN de haut-niveau</i>	<i>120</i>
<i>Figure 6.14 : EECSPN de bas-niveau</i>	<i>124</i>
<i>Figure 7.1 : Maximisation potentielle pour le MRO.....</i>	<i>131</i>
<i>Figure 7.2 : Importance des séquences de désassemblage</i>	<i>135</i>
<i>Figure 7.3 : Représentation de l'arbre de désassemblage.....</i>	<i>140</i>
<i>Figure 7.4 : Groupe pour la compatibilité des matériaux.....</i>	<i>141</i>
<i>Figure 7.5 : Arbre réduit de désassemblage</i>	<i>142</i>
<i>Figure 7.6 : Algorithme pour le réseau incident directeur positif.....</i>	<i>152</i>
<i>Figure 7.7 : Algorithme pour le coût du méta problème d'un nœud.....</i>	<i>152</i>
<i>Figure 7.8 : Algorithme directeur positif pour le coût minimum d'un nœud.....</i>	<i>153</i>
<i>Figure 7.9 : Algorithme de recherche inversée.....</i>	<i>155</i>
<i>Figure 7.10 : Exemple pour le réseau logique de désassemblage.....</i>	<i>156</i>
<i>Figure 7.11 : Algorithme d'une séquence de désassemblage optimale</i>	<i>158</i>
<i>Figure 7.12 : Exemple pour l'utilisation de l'algorithme génétique</i>	<i>159</i>

<i>Figure 7.13 :</i>	<i>Arbre de désassemblage pour le module R DOOR</i>	<i>160</i>
<i>Figure 8.1 :</i>	<i>Génération des séquences de désassemblage pour la maintenance.....</i>	<i>168</i>
<i>Figure 8.2 :</i>	<i>Lampe de poche et son graphique hybride.....</i>	<i>173</i>
<i>Figure 8.3 :</i>	<i>Le graphique hybride après l'enlèvement de $Sub(1, \{1,2\})$</i>	<i>174</i>
<i>Figure 8.4 :</i>	<i>Exemple d'assemblage avec graphique de liaison.....</i>	<i>178</i>
<i>Figure 8.5:</i>	<i>Illustration de l'algorithme de désassemblage sélectif global.....</i>	<i>179</i>
<i>Figure 8.6:</i>	<i>Graphique de liaison pour $C = (C3)$ avec présence d'un sous-ensemble</i>	<i>180</i>
<i>Figure 9.1 :</i>	<i>Organigramme de l'algorithme la solution de l'approche générale.....</i>	<i>185</i>
<i>Figure 9.2 :</i>	<i>Organigramme de l'algorithme de génération de ISN.....</i>	<i>187</i>
<i>Figure 9.3 :</i>	<i>Organigramme de l'algorithme d'exécution de ISN.....</i>	<i>189</i>
<i>Figure 9.4 :</i>	<i>Fonction findFT.....</i>	<i>193</i>
<i>Figure 9.5 :</i>	<i>Organigramme de l'algorithme de la tâche d'assignation optimum.....</i>	<i>195</i>
<i>Figure 9.6 :</i>	<i>Une matrice de priorité de désassemblage et sa représentation graphique</i>	<i>196</i>
<i>Figure 9.7 :</i>	<i>ISN et SN du produit étudié.....</i>	<i>197</i>
<i>Figure 9.8 :</i>	<i>SDP₆ du tableau 9.3</i>	<i>198</i>
<i>Figure 10.1 :</i>	<i>Effet des contraintes sur Ω_0</i>	<i>210</i>
<i>Figure 10.2 :</i>	<i>Exemple de diagramme de pétition.....</i>	<i>212</i>
<i>Figure 10.3 :</i>	<i>Proposition d'un algorithme d'identification des points faibles</i>	<i>217</i>
<i>Figure 10.4 :</i>	<i>Hierarchie des solutions de conception.....</i>	<i>220</i>

Notations :

ALBP :	Problème de balancement de la ligne d'assemblage (assembly Line balancing problem)
A-O-A :	Activité sur flèche (activity-on-arrow)
CAD :	Conception par ordinateur (computer-aided design)
CV :	Valeur de configuration (configuration-value)
DEI :	Index d'effort de désassemblage (disassembly effort index)
DFD :	Conception pour désassemblage (design for disassembly)
DLBP :	Problème de balancement de la ligne de désassemblage (disassembly line balancing problem)
DLBP-F :	Problème de balancement de ligne de désassemblage en présence d'échec de tâche (disassembly Line balancing problem in the présence de task failures)
D-L-N :	Réseau logique de désassemblage (disassembly logical net)
DPG :	Graphique de procédure de désassemblage (disassembly process graph)
DPM :	Matrice de priorité de désassemblage (disassembly precedente matrix)
DPN :	Désassemblage avec réseau de Petri (disassembly Petri net)
DPP :	Plan de procédures de désassemblage (disassembly process plan)
DSP :	Plan de séquence de désassemblage (disassembly sequence plans)
DSPNMG :	Modèle réseau de Petri de séquence de désassemblage (dissassembly system Petri net model generator)
DT :	Arbre de désassemblage (disassembly tree)
EECSPPN :	Réseau de Petri expert stochastique coloré et augmenté (expert enchanced coloured stochastic Petri net)
EOL:	Fin de vie (end-of-life)
GA :	Algorithme génétique (genetic algorithm)
HDT :	Arbre hiérarchique de désassemblage (hierarchical dissassembly tree)
ISN :	Réseau d'état incomplet (incomplete state network)
LCA :	Analyse du cycle de vie (life cycle assessment)

MRO :	Opportunités de récupération des matériaux (material recovery opportunities)
ODSPG :	Graphique réseau de Petri total de séquence de désassemblage (overall disassembly sequence Petri net graph)
OEM:	Équipements spécifiques au manufacturier (original equipment manufacturer)
PIC :	Composant avec contact d'empoîtement (contact interlocking components)
PLM :	Marge de profit/perte (profit-loss margin)
PN :	Réseau de Petri (Petri net)
PRM :	Gestion de récupération de produits (product recovery management)
QFD :	Déploiement de fonction de qualité (quality function deployment)
RBS :	Règle de base de système (rule-based system)
RHDT :	Arbre hiérarchique de désassemblage réduit (reduced hierarchical disassembly tree)
RRT :	Arbre réduit d'accessibilité (reduced reachability tree)
SN :	Réseau d'état (state network)
TCA :	Analyse du coût total (total cost assessment)
TTP :	Chemin transition-transition (transition-transition path)
WS :	Poste de travail (workstation)

Terminologie :

- Arc :** Branche ou flèche des arbres de désassemblage ou de leurs dérivés.
- Cannibalisation :** Récupération de certains sous-ensembles d'un produit dans le but d'en former un nouveau.
- Composant :** Objet solide dont la forme ne change pas durant la procédure de désassemblage et un élément de conception qui ne peut pas être brisé dans un futur désassemblage.
- Conception environnementale :** Conception considérant l'analyse du cycle de vie, les aspects environnementaux liés avec le produit, et la récupération. Elle est aussi nommée : conception verte (green design).
- Démontage :** Désassemblage destructif de composants.
- Désassemblage :** Procédure systématique de séparation d'un produit constitué de composants, de sous-assemblages ou d'autres groupes.
- Disposition :** Disposer d'un produit, action de rejet d'un produit (envoyer au ordure).
- Fixations :** Tous les éléments qui peuvent retenir deux ou plusieurs composants ensembles.
- Groupe rejet :** Deux composants ou plus qui sont regroupés dans le but d'être rejetés.

Groupe de matériaux : Deux composants ou plus qui ont des matériaux compatibles pour le recyclage.

Nœud : Composant dans les arbres de désassemblage ou leurs dérivés.

Nœud libre : Opération de désassemblage qui contient des composants libres qui ne sont pas directement inclus dans l'opération de désassemblage. Un nœud libre existe quand il n'y a pas de coût de désassemblage associé à la séparation des composants.

Pièce de travail : Sous-assemblage ou produit sur lequel le travail ou l'opération est effectué.

Produit : Composition de composants interconnectés formant une unité stable. Les composants du produit sont stabilisés en utilisant un ou quelques éléments de joint.

Quotient : Partie du produit à désassembler.

Réassemblage : Assemblage réalisé après un désassemblage.

Reconditionnement : Remettre les composants ou sous-ensembles en bonne condition dans un objectif de réutilisation.

Séparabilité : Facilité de séparation de deux ou plusieurs composants.

Séquence de désassemblage : Série de plusieurs tâches ou procédures de désassemblage. C'est le chemin à suivre pour désassembler tous les composants ou chercher un composant en particulier dans le produit.

Sous-assemblage : Sous-série contenant deux ou plusieurs composants qui sont interconnectés avec une ou plusieurs autres sous-séries.

Tâche ou procédure de désassemblage : Une opération de désassemblage pouvant être vue comme la décomposition d'un produit ou d'un sous-assemblage en deux ou plusieurs sous-assemblages ou composants.

1 Introduction

1.1 Problématique

Le problème de désassemblage a suscité beaucoup d'intérêt au cours des deux dernières décennies. En effet, la croissance économique a favorisé le développement de nouvelles technologies et l'émergence de nouveaux concepts d'intégration de ces technologies pour la conception d'équipements flexibles et performants. Pour répondre à des besoins sans cesse croissants du milieu, les équipements ont considérablement évolué rendant, du même coup, leur exploitation et leur maintenance très complexes. Dès lors, la problématique du désassemblage pour des fins de maintenance commençait à se poser autant pour les concepteurs que pour les utilisateurs des équipements. L'objectif étant de minimiser le temps d'interruption de service pour des fins de remplacement ou de réparation de composants défectueux d'un équipement.

La problématique du désassemblage a pris toute son importance avec la valorisation des produits en fin de vie motivée à la fois par des considérations économiques, légales et environnementales. De nos jours, les manufacturiers et les concepteurs de nouveaux produits font continuellement face aux consciences environnementales et aux législations pour le recyclage, le réusinage et la réutilisation des produits et des composants. Il faut changer les pratiques traditionnelles de conception pour considérer l'ultime fin de vie des produits. Lors des étapes de conception, il faut penser aux procédures économiques de récupération de matériaux et améliorer, de façon proactive, la disposition des déchets.

Le désassemblage de produits est l'un des éléments les plus importants du développement durable, qui est aujourd'hui un sujet politique et économique majeur. En 1971, le Club de Rome lance un vrai pavé dans la marre en publiant *Halte à la croissance*. Face à la surexploitation des ressources naturelles liées à la croissance économique et démographique, cette association privée internationale créée en 1968, prône la croissance zéro. En clair, le développement économique est présenté comme incompatible avec la

protection de la planète à long terme. C'est dans ce climat de confrontation et non de conciliation entre l'écologie et l'économie que se tient la Conférence des Nations Unies sur l'Environnement humain, à Stockholm, en 1972. Cette conférence sera à l'origine du premier vrai concept de développement durable baptisé, à l'époque, éco-développement. Des personnalités comme Maurice Strong, organisateur de la Conférence, puis le professeur René Dubos, Barbara Ward et Ignacy Sachs, insistent sur la nécessité d'intégrer l'équité sociale et la prudence écologique dans les modèles de développement économique du Nord et du Sud. Il en découlera la création du Programme des Nations Unies pour l'Environnement (PNUE) ainsi que le Programme des Nations Unies pour le Développement (PNUD). Mais plus le temps passe, plus la société civile prend conscience de l'urgence de mettre en place une solidarité planétaire pour faire face aux grands bouleversements des équilibres naturels. Ainsi, au cours des années 1980, le grand public découvre les pluies acides, le trou dans la couche d'ozone, l'effet de serre, la déforestation et la catastrophe de Tchernobyl ! Dès 1980, l'UICN parle pour la première fois de *Sustainable Development* (traduit à l'époque par développement soutenable). Mais le terme passe presque inaperçu jusqu'à sa reprise dans le rapport de Gro Harlem Brundtland, *Notre Avenir à tous*, publié en 1987. À l'époque Premier ministre en Norvège et présidente de la Commission Mondiale sur l'Environnement et le Développement, madame Brundtland s'attacha à définir ce concept de *Sustainable Development* par " un développement qui répond au besoin du présent sans compromettre la capacité des générations futures à répondre aux leurs ". Depuis cette date, le concept de développement durable a été adopté dans le monde entier.

Aujourd'hui les manufacturiers et les concepteurs de nouveaux produits font continuellement face aux consciences environnementales et aux législations pour le recyclage, le réusinage et la réutilisation des composants. Il faut changer les pratiques traditionnelles de conception pour inclure les considérations de l'ultime fin de vie des produits. Lors des étapes de conception, il faut penser aux procédures économiques de récupération de matériaux et améliorer, de façon proactive, la disposition des déchets.

Le travail sur les séquences de désassemblage est réalisé pour une variété de buts qui sont [31] : (1) la construction dans des lieux éloignés et la réparation dans des environnements inaccessibles ou hasardeux comme les fusées spatiales et les équipements nucléaires, (2) l'amélioration de l'efficacité des opérations de réparation et de maintenance et (3) les outils pour l'optimisation de l'assemblage. Parallèlement à cela, l'émergence de la performance environnementale des produits durant tout leur cycle de vie, doit aussi inclure les applications liées aux séquences de désassemblage dans : (1) la conception et l'optimisation des lignes de désassemblage et (2) la conception optimale considérant la phase de fin de vie des produits. La maintenance est le premier procédé nous poussant à se pencher sur le désassemblage de produits. Les études qui en ont découlé sont aujourd'hui réutilisées et améliorées pour le contexte de développement durable. Le désassemblage de produits en fin de vie est un procédé en croissance, directement lié au contexte, dû à l'augmentation du besoin des entreprises de considérer les facteurs environnementaux à la fin du cycle de vie du produit. Les procédures d'optimisation du désassemblage affectent les opérations de recyclage du produit. De 10 à 20 % du coût de recyclage dépend du désassemblage.

Nous devons aussi prendre en considération les différences entre le désassemblage et l'assemblage. Il existe sept différences essentielles [31] : (1) le désassemblage partiel est parfois préféré, (2) la procédure d'assemblage n'est souvent pas complètement réversible, (3) la valeur ajoutée dans les procédures de désassemblage est habituellement négligeable, (4) l'incertitude existe en regard de la qualité des composants, (5) la présence d'incertitude dans l'approvisionnement des produits en fin de vie du point de vue qualitatif et quantitatif et (6) le désassemblage est principalement réalisé par un opérateur plutôt que par des lignes automatisées ou des robots.

1.2 1.2 Objectifs

Notre objectif est de proposer des stratégies de désassemblage spécifiques aux contextes de maintenance et de revalorisation ; de faire une analyse complète du produit : son cycle

de vie et les coûts de ce cycle, sa structure géométrique, les matériaux avec lesquels il est composé, les caractéristiques de chacun de ses composants, sa complexité, les différentes façons de le représenter, ses options de fin de vie, et les façons possibles de le désassembler. Nous voyons en détails tous les aspects du produit pour une prise de décision éclairée pour les choix et l'utilisation d'une stratégie de désassemblage. Les facteurs de décision doivent comprendre aussi les aspects environnementaux, les coûts engendrés, et la façon dont le désassemblage va être effectué.

1.3 Méthodologie

L'ensemble du travail est conçu sous forme de guide du désassemblage, nous permettant de réaliser le désassemblage de produit de façon éclairée et réfléchie. Pour cela il est primordial de bien connaître tout notre produit. Une analyse détaillée du cycle de vie et des coûts doit être faite. Il est aussi essentiel de comprendre plusieurs autres aspects du produit comme tous les paramètres de récupération (la réparation, la remise à neuf, la nouvelle fabrication, la cannibalisation et le recyclage) et la complexité de la structure (réassemblage, profondeur et cheminement).

Le deuxième élément à comprendre et à analyser est le concept de désassemblage. La connaissance du produit est le point central de l'analyse. Il faut bien comprendre le désassemblage. On doit préciser pourquoi on désassemble? Qu'est-ce que l'on désassemble? En quoi consiste le désassemblage?

Nos besoins en terme de désassemblage sont déterminants pour un désassemblage réfléchi. Après avoir analysé le produit et compris le désassemblage, l'objectif visé donne tout son sens aux stratégies de désassemblage.

C'est seulement après avoir pris conscience de ces éléments que l'on peut choisir la meilleure stratégie possible pour nous. Ces stratégies étant très diversifiées et utilisant plusieurs concepts différents, nous devons les classer en catégories. Toutes les

stratégies sont présentées en fonctions des caractéristiques du produit, des options de récupération et de nos besoins. Chacune des catégories présente des regroupement de stratégies ayant plusieurs aspects en commun. Nous avons donc créé cinq catégories : les stratégies de base considérant des aspects particuliers des liaisons de composants, les stratégies utilisant les réseaux de Petri, les stratégies considérant l'aspect environnemental et économique, les stratégies spécifiques à la maintenance et une stratégie considérant plusieurs postes de désassemblage. Toutes les stratégies sont détaillées et comprennent un exemple illustratif. L'analyse de ces stratégies et de plusieurs autres sont aussi réalisées.

Avec cette analyse, serait-il possible de nous simplifier la vie lors du désassemblage? C'est pour cela que le retour sur la conception est primordial et encore mieux l'utilisation de la conception environnementale.

1.4 Structure de l'étude

Tout d'abord, il est essentiel d'analyser le cycle de vie et les coûts du cycle de vie d'un produit. Tous les points suivants représentent la partie post consommation dans le cycle de vie du produit sauf les cas de maintenance et de la conception liée au désassemblage.

Nous nous attaquons au problème du désassemblage de produit en définissant nos besoins, en réalisant une analyse de fin de vie des produits à désassembler par la gestion de la récupération de produits et la quantification des niveaux de désassemblage, et en évaluant la complexité du produit. Une fois ces éléments déterminés nous choisissons une stratégie de désassemblage parmi celles présentées. La figure 1.1 illustre la séquence des thèmes et sujets abordés dans ce mémoire.

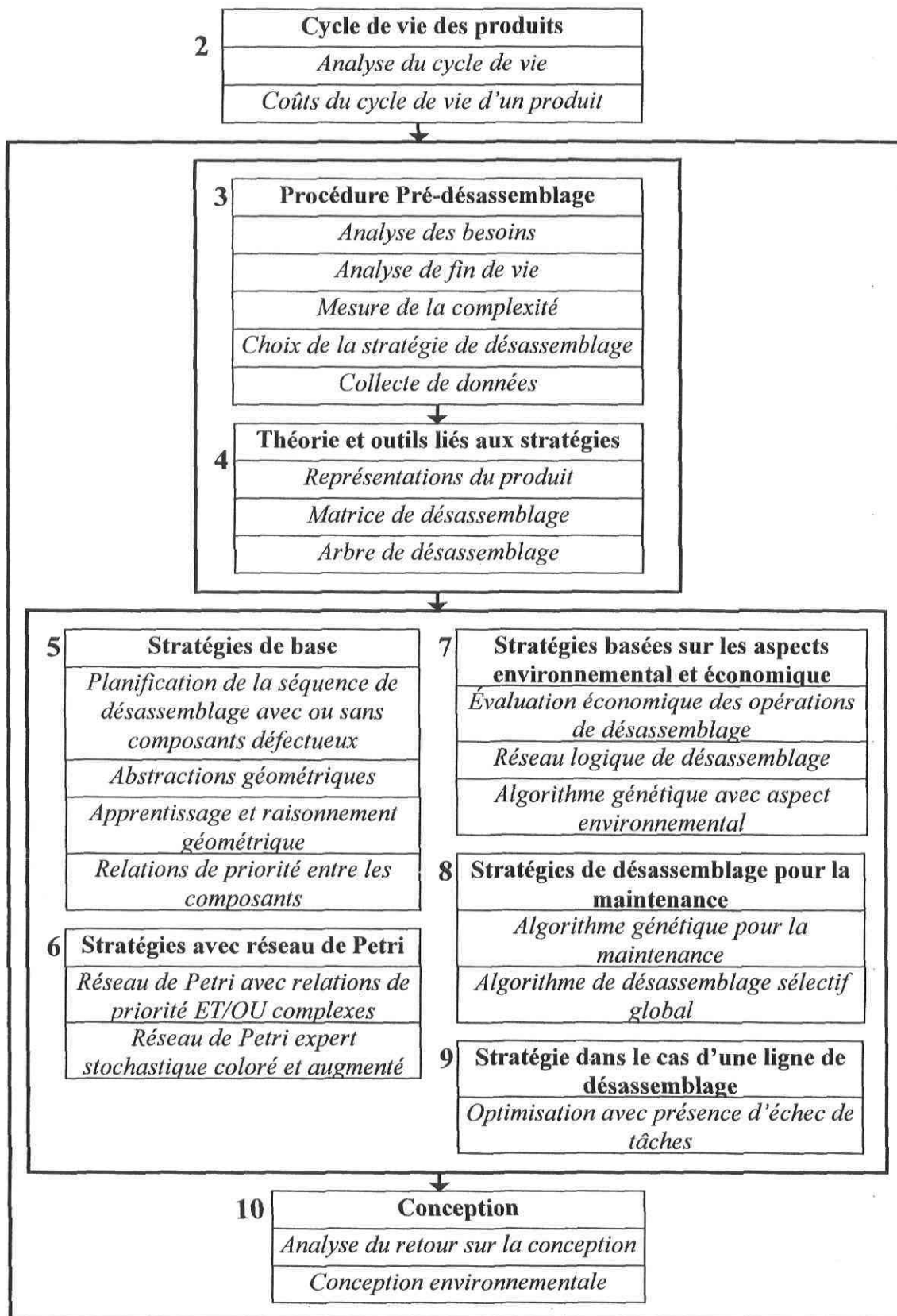


Figure 1.1 : Structure hiérarchique utilisée pour le document

La collecte de données sur le produit est effectuée en fonction des paramètres nécessaires à la stratégie de désassemblage utilisée. Avant de détailler les stratégies, nous présentons la théorie de représentation de produit ainsi que deux outils très utilisés et essentiels pour les méthodes et les algorithmes de désassemblage : les matrices et les arbres de désassemblage.

Par la suite, les différentes stratégies de désassemblage sont présentées. Elles sont nombreuses et permettent une variété de types de désassemblage. Plusieurs niveaux de détails existent, de l'analyse de la géométrie à la logistique complète entourant le désassemblage des produits en passant par les séquences et les lignes de désassemblage. Le classement des différentes méthodes peut être visualisé par un diagramme en oignon (figure 1.2). Dans le centre du diagramme, le niveau détaillé est représenté, il mise sur les propriétés physiques des composants et des connections. Pratiquement toutes les stratégies comprennent, inévitablement ce niveau. Le niveau de génération de séquences définit la structure du produit et les séquences possibles pour le désassembler.

Les stratégies de base, les méthodes graphiques et géométriques, les algorithmes génétiques et une partie des méthodes du Réseau de Petri présentent bien ce niveau d'analyse. Une stratégie pour les lignes de désassemblage analyse le niveau de planification de tâche. Le niveau de la logistique inversée est une analyse encore plus vaste qui considère toute la chaîne de désassemblage. Elle s'écarte du cas plus spécifique du problème de désassemblage et n'est donc pas prise en considération dans cette analyse.

Toutes les stratégies sont détaillées et analysées. En premier lieu des méthodes de base pour le désassemblage sont introduites. Des méthodes graphiques et géométriques permettent de visualiser plusieurs facettes de l'analyse de la structure du produit et du problème de génération de séquences. Quatre méthodes sont détaillées : la planification de la séquence de désassemblage avec ou sans composant défectueux, les abstractions géométriques, l'apprentissage et le raisonnement géométrique et les relations de priorités entre les composants d'assemblage.

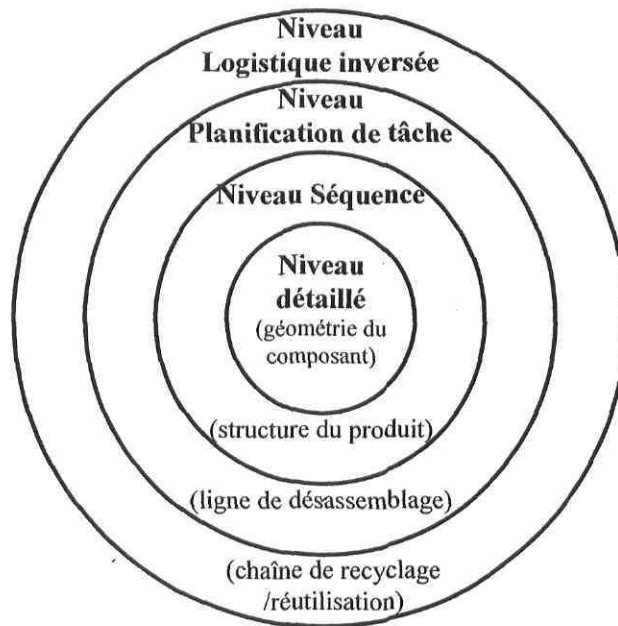


Figure 1.2 : *Positionnement de génération de séquences de désassemblage*

Source : Lambert [31]

Les méthodes utilisant le réseau de Petri pour résoudre les problèmes de génération de séquences de désassemblage étant très répandues dans l'étude du désassemblage, nous les expliquons en trois niveaux : le réseau de Petri normal, le réseau de Petri avec relations de priorité ET/OU complexes, et le réseau de Petri expert stochastique coloré et augmenté.

L'introduction de l'aspect environnemental et économique est introduite dans quelques stratégies : l'évaluation économique des opérations de désassemblage pour le recyclage, le remise à neuf et la réparation, le réseau logique de désassemblage et l'algorithme génétique considérant l'aspect environnemental et économique.

Dans une perspective différente deux stratégies pour la maintenance sont utilisées pour trouver les séquences de désassemblage : l'algorithme génétique et le désassemblage sélectif global.

Pour conclure sur les stratégies de désassemblage nous allons étudier une méthode considérant une ligne de désassemblage. Pour traiter le problème attaché à ces lignes, on présente l'étude des lignes de désassemblage avec possibilité d'échec de tâches.

Pour boucler la boucle sur l'analyse des stratégies de désassemblage dans un contexte de développement durable, il est inévitable de parler de la conception. Le retour sur la conception et la conception pour le désassemblage sont deux éléments importants. Le retour sur la conception permet de revenir sur la conception après le désassemblage dans le but de l'améliorer et de faciliter. La conception environnementale est développée comme un outil à utiliser dès la conception du produit.

2 Problématique du désassemblage

Il est important, en premier lieu, de savoir pourquoi on désassemble. Le produit à désassembler doit être vu dans tout son cycle de vie. Il nous faut aussi préciser en quoi consiste le désassemblage.

2.1 Pourquoi désassembler?

Il y a plusieurs raisons pour désassembler un produit : le recyclage, la récupération de composants, l'enlèvement de composants présentant un danger (toxiques, explosifs, etc.) et la maintenance.

Quelles que soient les raisons, les objectifs et les besoins face au désassemblage du produit, l'action de désassembler n'est jamais aussi facile que d'assembler. La conception est pensée en fonction de l'assemblage, l'ergonomie, la résistance, mais très rarement en prenant en considération le désassemblage.

2.2 Qu'est-ce qu'on désassemble?

Nous voulons désassembler un produit mais qu'est-ce qu'il faut savoir sur ce produit? Il est important de connaître le produit que l'on veut désassembler de façon plus détaillée que de savoir comment il a été conçu. Tout son cycle de vie est à considérer, des matériaux bruts à son utilisation. Normalement, après cela le produit est directement jeté mais dans une perspective de développement durable, plusieurs nouvelles étapes s'ajoutent. Suivant l'utilisation du produit une collecte est effectuée. Lors de sa réception, nous devons connaître les options qui s'offrent à nous. Pour cela l'analyse complète du cycle de vie est un élément important et nécessaire pour faire des choix éclairés par rapport aux stratégies de désassemblage.

2.2.1 Cycle de vie des produits

Le cycle de vie d'un produit comporte plusieurs étapes. Les étapes post utilisation du produit sont très importantes. Après l'utilisation, les produits sont normalement jetés. Pour le cycle de vie complet et écologique plusieurs autres étapes peuvent être présentes. La collecte des produits en fin de vie peut redonner une autre chance au produit.. La figure 2.1 illustre très bien toutes les étapes du cycle de vie d'un produit.

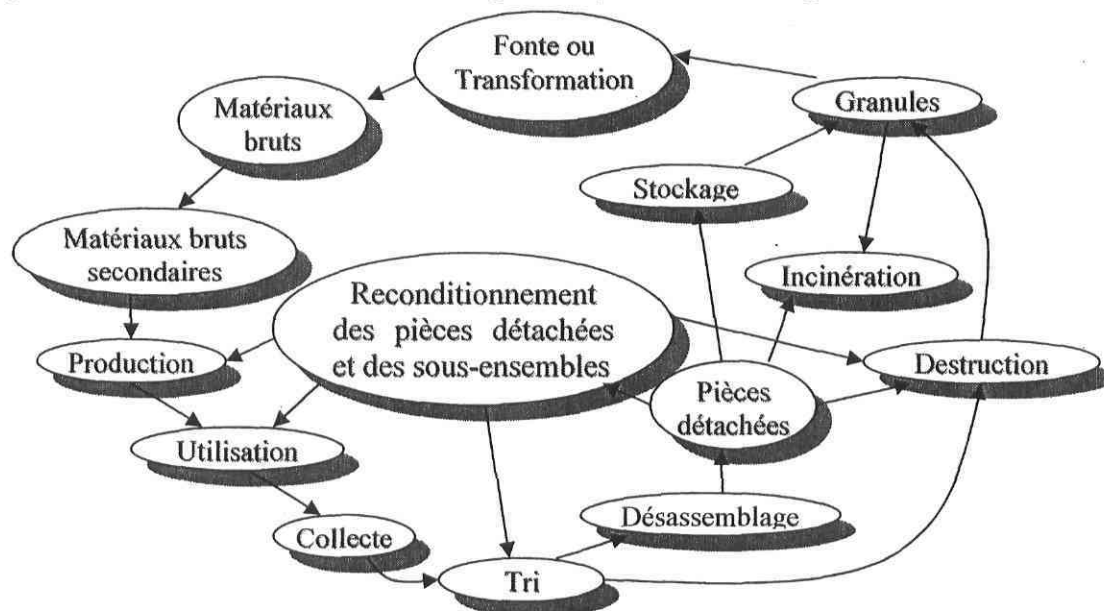


Figure 2.1 : Cycle de vie d'un produit

Source : Skaf, Gerner, Binder, Descotes-Genon, David [42]

Bien que le sujet porte sur la problématique de désassemblage du produit, tout le cycle de vie doit être pris en considération. Une analyse complète du produit est essentielle, car elle permet de faire un choix éclairé pour la stratégie à utiliser. De plus, dans notre collecte de données sur le produit, toutes les étapes prés désassemblage sont à considérer. Les étapes post désassemblage nous permettent de savoir exactement quel type de désassemblage nous devons effectuer (le désassemblage complet ou partiel, le retrait prioritaire de composant dangereux, les besoins en composants, etc.). L'analyse du cycle de vie du produit est donc à la base de la compréhension et de l'analyse de son désassemblage. Pour nous donner de l'information quantitative sur notre produit, les coûts du cycle de vie sont à considérer.

2.2.2 Le coût d'un cycle de vie de produit

Le coût total du cycle de vie d'un produit comprend les coûts encourus depuis l'extraction des métaux jusqu'au désassemblage du produit. La figure 2.2 nous offre une vision globale.

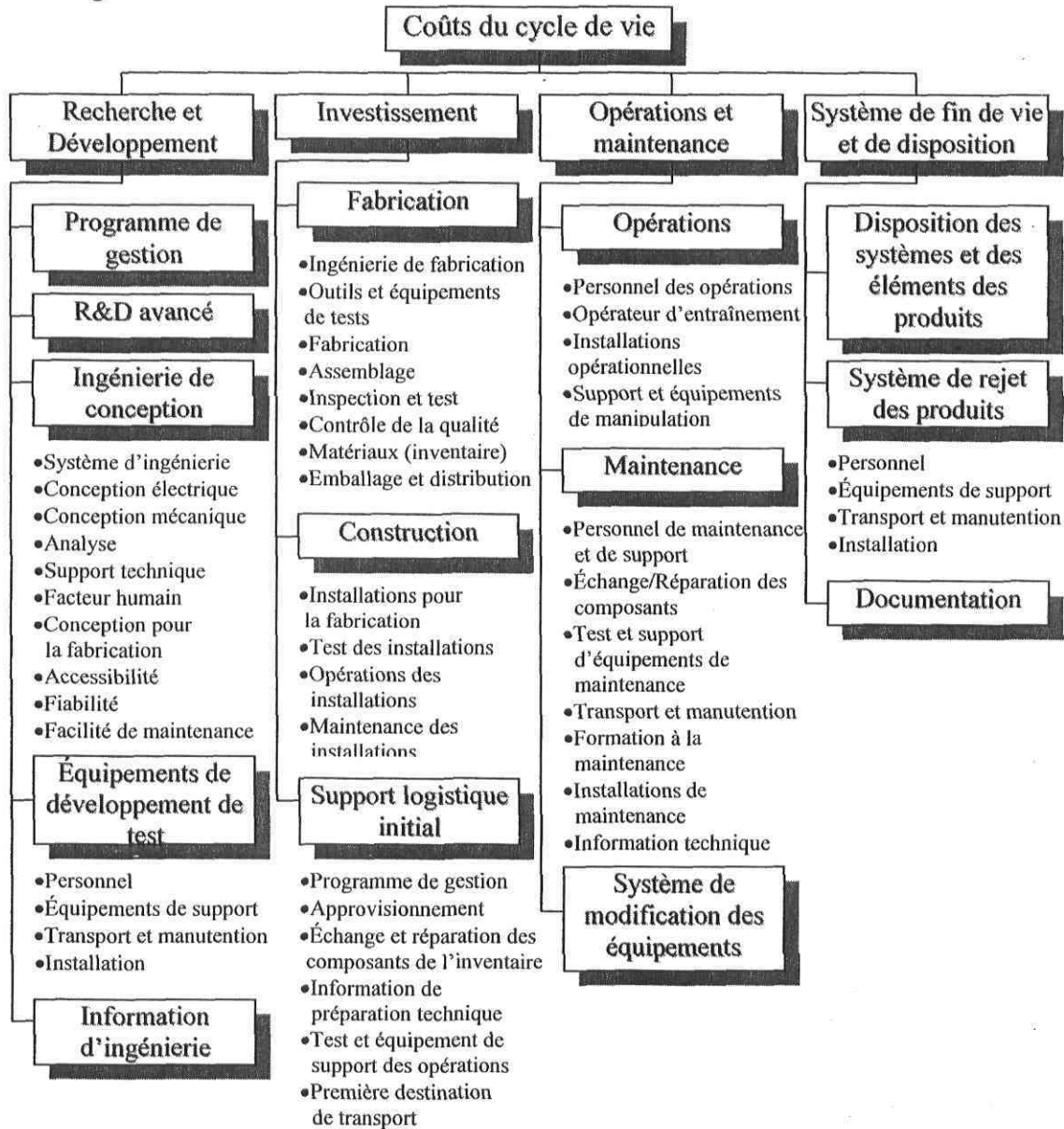


Figure 2.2 : Coûts du cycle de vie d'un produit

Source: Ertas, Jones [7]

Les quatre catégories de coûts pour le calcul des coûts d'un cycle de vie sont, la recherche et le développement (programme de management, recherche et développement avancée, le développement d'équipements et de tests, l'ingénierie de la conception), les investissements (la fabrication, les différentes constructions nécessaires, la logistique de support initiale) et les opérations de maintenance et de recyclage.

Les coûts pour un produit ne s'arrêtent pas à l'entreprise. La figure 2.3 permet de visualiser les coûts accumulés pour un produit. Le coût total d'un produit est essentiellement la somme des coûts encourus par le manufacturier, l'utilisateur et la société.

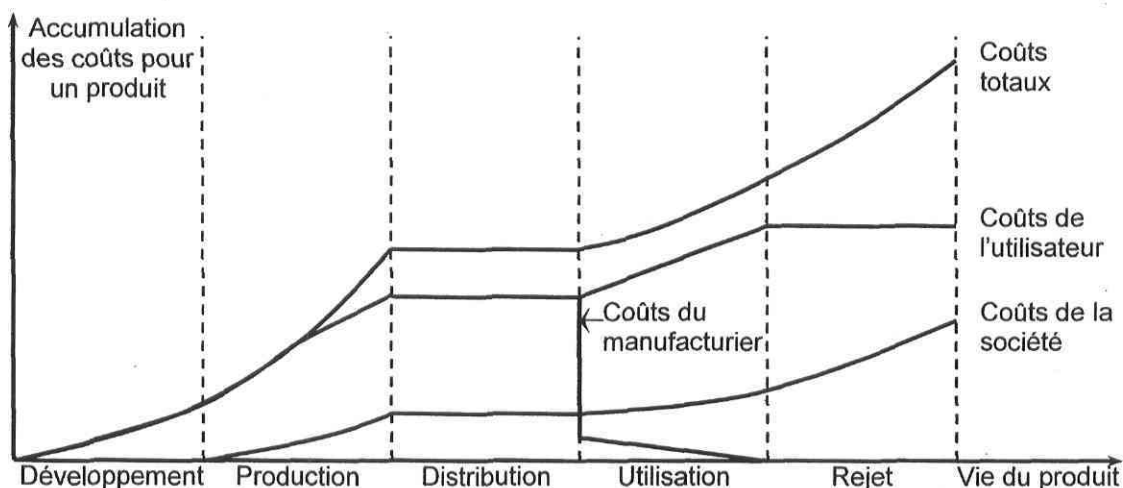


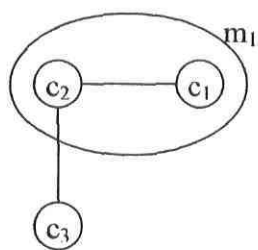
Figure 2.3 : Coûts cumulés d'un cycle de vie

Source: Alting [1]

2.3 En quoi consiste le désassemblage?

C'est une action de séparation de composant. Il existe plusieurs types d'actions qui peuvent influencer ou modifier les choix à faire et les stratégies à utiliser. La simplicité ou la complexité des algorithmes de désassemblage est très souvent liée aux types d'actions de désassemblage. Des hypothèses sont utilisées pour éviter ou éliminer les produits nécessitant certains types d'actions. Dans notre analyse les six types d'actions prises en considération sont :

- Union de type joint (figure 2.4(a)). L'action consiste à défaire le joint et séparer les composants ;
- Union de type contact :
 - 1) Entre deux composants (figure 2.4(b)). L'action consiste en la séparation d'un composant de tous les autres ;
 - 2) Entre plus de deux composants (figure 2.4(c)). L'action consiste en la séparation des composants d'une des extrémités, jusqu'au nœud désiré.
- Assemblages sans relation de priorité entre les composants (figure 2.4(d)). Dans ce cas on ne fait rien ;
- Union de type joint entre un élément de fixation (vis, écrou, etc) et le sous-assemblage (figure 2.4(e)). Dans ce cas l'élément de fixation doit être retiré ;
- Union avec un élément de fixation entre des sous-ensembles et/ou composants (figure 2.4(f)). Dans ce cas, l'élément de fixation doit être retiré et alors les sous-assemblages/composants qui étaient unis par les éléments de fixation sont séparés ;
- Sous-assemblages unis par un élément de fixation externe où la séquence de désassemblage est interchangeable (figure 2.4(g)). Dans ce cas, l'action consiste à retirer l'élément de contact.



(a)
Représentation



Action
Défaire le joint et séparer
 c_3 et m_1

(b)
Représentation

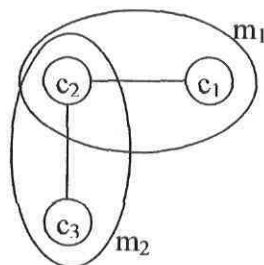


Action
Séparer c_1 et c_2

(c)
Représentation



Action
Séparer c_1 et c_2 , et ainsi de
suite jusqu'à c_i ; ou c_n et
 c_{n-1} à c_i



(d)
Représentation



Action

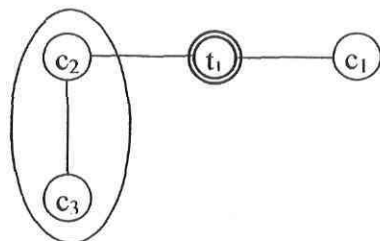
Rien

(e)
Représentation



Action

Enlever t_1

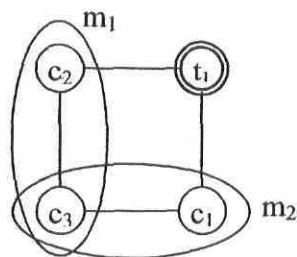


(f)
Représentation

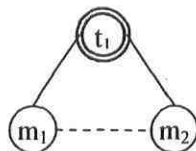


Action

Enlever la fixation t_1 puis
séparer c_1 et m_1



(g)
Représentation



Action

Enlever la fixation t_1

Légende : m_i : sous-ensemble ; t_i : fixation ; c_i : composant

Figure 2.4 : Représentation et actions de désassemblage

Source : Torres, Puente, Aracil [51]

3 Procédure pré-désassemblage

L'étape la plus importante pour l'analyse de stratégies de désassemblage est la procédure de pré-désassemblage. Elle consiste à l'analyse des besoins par la définition des objectifs visés. Les options de récupérations doivent être déterminées pour connaître les niveaux de désassemblage à considérer. À partir des besoins et des indices sur les niveaux de désassemblage, nous pouvons analyser quelle est la stratégie à utiliser pour le désassemblage de notre produit. Finalement, la collecte de données vient compléter toute la procédure de pré-désassemblage.

3.1 Analyse des besoins

Il est important de fixer les objectifs à la base de la stratégie du désassemblage. Ils déterminent quelles sont nos priorités de désassemblage, les différents outils à utiliser (chapitre 4) et la stratégie de désassemblage à utiliser (voir chapitres 5 à 9). Penev et Ron [37] ainsi que Johnson et Wang [20,21] définissent l'ensemble de ces objectifs :

- Spécifier les matériaux et composants valables et toxiques par produit ;
- Maximiser le profit et minimiser le coût pour exécuter les opérations de désassemblage ;
- Trouver les chemins les moins coûteux pour retirer les composants toxiques. Considérer les produits à éliminer comme une source de perte ou de profit pouvant recouler de l'exécution des opérations de désassemblage ;
- Générer toutes les opérations de désassemblage faisables ;
- Déterminer l'exécution de séquences en considérant le chemin le plus économique et efficace possible ;
- Déterminer le niveau optimal de désassemblage ;
- Récupérer les matériaux ou groupes de matériaux spécifiques ;
- Récupérer les parties de composants spécifiques ;

- Désassembler les produits usagés ;
- Analyser en considérant les matériaux les plus attirants et les moins attirants à récupérer.

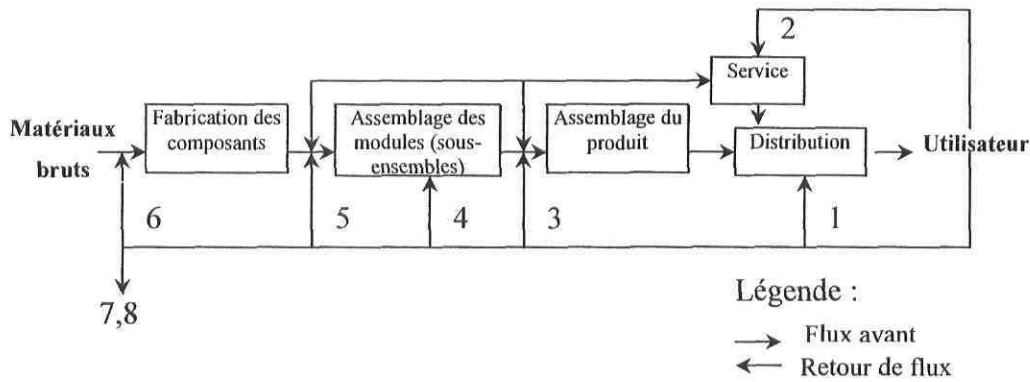
3.2 Analyse de fin de vie du produit à désassembler

L'analyse de fin de vie est requise pour obtenir toutes les informations qualitatives et quantitatives sur le produit. Elle permet d'avoir une vision globale de tout le produit. La gestion de la récupération de produits nous permet de voir les options qui s'offrent à nous lors du choix de la stratégie de désassemblage. Ces options vont nous faire réaliser qu'il y a plusieurs niveaux de désassemblage. Les différents niveaux peuvent être quantifiés pour une analyse plus détaillée.

3.2.1 Gestion de la récupération de produits

Avant de procéder au désassemblage, il est important de connaître et d'analyser les différentes options de fin de vie du produit. La figure 3.1 nous permet de visualiser les différentes étapes de fabrication, d'assemblage, de distribution et les options de récupération des produits. Le flux avant présente les étapes de fabrication d'un produit, des matériaux bruts à sa distribution. Les retours de flux sont définis en trois grandes catégories : la gestion de l'élimination (incinération (7) et enfouissement (8)), la gestion de récupération du produit (cannibalisation (5), recyclage (6), réparation (2), remise à neuf (3) et nouvelle fabrication (4)) et la réutilisation directe (1).

L'analyse des opportunités et du traitement qui doit être envisagé à la première étape pour toute entreprise qui évolue dans la gestion de la récupération de produits . Les options de récupération des produits représentent les choix qui s'offre à nous lors du tri (voir figure 2.1 de la section 2.2.1). Les cinq options de récupération de produits sont : la réparation, la mise à neuf, la nouvelle fabrication, la cannibalisation, et le recyclage.



Gestion de l'élimination	Gestion de la récupération du produit	Réutilisation directe
7 : Incinération	5 : Cannibalisation	2 : Réparation
8 : Enfouissement	6 : Recyclage	3 : Reconditionnement
	4 : Remise à neuf	1 : Réutilisation directe / revente

Figure 3.1 : Chaîne de montage intégrée

Source : Thierry, Salomon, Nunen et Wessenbove [49]

La réparation :

La réparation consiste à remettre un composant hors d'usage l'état d'opération. La qualité des produits réparés est généralement inférieure à la qualité du produit neuf. La réparation du produit inclut la réparation et/ou le remplacement des composants brisés. Les autres composants ne sont généralement pas affectés. La réparation nécessite habituellement très peu de désassemblage et réassemblage. Les opérations de réparation peuvent être effectuées directement chez le consommateur ou dans un centre de réparation de l'entreprise.

Le reconditionnement:

Le but de but du reconditionnement est de remettre le produit usagé à l'état neuf ou à un état spécifié à l'avance. Le produit usé est donc désassemblé en sous-ensembles qui sont inspectés et réparés ou remplacés. Les sous-assemblages sont ensuite réassemblés pour donner notre produit remis à neuf.

La remise à neuf :

Le but de la nouvelle fabrication est de remettre le produit utilisé pratiquement à la qualité d'un produit neuf. Le produit est complètement désassemblé et tous les composants et sous-ensembles sont inspectés minutieusement. Les composants usagés ou périmés sont automatiquement remplacés par des neufs. Les composants réparables sont réparés et testés. La nouvelle fabrication peut être combinée avec des améliorations technologiques.

La cannibalisation :

Dans les trois options de récupération précédentes, une large proportion du produit utilisé est réutilisée. Dans la cannibalisation, seulement une petite proportion est réutilisée. Le but de la cannibalisation est de récupérer une série limitée de composants réutilisables du produit utilisé. Ces composants sont réutilisés en réparation, remise à neuf ou dans le cadre d'une nouvelle fabrication d'autres produits ou sous-ensembles. La qualité de base pour les composants cannibalisés dépend du processus dans lequel ils seront réutilisés.

Le recyclage :

Le but des options précédentes est de garder, le plus possible, l'identité et la fonctionnalité des produits et de ses composants. Dans le recyclage, on perd l'identité et la fonctionnalité des produits et des composants. Le but du recyclage est la réutilisation des matériaux. Ces derniers peuvent être réutilisés dans la production de composants d'origine si la qualité des matériaux est élevée. Le recyclage débute quand le produit utilisé et les sous-ensembles sont désassemblés en composants. Ces derniers sont séparés en catégories de matériaux distincts.

Les options de récupération de produits requièrent différents niveaux de désassemblage, comme on a pu le remarquer dans la description des options. Le tableau 3.1 présente les différentes options de récupération de produit.

Tableau 3.1 : Les options de récupération de produit

[49]

	Niveau de désassemblage	Qualité requise	Résultat sur le produit
Réparation	Au niveau du produit (désassemblage de base)	Restauration du produit	Certains composants réparés ou remplacés par nécessité seulement
Remise à neuf	Au niveau du sous-ensemble	Remplacement de tous les sous-ensembles et amélioration du niveau spécifique de qualité	Certains sous-ensembles réparés ou remplacés
Nouvelle fabrication	Au niveau du composant	Remplacement de tous les sous-ensembles et composants, et amélioration importante de la qualité	Sous-ensembles usés et nouveaux/ composants combinés dans le nouveau produit, potentiellement améliorés (mise à jour technologique)
Cannibalisation	Récupération sélective des composants	Dépend des procédures pour lesquelles les composants sont réutilisés	Certains composants réutilisés, produit recyclé
Recyclage	Au niveau des matériaux	Production de composants originaux	Matériaux réutilisés pour produire de nouveaux composants

Ces options peuvent être quantifiées pour nous permettre de faire un choix éclairé sur le niveau de désassemblage nécessaire.

3.2.2 Quantification des niveaux de désassemblage

La partie sur les options de fin de vie est appliquée normalement avant la réalisation du désassemblage. Ces options représentent les choix possibles lors de la fin de vie du produit. Il n'est pas nécessairement obligatoire de réaliser le désassemblage complet du produit.

Comme nous l'avons vu précédemment, plusieurs choix s'offre à nous lors de la réception d'un produit : la réparation, le reconditionnement, la remise à neuf, la cannibalisation, le recyclage, le réutilisation directe, l'incinération et l'enfouissement. De ces options nous allons en retirer des indices pour les différents niveaux de désassemblage. La réparation est directement associée à la réutilisation du produit dans sa presque totalité. Le désassemblage partiel ou la maintenance nous donne l'indice de réutilisation. Le reconditionnement et la remise à neuf demandent tous deux un

désassemblage complet en composants ou sous-ensembles, ils ont donc le même indice, nommé indice de désassemblage. La cannibalisation étant un désassemblage complet mais en prenant en considération seulement quelques sous-ensembles et/ou composants, nous donne un indice différent nommé indice de démontage. Ce dernier prend en considération le désassemblage destructif (pouvant briser certains composants), ce que l'indice précédent exclut dû au besoin de récupérer le plus de composants possibles. Le recyclage a son propre indice qui est l'indice de recyclage. Le dernier indice n'est pas une option de récupération car il ne considère, pas la récupération du produit. L'indice de rejet est donc le dernier et l'ultime choix possible, c'est-à-dire, l'élimination du produit soit par incinération ou par enfouissement comme mentionné à la figure 3.1. Nous obtenons donc cinq indices pour les niveaux de désassemblage : la réutilisation, le désassemblage, le démontage, le recyclage, et le rejet.

Les valeurs de ces options de fin de vie sont souvent déterminées dans les articles traitant du désassemblage. Elles sont présentées de façon différente mais incluent les mêmes paramètres de calcul. Tiwari, et al. [50] ainsi que Penev et Ron [37] définissent de façon clair ces valeurs. Dans ce cas particulier, les valeurs de fin de vie sont déterminées dans une perspective de désassemblage des réseaux de Petri (chapitre 6). Ces calculs sont facilement transposables avec toutes les stratégies de désassemblage car ils sont totalement indépendants de l'utilisation future des données fournies. Les valeurs de fin de vie sont un complément d'information important pour l'analyse du désassemblage.

Nomenclature

$I_{réu}$: Indice de réutilisation	I_{dm} : Indice de démontage
J_s : Revenu de réparation du composant (\$/kg)	I_{ds} : Indice de désassemblage
J_{ds} : Revenu de désassemblage (\$/kg)	I_r : Indice de recyclage
J_{dm} : Revenu de démontage (\$/kg)	T_{sep} : Coût de séparation (\$/kg)
J_r : Revenu de recyclage (\$/kg)	T_{ds} : Coût de désassemblage (\$/kg)
T_k : Coût de rejet sans désassemblage (\$/kg)	δ : Facteur de masse
T_d : Coût de rejet après désassemblage	U_{dp} : Indice de rejet (élimination)

Description du contrôleur de décision pour proposer un système de désassemblage

Tout d'abord pour bien situer ces indices dans l'analyse du produit et de son désassemblage, nous voyons le contrôleur de décision pour un système de désassemblage. La figure 3.2 permet de situer les indices de fin de vie dans un système visant le désassemblage. Ici la méthode des réseaux de Petri est utilisée mais il est possible d'utiliser n'importe quelle autre méthode de désassemblage. C'est une figure résumant tout le contrôle de décision pour un système de désassemblage.

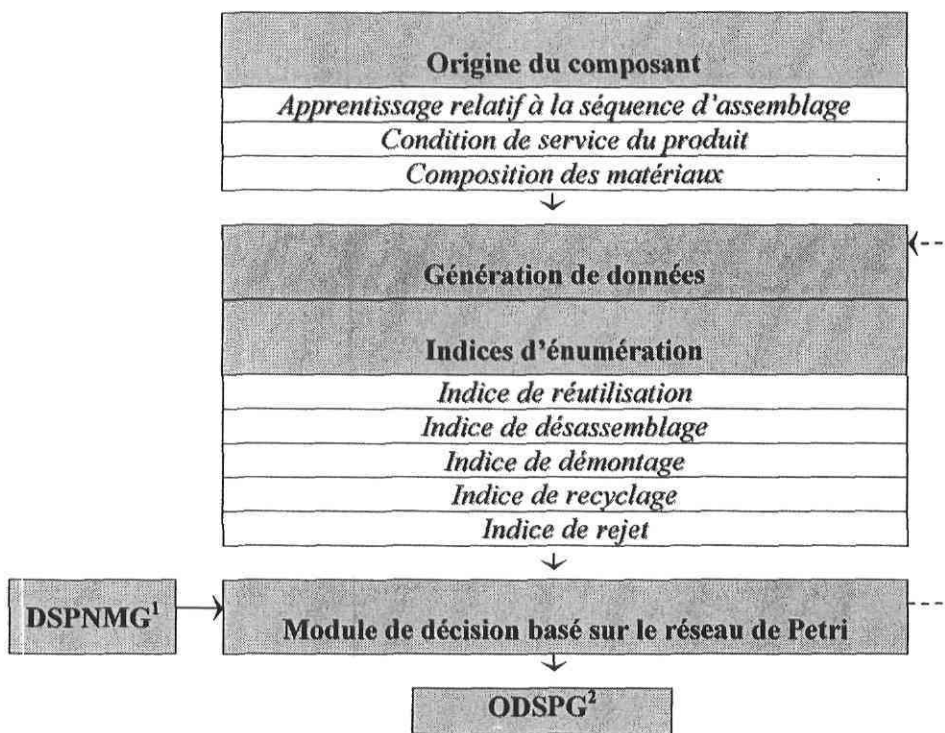


Figure 3.2 : Contrôleur de décision pour un système de désassemblage

Source : Tiwari, Sinha, Kumar, Rai, Mukhopadhyay [50]

¹ DSPNMG : Modèle du réseau de Petri pour une séquence de désassemblage

² ODSPG : Graphique du réseau de Petri pour une séquence de désassemblage

Étape 1 : Calcul de la réutilisation

Si un produit n'est pas complètement inopérationnel, il peut être remis en opération à un niveau spécifique de qualité avec un minimum d'effort en remplaçant seulement un ou quelques composants. La réparation inclut un désassemblage des fixations ainsi que le remplacement limité de composants. Les autres composants ne sont pas affectés. La réparation est justifiée seulement si les revenus générés du produit utilisé n'excèdent pas les revenus de n'importe quel autre indice.

$$\text{Index de la réutilisation : } I_{réu} = (J_{réu} - \max[J_{ds}, d_{dm}, J_r]) / J_{réu}$$

Si $I_{réu} > 0$, le processus de service doit continuer, autrement on arrête.

Étape 2 : Calcul du désassemblage

Le désassemblage est faisable si les revenus générés par cette action sont plus grand que le revenu maximum des autres indices et que le coût du désassemblage est plus bas que le coût de séparation. Tous les paramètres du désassemblage sont considérés : l'identification de la tâche, l'approvisionnement des outils nécessaires, l'enlèvement de fixations et le positionnement des outils.

$$\text{Index de désassemblage : } I_{ds} = (J_{ds} - \max[J_{réu}, J_{dm}, J_r]) / (T_{sep} - T_{ds}) \times (T_{sep} / J_{ds}),$$

Si $I_{ds} > 0$, le processus continu, autrement on arrête.

Étape 3 : Calcul du démontage

Le démontage est une technique destructive. Par l'application du démontage, un produit peut être brisé à bas coût. Si l'assemblage est composé du même type de matériaux, il est possible de le recycler. Autrement, si l'assemblage comprend plusieurs types de matériaux, les matériaux doivent être séparés avant le recyclage. Cela peut être fait par le

démontage. L'utilisation de cette procédure est justifiée seulement si les revenus du démontage sont plus élevés que le revenu maximum des autres indices.

$$\text{Index de démontage : } I_{dm} = (J_{dm} - \max[J_{réu}, J_{ds}, J_r]) / J_{dm}$$

Si $I_{dm} > 0$, le processus continu, autrement on arrête.

Étape 4 : Calcul recyclage

Le recyclage est directement lié avec la réutilisation des matériaux. Le recyclage peut être divisé en trois options :

1. Recyclage de haut niveau. La qualité originale des matériaux peut être récupérée par cette procédure. Ex. : aluminium.
2. Recyclage de bas niveau. Les matériaux recyclés de faible qualité par rapport aux originaux. Ex. : matériaux contaminés.
3. Recyclage alternatif. Dans cette catégorie, des matériaux complètement différents peuvent être obtenus par recyclage des matériaux originaux. Ex. : plastique.

Avant de recycler, une option de recyclage optimale doit être déterminée. Le recyclage est permis si la fraction des revenus obtenus de la séparation de matériaux usagés est plus élevée que le maximum de revenus des autres indices.

$$\text{Index de recyclage: } I_r = (\delta J_r - \max[J_{réu}, J_{ds}, J_{dm}]) / (\delta J_r)$$

Il est important que δ soit le plus petit possible et compris entre 0 et 1.

Si $I_r > 0$, le processus continu, autrement on arrête.

Étape 5 : Calcul du rejet

L'étape finale quand nous faisons du désassemblage pour le rejet, où aucun revenu n'est généré. Le coût du rejet est séparé en deux portions : le transport et l'enfouissement ou incinération.

$$\text{Index de rejet : } I_{dp} = (T_k - T_d) / T_k$$

Si $U_{dp} > 0$, le processus continu, autrement le composant est jeté directement.

Une fois les indices déterminés, la stratégie de désassemblage peut être choisie. Il est à noter que, bien que ces indices soient un paramètre d'évaluation important, plusieurs autres éléments peuvent influencer le choix de la stratégie. Nous allons donc voir un autre élément essentiel : la complexité du produit.

3.3 Évaluation de la complexité du produit

La complexité du produit est aussi une facette importante qui mérite une attention particulière. Plus il y a d'actions à effectuer, de composants, de séparations difficiles de composants, s'il y a présence de composants particuliers (dangereux ou toxiques), plus le produit est complexe.

3.3.1 Évaluation de la complexité

Le tableau 3.2 présente les résultats du calcul théorique pour connaître le nombre maximum de séquences de désassemblage selon le nombre de composants dans le produit.

Tableau 3.2 : Nombre total de séquences de désassemblage [28]

Nombre de composants	Nombre de sous assemblages	Séquences pour le désassemblage complet	Nombre total de séquences de désassemblage
1	1	1	1
2	3	1	2
3	7	3	7
4	15	15	41
5	31	105	346
6	63	945	3797
7	127	10395	51157
8	255	135135	816357
9	511	2027025	15050590
10	1023	34459425	314726297

Les éléments du tableau sont calculés comme suit :

Si le produit a n composants, le nombre possible de sous-assemblages N_n est :

$$N_n = (2N_{n-1}) + 1 \quad \text{avec } N_0 = 0$$

Le nombre de chemins pour le désassemblage complet C_n est :

$$C_n = (2n - 3) C_{n-1}$$

Le nombre total de chemins de désassemblage P_n est :

$$P_1 = 1$$

$$P_2 = \frac{1}{2} \binom{1}{2} P_1 + 1$$

$$P_3 = \binom{1}{3} P_2 + 1$$

...

$$P_{10} = \binom{1}{10} P_9 + \binom{2}{10} P_2 P_8 + \binom{3}{10} P_3 P_7 + \binom{4}{10} P_4 P_6 + \frac{1}{2} \binom{5}{10} P_5 P_5 + 1$$

Ce calcul demeure théorique mais nous fait réaliser la complexité potentielle de la détermination d'une séquence de désassemblage optimale. En réalité nous retrouvons beaucoup moins de séquences et cela pour plusieurs raisons. Les contraintes géométriques de déplacement, d'emplacement, et la forme des composants réduisent considérablement le nombre de séquences. L'ordre de désassemblage est aussi un facteur de réduction. Tous ces éléments doivent être considérés pour les stratégies de désassemblage. Les matrices de désassemblage et les arbres de désassemblage sont des outils importants pour les stratégies en prenant en considération ces contraintes et en les introduisant dans les algorithmes (section 4.2). Les forces et les différences entre ces algorithmes sont souvent dues à la façon donc ils peuvent travailler avec les outils de contraintes. La complexité d'un produit est donc plus souvent un problème de structure que de nombre de composants. Pour connaître la complexité de la structure du produit, nous pouvons utiliser certaines mesures.

3.3.2 Mesure de la complexité de la structure du produit

La complexité de la structure d'un produit est le principal problème des stratégies de désassemblage. Pour contourner ces problèmes de structures nous posons des hypothèses. Ces dernières nous permettent de trouver plus facilement la séquence de désassemblage optimale mais nous empêche de travailler avec certains produits plus complexes. Donc, pour choisir la stratégie appropriée nous ajoutons les contraintes de structure de produits en plus de nos besoins et des options de fin de vie. Pour quantifier la complexité du produit nous allons donc utiliser les trois dimensions suivantes [10] :

1. Complexité de réassemblage
2. Complexité de profondeur
3. Complexité de cheminement

Ces dimensions sont mesurées en fonction de plusieurs types de produit dans notre analyse de la complexité.

Complexité de réassemblage

La complexité de réassemblage est définie par le nombre de composants devant être remplacés pour réassembler complètement le produit et leur ordre d'assemblage. Dans le cas où différentes structures de produit seraient incluses dans une procédure de récupération pour la réutilisation, une évaluation de complexité doit être faite. En terme de complexité de réassemblage, le produit ayant un grand nombre de composants devant être remplacés est considéré comme plus complexe. La complexité de réassemblage peut être exprimée comme suit :

$$C_{rm} = \frac{NU_m}{NU_{\max}}$$

où :

C_{rm} est la complexité de réassemblage pour un produit de type m ;

NU_m est le nombre de composants qui doivent être remplacés pour le réassemblage du produit de type m ;

NU_{\max} est le nombre maximum de composants qui doivent être remplacés pour réassembler tous les types de produit dans notre environnement de travail.

Complexité de profondeur :

La complexité de profondeur est le nombre de niveaux de matériaux à traverser avant d'accéder à celui dont on a besoin (le niveau 0 représentant un composant accessible directement sur le produit). En terme de complexité de profondeur, un produit avec un grand nombre de niveaux pour atteindre ses composants est considéré plus complexe qu'un produit avec seulement quelques niveaux. La complexité de profondeur est exprimée comme suit :

$$C_{pm} = \frac{NL_m}{NL_{\max}}$$

C_{pm} est la complexité de profondeur pour un produit de type m ;

NL_m est le nombre de niveaux de profondeur pour un produit de type m ;

NL_{\max} est le nombre maximum de niveaux pour tous les types de produit dans notre environnement de travail.

Complexité de cheminement :

La complexité de cheminement est le nombre maximum d'opérations requises pour chaque composant dans un produit. Cette complexité d'un type de produit peut être évaluée en comparaison avec le nombre maximum d'opérations nécessaires pour tous les types de produits de notre environnement de travail. Plus le nombre d'opérations requises pour chaque composant est élevé pour un type de produit, plus le cheminement est considéré comme étant complexe. La complexité de cheminement est exprimée comme suit :

$$C_{cm} = \frac{\text{Max}\{R\}}{NR_{\max}} \quad \text{et} \quad R_j = \sum_{i=1}^i OF_{ij}$$

Où :

$\{R\}$ est la série de j composants du produit de type m ;

C_{cm} est la complexité de cheminement pour le produit de type m ;

R_j est le nombre d'opérations de cheminement requis pour le composant de type j ;

OF_{ij} est le facteur d'occurrence (probabilité) pour le composant j ;

NR_{\max} est le nombre maximum d'opérations de cheminement pour tous les composants dans tous les types de produits dans notre environnement de travail.

Ces trois dimensions de la complexité sont liées, alors la complexité d'un produit de type m est définie comme suit :

$$C_m = C_{rm} \times C_{pm} \times C_{cm}$$

Les valeurs de C_m sont entre 0 et 1 inclusivement. Plus la valeur de C_m est élevée plus la structure du produit est complexe.

La figure 3.3 nous permet de visualiser les résultats de la complexité pour trois types de produits.

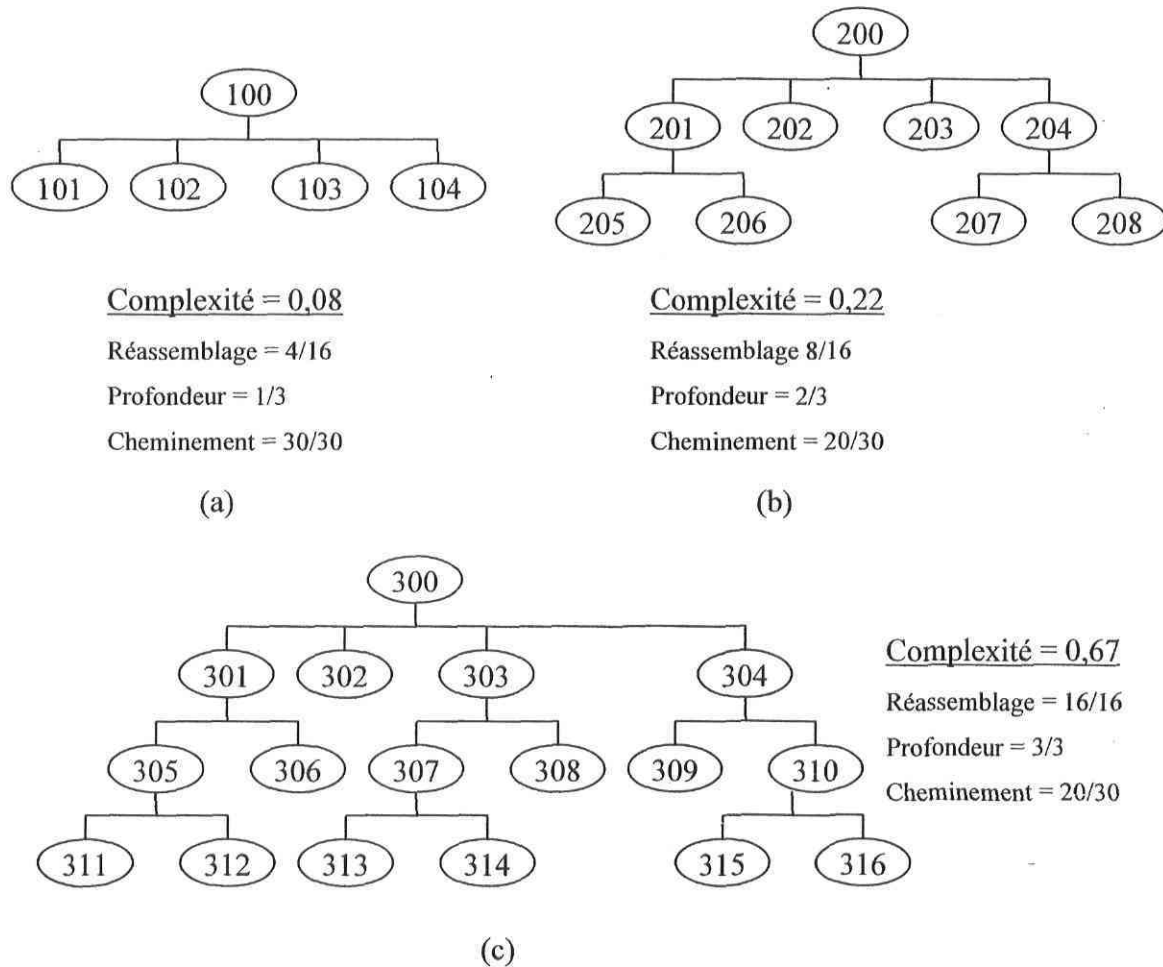


Figure 3.3 : *Comparaison de la complexité de trois types de produits*

Source : Guide, Srivastava, Kraus [10]

Cette mesure de la complexité est utilisée principalement pour faire une comparaison entre des produits. Chacune des dimensions nous donnent une information importante sur la complexité du produit à désassembler.

3.4 Choix de la stratégie de désassemblage

Une fois les besoins déterminés et les paramètres du produit complètement analysés, il est possible de choisir une stratégie de désassemblage.

Une liste des différents besoins possibles est présentée à la section 3.1. Pour les stratégies que nous allons étudier, plusieurs de ces besoins ou objectifs peuvent être atteints. Les différents besoins de notre étude sont : (1) l'optimisation des coûts de désassemblage, (2) l'optimisation des temps de désassemblage, (3) l'optimisation de la séquence, (4) le désassemblage partiel du produit, (5) le désassemblage complet du produit, (6) l'optimisation d'une ligne de désassemblage, (7) la maintenance du produit, (8) l'optimisation à la fois de l'assemblage et du désassemblage et (9) la prise en considération de l'aspect environnemental.

Les caractéristiques du produit sont aussi importantes dans les stratégies de désassemblages analysées. Les sections 3.2 et 3.3 nous permettent d'évaluer l'option utilisable pour la récupération de notre produit et nous informent sur sa complexité. Nous prenons donc en considération les niveaux de désassemblage et de complexité du produit : (a) la réparation, (b) la remise à neuf, (c) la nouvelle fabrication, (d) la cannibalisation, (e) le recyclage, (f) un produit avec beaucoup de composants, et (g) un produit avec des relations de priorité complexes. Un algorithme pouvant travailler avec des produits complexes peut normalement prendre en considération des produits simples, mais pas l'inverse. Des valeurs quantitatives peuvent être évaluées pour chacun de ces paramètres, il est donc facile de choisir la stratégie qui leur est associée.

Le tableau 3.3 présente les différentes stratégies en fonction des besoins du désassembleur et des paramètres du produit.

Tableau 3.3 : Stratégies en fonction des besoins et des paramètres du produit

	Catégories	Stratégies	Besoins et paramètres du produit
Chapitre 5	Stratégies de base	Planification du désassemblage avec ou sans composant défectueux	(2), (3), (5), (a), (b), (c)
		Abstractions géométriques	(3), (4), (a), (b), (c), (g)
		Apprentissage et raisonnement géométrique	(3), (5), (a), (b), (c), (d), (f)
		Relations de priorité entre les composants d'assemblage	(3), (4), (a), (b), (c), (d)
Chapitre 6	Stratégies utilisant les réseaux de Petri	Réseau de Petri avec relation de priorité complexe	(3), (5), (a), (b), (c), (d), (g)
		Réseau de Petri expert stochastique coloré et augmenté	(3), (5), (8), (a), (b), (c), (d), (g)
Chapitre 7	Stratégies considérant l'aspect environnemental et économique	Évaluation économique des opérations de désassemblage pour le recyclage, la remise à neuf et la réutilisation	(1), (2), (3), (5), (9), (b), (c), (d), (e)
		Réseau logique de désassemblage	(1), (3), (5), (9), (a), (b), (c), (d), (e)
		Algorithme génétique en considération de l'aspect économique et environnemental	(1), (3), (5), (9), (b), (c), (d), (e)
Chapitre 8	Stratégies liées à la maintenance et la réparation de produit	Algorithme génétique pour la maintenance	(3), (4), (7), (a)
		Désassemblage sélectif global	(3), (4), (7), (a)
Chapitre 9	Stratégie d'optimisation d'une ligne de désassemblage	Optimisation d'une ligne de désassemblage avec présence d'échec de tâches	(3), (6), (a), (b), (c), (d), (e), (f)

L'analyse des stratégies est effectuée en fonction des besoins principaux. Un ou plusieurs besoins peuvent être atteint par une seule stratégie. Toutes les stratégies visent directement ou indirectement l'optimisation des séquences peu importe les éléments qu'elles peuvent prendre en considération.

3.5 Collecte d'information nécessaire

Pour conclure sur l'analyse pré-désassemblage, nous devons faire une collecte de données pour pouvoir quantitatifs et qualitatifs les éléments nécessaires à l'utilisation de la stratégie choisie. Cette collecte prend en considération toute l'information sur le produit et son désassemblage. Les outils de collecte de données présentés ci-dessous, sont général. Il est possible que pour utiliser notre stratégie plusieurs types d'information ne soient pas nécessaires. Voici donc, deux outils pour la collecte de données qui peuvent être appliqués avec toutes les stratégies de désassemblage.

Nous pouvons classifier l'information requise en quatre catégories [49] :

- Information sur la composition du produit ;
- Information sur la magnitude et l'incertitude des retours de flux ;
- Information sur le marché pour réexaminer les produits, composants et matériaux ;
- Information sur la récupération actuelle du produit et la gestion des opérations.

3.5.1 L'outil de collecte de données

La collecte de données sur le produit est essentielle pour que nous soyons capable de travailler avec les méthodes de désassemblage proposées (chapitre 5 à 9). L'information nécessaire diffère d'une méthode à l'autre mais l'analyse des liaisons entre les composants, et le temps de séparation des composants est pratiquement toujours présent.

L'outil de prise de données comprend l'information de base sur le produit (la marque, l'âge, le type de fin de vie pour lequel il est destiné, le coût de l'intervenant, les sous-ensembles de composants et le nombre de composants à désassembler), le tableau d'analyse du désassemblage, la construction de la matrice de désassemblage et de l'arbre de désassemblage. Le tableau d'analyse du désassemblage comporte les éléments d'information suivants : le numéro de chaque composant, la quantité, le nombre minimum de composants, le type de tâche, le nombre de répétition de la tâche, les outils

nécessaires, l'accessibilité, le positionnement, la force nécessaire, le temps, l'information spécial, le coût de désassemblage, les composants précédents, les composants à désassembler avant et la destination de ce composant. Ce tableau peut être utilisé directement avec la carte de pointage DEI (index d'effort de désassemblage). Avec l'information recueillie dans cet outil et la construction de la matrice et de l'arbre de désassemblage, il est possible de travailler avec la plupart des méthodes de désassemblage. Cet outil est présenté dans l'**Annexe.B**.

Donc les données nécessaires pour l'évaluation sont :

- Temps pour chacune des étapes du désassemblage (pour tous les ordres de désassemblage possibles) ;
- Matériel nécessaire (coût et temps d'utilisation, pour tous les ordres de désassemblage possibles) ;
- Coût pour les différentes stratégies de fin de vie du produit (désassemblage destructif ou non, rejet) ;
- Temps et coûts engendrés par un ordre de désassemblage différent et/ou une façon différente pour faire le désassemblage ;
- Valeur résiduelle des pièces après le désassemblage ;
- Liens entre chacune des pièces (liens ET/OU, sous forme de tableau ou graphique) ;
- Regroupement des matériaux compatibles.

3.5.2 La carte de pointage DEI

Pour une analyse complète du désassemblage, la carte de pointage DEI est un outil très utile. Elle permet de détailler au maximum le désassemblage de chacun des composants ou sous-ensembles d'un produit. Les résultats donnent une information numérique sur le niveau de difficulté du désassemblage. La carte est séparée en sept catégories (temps, outils, fixations, accès, instruction, hasard et force). Chaque catégorie présente une échelle de résultats possibles. Ces résultats sont associés à un nombre. Le niveau de

difficulté de désassemblage est calculé en additionnant le nombre obtenu pour chacune des catégories. Plus le total est élevé plus le composant ou le sous-ensemble est difficile à extraire du produit. Il est ensuite possible de comparer les composants entre eux ou même entre différents produits. Cependant, l'utilisation des cartes de pointage DEI nécessite beaucoup de temps pour analyser les produits avec beaucoup de composants. Il est préférable d'avoir des produits avec un nombre réduit de composants ou travailler avec des sous-ensembles. Das, Yedlaraliah et Narendra [6] présentent un modèle multi-facteurs pour calculer les scores d'effort de désassemblage qui est représentatif du total des opérations de désassemblage d'un produit. Les sept facteurs suivants sont considérés dans l'index : (i) le temps, (ii) les outils, (iii) les fixations, (iv) l'accès, (v) les instructions, (vi) les produits hasardeux, et (vii) la force requise.

Voici donc, les détails pour la cotation de chacune des catégories du tableau :

1. Temps

Temps par étape de désassemblage, incluant le temps de réglage, le temps de manipulation et le temps actuel de désassemblage. Le temps de réglage inclut le temps de préparation de l'outillage et de l'équipement. Pour une opération de démontage, cela inclut le temps de déblocage, de desserrage et de déplacement d'une attache.

2. Outils

Aucun :	Quand aucun outil n'est nécessaire ;
Fusil à air :	Prise manuelle pneumatique pour vis et boulon ;
Outils mécaniques :	Outil commun que l'on retrouve dans une boîte à outil ;
Outils OEM ³ :	Outil que l'on a besoin de se procurer à partir des OEM ;
Outils spéciaux :	Outil unique conçu pour le désassemblage spécifique du composant ;
Improvisation :	Aucun outil prédéterminé et spécialement fabriqué. L'utilisateur doit improviser une solution pour le désassemblage.

³ OEM : Équipements spécifiques au manufacturier

3. Fixations

Aucune :	Quand aucune fixation n'est nécessaire ;
Une main :	Habituellement applicable pour des petits produits et n'impliquant qu'une main pour faire le désassemblage ;
Deux mains :	Implique habituellement que le produit soit levé et orienté dans la position désirée pour que les activités de désassemblage soient exécutées ;
Serre-joints :	Demande d'être fixé à un outil stable ;
Treuils :	Représente les fixations alimentées tel un treuil au-dessus du produit ;
Automatisation :	Représente l'équipement fixé ou programmable.

4. Accessibilité

Axe Z :	Quand l'accès est possible directement sur le produit ;
Axe X/Y :	Quand l'accès est sur le côté du composant ;
>15 cm profondeur :	Quand le point de désassemblage est à plus de 15 cm de profondeur de la surface du produit ;
Par en dessous :	Dans l'axe négatif de l'axe Z ;
Deux axes :	Quand le point de désassemblage est accessible avec un outil flexible ou tordu ;
Non visible :	Quand le désassemblage est dans le produit et non visible.

5. Instructions

Aucun :	Quand l'étape suivante du plan de désassemblage est évidente ou apparente ;
10-20s et >30s :	Écart de temps pour que le travailleur accède à l'étape suivante. Cela inclut l'observation visuelle des composants du produit ou dans une base de données ;
Groupe discussion :	Si les travailleurs ont besoin de discuter de la situation avec d'autres ou avec le superviseur ;

Contact OEM :	Un document avec les OEM doit être consulté avant de pouvoir continuer le désassemblage ;
Formation :	Une formation spéciale est nécessaire due à une étape de désassemblage difficile.

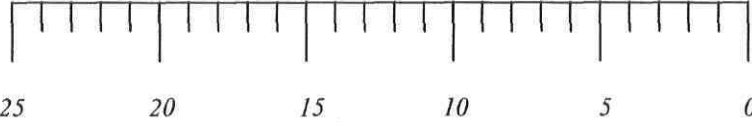
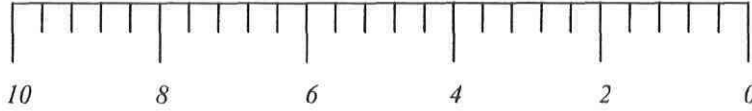
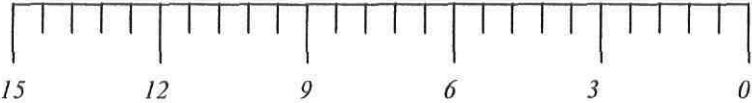
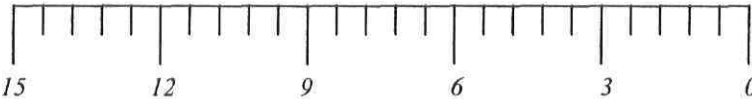
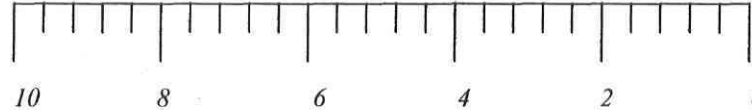
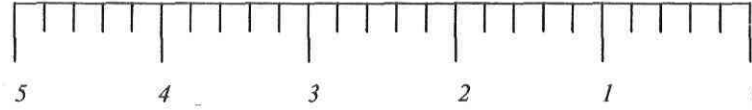
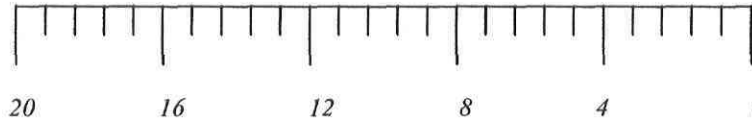
6. Hasard

Aucun :	Quand aucune protection n'est nécessaire ;
Gants :	La protection des mains, due à la présence d'une partie coupante ou de produits chimiques locaux ;
Masque/protection :	Quand il y a présence de particules volatiles lors du désassemblage ;
Protection contre le feu :	Nécessaire quand il y a présence de matériaux explosifs ou que l'on doit désassembler à l'aide d'un chalumeau ;
Alimentation d'air :	Quand l'atmosphère ambiante doit être nettoyée par un équipement spécial de ventilation ;
Habit la protection :	Quand le travailleur doit être recouvert en entier par un équipement protecteur.

7. Force

Axiale :	Force linéaire appliquée le long de l'axe primaire avec un outil ;
Torsion :	Une force de rotation est appliquée le long de l'axe primaire avec un outil ;
Orthogonale :	Cela implique que la force est perpendiculaire avec l'outil ;
Levier :	Cela demande un pied de biche pour le désassemblage ;
Impact faible :	Cela représente l'utilisation d'un marteau ou d'une force répétitive de moins de 10 kg d'intensité ;
Impact élevé :	Une force répétitive avec plus de 10 kg d'intensité.

Tableau 3.5 : La carte de pointage DEI [6]

<p>1. Temps (sec)</p>	<p>>210 140 90 50 25 <5</p> 	<p>Note :</p>
<p>2. Outils</p>	<p>Improvisation Spéciale OEM Mécanique Fusil à air Aucun</p> 	<p>Note :</p>
<p>3. Fixations</p>	<p>Automat. Treuil Serre-joins Deux mains Une main Aucune</p> 	<p>Note :</p>
<p>4. Accès</p>	<p>Non visible Deux Axes Par dessous >15 cm prof. Axe X/Y Axe Z</p> 	<p>Note :</p>
<p>5. Instructions</p>	<p>Formation Contact Groupe > 30 secs 10-20 secs Aucun OEM Discussion</p> 	<p>Note :</p>
<p>6. Hasard</p>	<p>Habit Alimentation Protection Masque/ protection d'air feu prot. bras</p> 	<p>Note :</p>
<p>7. Force - Défaire - Humain - Machine</p>	<p>Impact élevé Impact faible Levier Orthogonale Torsion Axiale</p> <p>> 25 kg 18 12 7 3 1</p> <p>> 150 kg 110 80 55 38 25</p> 	<p>Note :</p>

4 Théorie et outils liés aux stratégies

Pour bien comprendre la structure de base des stratégies de désassemblage, nous devons définir les représentations du produit possibles dans leur analyse, et comprendre deux outils essentiels, les matrices de désassemblage et les arbres de désassemblage.

4.1 Représentations du produit

La représentation du produit joue un rôle majeur dans l'utilisation des méthodes de désassemblage. Souvent, toute la méthode d'optimisation est basée sur la façon dont nous représentons le produit. La compréhension de la procédure et de son résultat est directement liée à l'aspect représentatif du produit. Quelles que soient les méthodes utilisées pour effectuer le désassemblage, un ou plusieurs types de ces représentations sont nécessaires. Torres, Puente et Aracil [51] énumèrent la plupart des types de représentation :

- Utilisation d'un graphique d'étiquette comprenant quatre types différents d'étiquettes (composants, contacts, liens, relations) dans lequel les pièces solides du produit sont les composants ;
- Utilisation de la représentation hiérarchique du produit par un graphique dans lequel les nœuds représentent les liens entre les surfaces des composants ;
- Utilisation des caractéristiques géométriques du produit. Avec cette information, géométrique, topologique et technologique, les restrictions peuvent être identifiées ;
- Approche basée sur la génération d'un arbre avec chacune des séquences de désassemblage possibles ;
- Utilisation du graphique ET/OU qui est une représentation compacte du produit donnant les relations entre les composants ;

- Utilisation d'un graphique de fabrication avec les données géométriques, topologiques et fonctionnelles du produit, lequel représente les relations de contacts et les composants du produit ;
- Utilisation d'un graphe joint-composant, où les nœuds correspondent aux composants, et où les arêtes représentent les joints entre les composants ;
- Approche basée sur la définition des relations entre les groupes de composants ;
- Utilisation de graphes orientés où les nœuds sont les composants et les arcs matérialisent les relations entre les composants.

La plupart de ces représentations sont en lien ou très semblables à l'outil général qu'est l'arbre de désassemblage (voir section 4.2.2).

4.2 Outils de base pour l'utilisation des stratégies

Les deux outils de base pour le désassemblage sont les matrices et les arbres de désassemblage. Il existe plusieurs façons de les nommer et de les représenter mais le principe reste le même. Ce sont des outils utilisés par toutes les stratégies de désassemblage étudiées dans les chapitres 5 à 9, sauf pour quelques exceptions. Les deux outils demandent de l'information tirée des documents de collecte de données sur les produits (section 3.5). Donc pour bien comprendre la structure des algorithmes et des méthodes présentées voici les détails de construction des matrices et des arbres de désassemblage.

4.2.1 Matrice d'interférence et de désassemblage

Plusieurs articles décrivent avec plus ou moins de détails les matrices d'interférence et de désassemblage. La meilleure représentation est effectuée par Huang et Huang [18].

Matrice de priorité de désassemblage (DPM)

Ces matrices représentent des relations géométriques entre les composants d'un produit. Les trois types de relations sont : ET, OU, et relations ET/OU complexes. Nous pouvons définir une DPM comme $DP=\{dp_{gh}\}, g,h=1,\dots,k$ où k est le nombre de composants et g, h sont des composants :

$$Dp_{gh} = \begin{cases} 0, \text{l'enlèvement de } g \text{ ne précède pas l'enlèvement de } h \\ 1, \text{l'enlèvement de } g \text{ (ET) précède l'enlèvement de } h \\ d, \text{l'enlèvement de } g \text{ (OU) précède l'enlèvement de } h \end{cases}$$

La relation entre deux composants dans un assemblage définie dans un ordre possible de détermination de séquence de désassemblage. Si le composant e_i de l'assemblage n'interfère pas avec un autre composant e_j dans la direction de l'axe k , le composant e_i peut être désassemblé librement du composant e_j dans la direction de l'axe k . Conséquemment le composant e_j peut être désassemblé du composant e_i dans la direction de l'axe $-k$. Si le composant e_i qui bouge dans la direction de l'axe k interfère avec un autre composant e_j , alors le composant e_j bouge dans la direction de l'axe $-k$ va à son tour interférer avec le composant e_i . Les matrices d'interférences dans la direction de l'axe k peuvent être générées comme :

$$H_k = \begin{matrix} & \begin{matrix} e_1 & e_2 & \dots & e_n \end{matrix} \\ \begin{matrix} e_1 \\ e_2 \\ \dots \\ e_n \end{matrix} & \begin{bmatrix} 0 & h_{12k} & \dots & h_{1nk} \\ h_{21k} & 0 & \dots & \dots \\ \dots & \dots & 0 & \dots \\ h_{n1k} & h_{n2k} & \dots & 0 \end{bmatrix} \end{matrix} \quad (1)$$

Les éléments dans la ligne e_i de la matrice (1) représentent la relation d'interférence du composant e_i avec tous les composants dans la direction de l'axe k , et les éléments dans la colonne e_j représentent la relation d'interférence du composant e_j avec tous les

composants e_i et e_j , l'élément h_{ijk} dans la matrice est 1. Vice versa, l'élément h_{ijk} dans la matrice est 1.

Matrice de désassemblage

Les matrices d'interférences dans les directions des axes x , y , et z comme illustrées dans la matrice (1) peuvent être combinées avec la matrice de désassemblage, en utilisant un système binaire comme suit :

$$A = \begin{matrix} & e_1 & e_2 & \dots & e_n & \text{résultats} \\ \begin{matrix} e_1 \\ e_2 \\ \dots \\ e_n \\ \text{résultats} \end{matrix} & \left[\begin{array}{cccc} h_{11z}h_{11y}h_{11x} & h_{12z}h_{12y}h_{12x} & \dots & h_{1nz}h_{1ny}h_{1nx} \\ h_{21z}h_{21y}h_{21x} & h_{22z}h_{22y}h_{22x} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ h_{n1z}h_{n1y}h_{n1x} & h_{n2z}h_{n2y}h_{n2x} & \dots & h_{nnz}h_{nny}h_{nnx} \end{array} \right]_{BIN} & \begin{matrix} h_{az}h_{by}h_{cx} \\ h_{dz}h_{ey}h_{fx} \\ \dots \\ h_{gz}h_{hy}h_{lz} \end{matrix} \end{matrix} \quad (2)$$

Les éléments h_{ijz} , h_{ijy} et h_{ijx} des matrices H_z , H_y et H_x , respectivement, dans l'équation (1) sont le chiffre de gauche, le chiffre du milieu et le chiffre de droite, de l'élément $h_{ijz}h_{ijy}h_{ijx}$ dans la matrice de désassemblage. Si un système de décimal est utilisé, l'élément dans la matrice de désassemblage est $h_{ijz}x2^2 + h_{ijy}x2^1 + h_{ijx}x2^0$ et la matrice de désassemblage est :

$$A = \begin{matrix} & e_1 & e_2 & \dots & e_n \\ \begin{matrix} e_1 \\ e_2 \\ \dots \\ e_n \end{matrix} & \left[\begin{array}{cccc} h_{11z} \cdot 2^2 + h_{11y} \cdot 2 + h_{11x} & h_{12z} \cdot 2^2 + h_{12y} \cdot 2 + h_{12x} & \dots & h_{1nz} \cdot 2^2 + h_{1ny} \cdot 2 + h_{1nx} \\ h_{21z} \cdot 2^2 + h_{21y} \cdot 2 + h_{21x} & h_{22z} \cdot 2^2 + h_{22y} \cdot 2 + h_{22x} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ h_{n1z} \cdot 2^2 + h_{n1y} \cdot 2 + h_{n1x} & h_{n2z} \cdot 2^2 + h_{n2y} \cdot 2 + h_{n2x} & \dots & h_{nnz} \cdot 2^2 + h_{nny} \cdot 2 + h_{nnx} \end{array} \right]_{DEC} \end{matrix} \quad (3)$$

Les éléments dans la matrice de désassemblage indiquent l'interférence d'un composant avec les autres composants du produit. Si les chiffres à droite de tous les éléments h_{ijx} , dans la ligne e_i sont égaux à zéro, alors le composant e_j ne doit pas interférer dans la direction de l'axe x avec le composant e_i si le composant e_i peut être désassemblé librement dans la direction de l'axe x du composant e_j . Si le chiffre de droite de chaque

élément dans la ligne e_i est égal à 1, alors le composant e_i ne peut pas être désassemblé dans la direction de l'axe x . Si les chiffres de droite de tous les éléments h_{ijx} , dans la colonne e_j sont égaux à zéro, alors le composant e_j peut être désassemblé librement dans la direction de l'axe $-x$.

Génération de la séquence de désassemblage

L'état d'un assemblage avec n composants est exprimé comme un nœud (figure 4.1). Si le composant e_i dans l'assemblage peut être désassemblé dans la direction de l'axe $+k$, alors l'assemblage sans le composant e_i et le composant désassemblé e_i dans la direction de l'axe $+k$, est ajouté au-dessous du nœud précédent comme montré dans la figure 4.1. Le symbole $+k$ entre accolade représente la direction de l'axe de désassemblage du composant e_j .

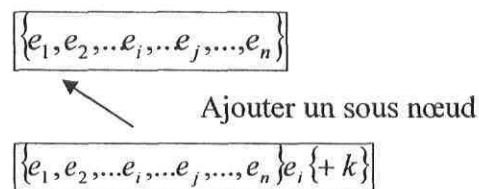


Figure 4.1 : Ajout d'un sous nœud

Source : Huang et Huang [18]

Désassembler le composant e_i et éliminer la ligne correspondante et la colonne pour le composant e_i de la matrice de désassemblage, équation (2) devient :

$$A = \begin{array}{c} e_1 \\ e_2 \\ \dots \\ e_n \\ \text{résultats} \end{array} \begin{array}{c} e_1 \quad e_2 \quad \dots \quad e_n \\ \left[\begin{array}{cccc} h_{11z} h_{11y} h_{11x} & h_{12z} h_{12y} h_{12x} & \dots & h_{1nz} h_{1ny} h_{1nx} \\ h_{21z} h_{21y} h_{21x} & h_{22z} h_{22y} h_{22x} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ h_{n1z} h_{n1y} h_{n1x} & h_{n2z} h_{n2y} h_{n2x} & \dots & h_{nnz} h_{nny} h_{nnx} \end{array} \right] \text{BIN} \\ \begin{array}{c} h_{mz} h_{oy} h_{px} \quad h_{qz} h_{uy} h_{sx} \quad \dots \quad h_{tz} h_{uy} h_{vx} \end{array} \end{array} \begin{array}{c} \text{résultats} \\ h_{az} h_{by} h_{cx} \\ h_{dz} h_{ey} h_{fx} \\ \dots \\ h_{gz} h_{hy} h_{iz} \end{array}$$

Les lignes tirées représentent la rangée et la colonne éliminées. L'opération Booléenne 'OU' pour la simplification de la matrice de désassemblage est encore utilisée pour continuer les procédures de désassemblage. S'il y a plus qu'un composant a ce nœud et qu'aucun composant ne peut être désassemblé dans le futur, ce nœud peut être éliminé automatiquement. Si seulement un composant existe à ce nœud, il est impossible de désassembler ce nœud dans le futur.

Donc, il existe des contraintes devant être considérées lors de la génération des séquences de désassemblage. Deux symboles «>» et «-» sont utilisés pour contraindre le désassemblage. Le symbole $e_j > e_k$ signifie que le composant e_j doit être désassemblé avant le composant e_k . Il est nécessaire d'investiguer si le composant e_j existe encore après que le composant e_k soit désassemblé. Si le composant e_j existe encore, le désassemblage en sous nœud est exclu (figure 4.2). Le symbole entre parenthèse représente les directions dans lesquelles le composant e_j peut être désassemblé, c'est-à-dire les directions d'axes $+k$ et $-k$.

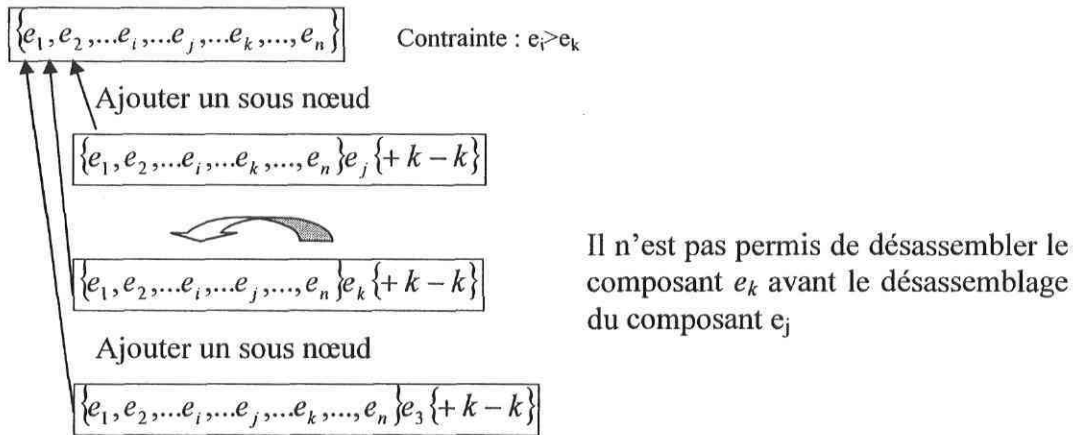


Figure 4.2 : Génération de sous nœuds avec contrainte

Source : Huang et Huang [18]

Le symbole $e_j - e_k$ signifie que seulement le composant e_k et aucun autre ne doit être désassemblé directement après le composant e_j . Cependant, le composant e_j n'a pas nécessairement besoin d'être désassemblé avant le composant e_k . Si le composant e_k a été désassemblé, alors la contrainte n'existe plus. Il est nécessaire d'examiner l'existence du

composant e_k après que le composant e_j soit désassemblé. Si le composant e_k existe, il doit être désassemblé directement après le composant e_j , et le sous-nœud du désassemblage du composant e_k doit suivre comme illustré à la figure 4.3.

Si le composant e_j doit toujours être désassemblé avant que le composant e_k soit désassemblé, et le composant e_k doit uniquement être désassemblé exactement après que le composant e_j soit désassemblé, les symboles $e_j > e_k$ et $e_j - e_k$ sont utilisés simultanément. Toutes les séquences de désassemblage peuvent être déterminées en accord avec leurs contraintes par l'utilisation de la méthode de première profondeur de recherche.

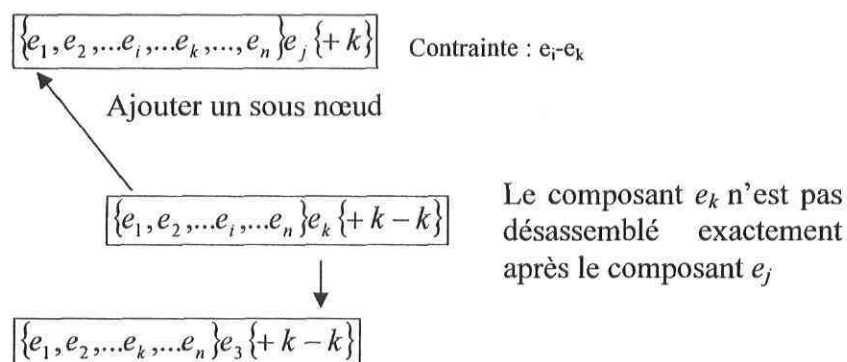


Figure 4.3 : Génération d'un sous nœud avec contrainte

Source : Huang et Huang [18]

Vision globale de la construction des matrices

Étapes de construction (figure 4.4)

1. Matrice de liens et d'interférence.

Une matrice par dimension donc trois matrices (matrice binaire).

Dimension de la matrice (imp. : matrice carrée) = nombre de composants.

2. Construction de la matrice globale de désassemblage

1. Matrice binaire (trois chiffres par combinaison)

2. Matrice décimale (transformation du binaire en décimale)

3. Utilisation des opérations Booléennes pour déterminer le ou les composants que l'on peut désassembler.

4. Nous enlevons le composant et nous retranchons les éléments qui lui correspondent dans la matrice (obtention d'une matrice avec une dimension de moins)

5. Nous analysons des nouvelles possibilités de désassemblage.

6. Nous répétons les étapes 3 à 5 jusqu'au désassemblage complet.

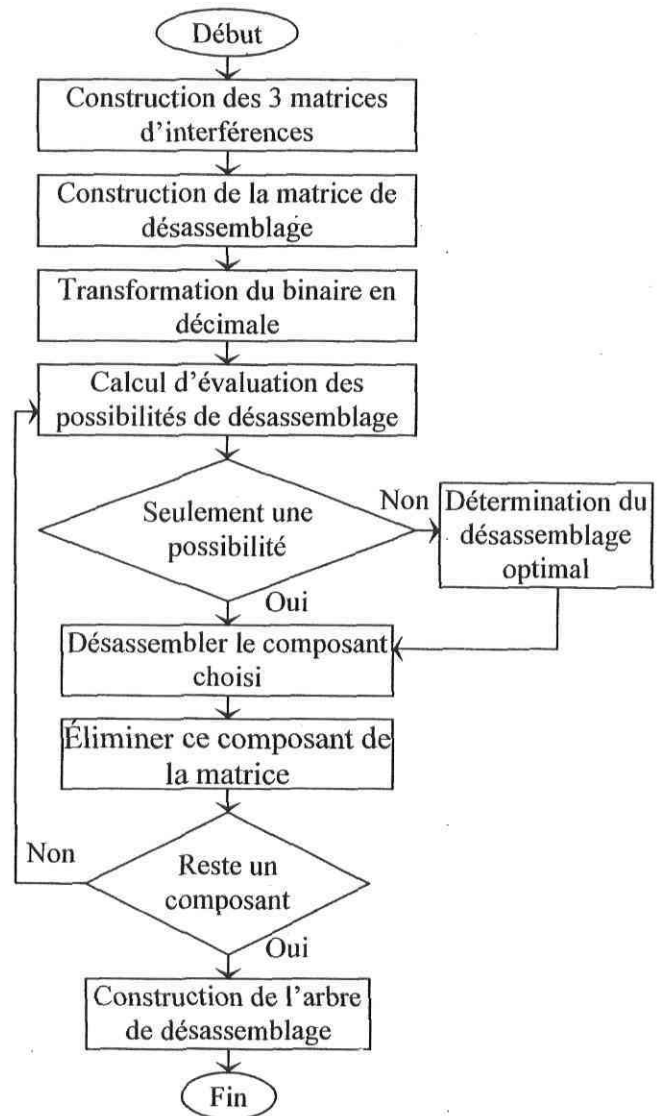


Figure 4.4 : *Algorithme de construction des matrices*

Exemple de construction des matrices

Voici un exemple de construction de matrices du Montant de dos télescopiques 0 degré, du fauteuil manuel Prima, d'Orthofab.

Nombre de pièces : 5

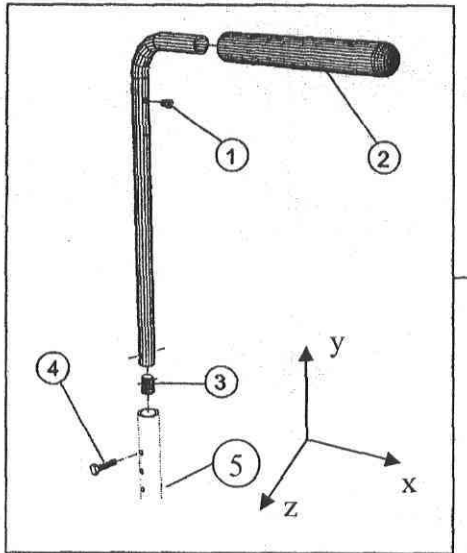


Figure 4.5 : Montant de dos télescopiques du fauteuil Prima d'Orthofab

Source : Catalogue de pièces Orthofab, fauteuil manuel Prima (octobre 1999)

Matrices d'interférence :

$$H_x = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

$$H_y = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

$$H_z = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

Matrice de désassemblage :

$$\begin{array}{c}
 \begin{array}{ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
 1 & \left[\begin{array}{ccccc}
 000 & 111 & 110 & 110 & 000
 \end{array} \right] & 110 \\
 2 & \left[\begin{array}{ccccc}
 111 & 001 & 111 & 111 & 000
 \end{array} \right] & 111 \\
 A = 3 & \left[\begin{array}{ccccc}
 111 & 011 & 000 & 111 & 000
 \end{array} \right] & 111 \\
 4 & \left[\begin{array}{ccccc}
 111 & 010 & 011 & 000 & 111
 \end{array} \right] & 111 \\
 5 & \left[\begin{array}{ccccc}
 000 & 000 & 000 & 101 & 000
 \end{array} \right] & 101 \\
 & 111 & 111 & 111 & 111 & 111
 \end{array}
 \end{array}$$

$1\{x\}$ ou $5\{y\}$

On enlève $a\{x\}$ et on enlève $e\{y\}$

$$\begin{array}{c}
 \begin{array}{cccccc}
 & |1| & 2 & 3 & 4 & |5| \\
 1 & \left[\begin{array}{ccccc}
 000 & 111 & 110 & 110 & 000
 \end{array} \right] & 110 \\
 2 & \left[\begin{array}{ccccc}
 111 & 001 & 111 & 111 & 000
 \end{array} \right] & 111 \\
 A = 3 & \left[\begin{array}{ccccc}
 111 & 011 & 000 & 111 & 000
 \end{array} \right] & 111 \\
 4 & \left[\begin{array}{ccccc}
 111 & 010 & 011 & 000 & 111
 \end{array} \right] & 111 \\
 5 & \left[\begin{array}{ccccc}
 000 & 000 & 000 & 101 & 000
 \end{array} \right] & 101 \\
 & 111 & 111 & 111 & 111 & 111
 \end{array}
 \end{array}$$

Nouvelle matrice :

$$\begin{array}{c}
 \begin{array}{ccc}
 & 2 & 3 & 4 \\
 2 & \left[\begin{array}{ccc}
 001 & 111 & 111
 \end{array} \right] & 111 \\
 A = 3 & \left[\begin{array}{ccc}
 011 & 000 & 111
 \end{array} \right] & 111 \\
 4 & \left[\begin{array}{ccc}
 010 & 011 & 000
 \end{array} \right] & 011 \\
 & 011 & 111 & 111
 \end{array}
 \end{array}$$

$4\{z\}$ ou $2\{-z\}$

Et ainsi de suite jusqu'au désassemblage complet du produit.

4.2.2 Arbre de désassemblage

L'arbre de désassemblage est un outil important dans la représentation et la construction des séquences de désassemblage. C'est un outil visuel important qui peut être utilisé comme résultat des algorithmes. Ils sont directement liés avec les matrices d'interférences. Ils représentent de façon hiérarchique les différentes interférences entre les composants d'un produit ou d'un sous-ensemble. Pour les produits complexes, l'arbre principal peut être décomposé en plusieurs arbres qui représentent les sous-ensembles d'un produit. Les arbres de désassemblage peuvent aussi comprendre tous les chemins possibles pour le désassemblage ou les chemins optimaux pour aller chercher chacun des composants. Ils sont pratiques comme outil de travail pour la personne effectuant le désassemblage ou la maintenance d'un produit. Ils indiquent de façon graphique le chemin à suivre. Étant un outil essentiel pour l'utilisation et la compréhension des algorithmes, la construction de ces arbres est donc nécessaire.

Théorie des graphiques de désassemblage

L'arbre de désassemblage se construit à partir de l'information sur les liaisons entre les composants du produit. Les nœuds représentent les composants ou les sous-assemblages et les arcs les opérations ou tâches de désassemblage. Les nœuds n'ayant pas d'arc entrant sont les composants en surface du produit, n'ayant aucune relation de priorité. Les nœuds qui n'ont pas d'arc sortant sont les derniers composants que nous pouvons aller chercher dans le produit, n'ayant aucune relation de subséquence. Sur les arcs, nous pouvons retrouver l'information sur le temps (b) pour réaliser le désassemblage des composants.

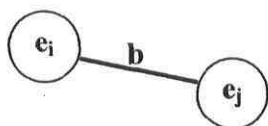


Figure 4.6 : Représentation : nœuds et arc

Dans la figure 4.6, e_i est désassemblé avant e_j et b est le temps de désassemblage de e_j . Généralement le chemin de la séquence de désassemblage se trace sur l'arbre.

La théorie des graphiques peut être appliquée pour :

- Établir une représentation distincte du plan de désassemblage;
- Trouver le plus court chemin pour accomplir le désassemblage.

Voici un exemple simple d'arbre de désassemblage.

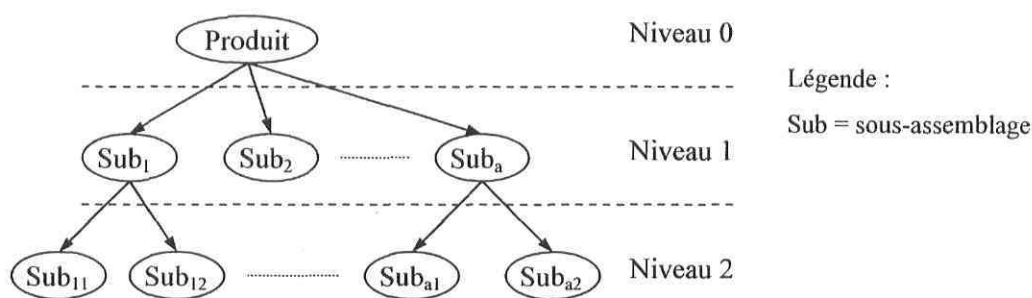


Figure 4.7 : Exemple d'arbre de désassemblage d'un produit

Source : Gao, Zhou et Caudill [8]

Il est important de noter qu'il existe plusieurs représentations différentes de l'arbre de désassemblage (section 4.1). Dans certaines méthodes, des arcs entre les nœuds peuvent représenter des activités de désassemblage. En informatisant l'arbre de désassemblage nous pouvons intégrer l'information complète des nœuds et sur les arcs, toute l'information sur les activités de désassemblage (ex : temps et coûts).

Faire un arbre de désassemblage avec un graphe réseau de Petri

Les règles suivantes sont appliquées pour générer le modèle d'assemblage des réseaux de Petri où la faisabilité d'assemblage et la décomposition est obtenue [24] :

- Chaque place dans le graphique réseau de Petri correspond aux composants impliqués durant les opérations d'assemblage/désassemblage ;
- Chaque transition dans le graphique du réseau de Petri reprend l'occurrence d'assemblage/désassemblage des composants à enlever ou à mettre ;
- Les têtes de flèches représentent directement l'occurrence d'un évènement impliqué dans l'assemblage/désassemblage.

Les règles suivantes sont appliquées pour dessiner un arbre ET/OU avec un graphique réseau de Petri [24] :

- Chaque nœud dans l'arbre ET/OU correspondent à la place dans le graphique réseau de Petri ;
- Transformation d'un nœud où les sous-nœuds sont représentés par une transition durant l'assemblage/désassemblage ;
- La relation OU correspond à la dernière activité de transition des opérations non similaires pour l'assemblage/désassemblage ;
- La relation ET correspond à deux activités ou plus de transitions simultanées des différentes opérations gérées ;
- Pour un état de changement interne, ajouter une place et deux états internes de transition.

5 Stratégies de base

Après avoir vu le cycle de vie, définit le désassemblage de produit, fait une analyse et une prise de données, analysé la complexité et les outils, voici maintenant le temps de désassembler le produit. Plusieurs outils et algorithmes sont utilisés pour permettre de trouver la ou les séquences de désassemblage optimale.

Au cours des dernières années le problème de planification des séquences de désassemblage pour différents types de configuration de produit, a été traité de plusieurs façons et à plusieurs niveaux. Il existe plusieurs méthodes et algorithmes, très différents les un des autres mais ayant toujours le même objectif, celui de trouver la séquence de désassemblage optimale. Les algorithmes présentés dans ce chapitre présentent une très bonne introduction aux problèmes du désassemblage. Ils comportent plusieurs hypothèses restrictives sur la structure du produit et de ses composants.

Analyse des stratégies de désassemblage de base ou de cas particuliers

Gao, Zhou et Caudill [8] proposent un algorithme pour déterminer les niveaux et séries de désassemblage dans une méthode intégrée pour la maximisation du profit d'un produit usagé par l'entremise du balancement des ressources inverties dans la procédure de désassemblage, le retour, et les impacts environnementaux ; la définition de la structure du produit, et l'utilisation de l'algorithme nommé « algorithme d'assignation de série » et intégration de l'algorithme dans un système informatique.

Taleb et Gupta [46] étudient le problème de prévision du désassemblage à partir de composants des produits caractérisés par la définition des structures de produits. Ils tiennent en compte l'existence de structures multiples de produits. Ils présentent deux algorithmes (« core algorithm » et « allocation algorithm ») utilisés ensemble pour obtenir un schéma de désassemblage. Les algorithmes déterminent la quantité et la

séquence d'opérations de désassemblage pour toutes les structures de produits dans l'ordre pour satisfaire la demande pour les composants.

Martinez, Pham et Favrel [33] présentent une méthode de génération de séquences de désassemblage optimales et l'utilisation du système de décision multi-agents. C'est une méthode de contrôle dynamique du désassemblage. Ils analysent les événements non prévus et des échecs dans les opérations de désassemblage. Ils utilisent la matrice de désassemblage (section 4.2.1) et l'arbre de désassemblage (section 4.2.2) pour l'analyse des liaisons et la représentation du produit.

Gupta et Taleb [17] présentent un algorithme pour faire une séquence de désassemblage des caractéristiques individuelles des composants d'un produit en définissant la structure du produit. Pour la production et la gestion de l'inventaire le système de planification de besoin en matériaux (MRP). Alors dans leur analyse la méthode inverse de MRP est directement appliquée pour l'analyse de la structure du produit à désassembler. C'est une méthode simple et intéressante pour les connaisseurs de la méthode MRP.

Lambert [29] fait la génération de la séquence de désassemblage optimale en utilisant la programmation linéaire. Pneli et Zussman [38] propose un algorithme pour le désassemblage optimal. C'est un algorithme de désassemblage relativement simple pour permettre son évaluation dès l'étape de conception.

Analyse des stratégies présentées

Dans les sections suivantes on va se concentrer sur quelques stratégies de désassemblage utilisant des procédures plus particulières. Les méthodes présentées sont : la planification du désassemblage avec ou sans composant défectueux [11,12], les abstractions géométriques [41], l'apprentissage et le raisonnement géométrique [19] et les relations de précédente entre les composants d'assemblage [51].

L'algorithme de Gungor, Gupta [11,12] est présenté en premier pour nous donner une idée du principe optimum par une analyse plus spécifique du produit. C'est un algorithme de base qui inclut le temps de désassemblage. Il peut être utilisé pour trouver la séquence optimale de désassemblage pour le recyclage et de la remise à neuf de produit. Ce qui est intéressant dans cette méthode c'est qu'elle permet de travailler avec la présence d'incertitude durant le désassemblage. Durant notre désassemblage non-destructif, il est possible d'utiliser un désassemblage destructif si un composant est défectueux (modification du temps de désassemblage).

Pour faire suite, quelques méthodes graphiques et géométriques sont présentées. Leurs utilisations et applications sont plus visuelles que les méthodes des réseaux de Petri et les algorithmes génétiques. Ce sont les algorithmes qui sont généralement spécifiques à un type de produit ou un but particulier le désassemblage. Contrairement aux autres méthodes, elles utilisent généralement d'autres outils de désassemblage que les matrices de désassemblage. Par exemple : Shyamsundar, Gadh [41], et Srinivasan, Gadh [43] utilisent le graphique topologique de désassemblage et les graphiques de liaisons. Dans les pages suivantes nous allons présenter trois méthodes. Le désassemblage basé sur les abstractions géométriques [41] (section 5.2), l'apprentissage et le raisonnement géométrique [19] (section 5.3), et les relations de priorité entre les composants d'assemblage [51] (section 5.4). Quelques autres méthodes géométriques et graphiques sont utilisées comme le « branch-and-bound algorithm » [13], l'utilisation directe des matrices et des arbres de désassemblage [55], et une méthode spécifique au problème de désassemblage de produits électromécaniques [27].

Shyamsundar, Gadh [41] présentent une méthode basée sur les abstractions géométriques. Ils utilisent deux abstractions dérivées des modèles géométriques d'assemblage qui déterminent la validité de l'assemblage, (1) le graphe topologique d'assemblage et (2) les séries de composants limites. La première abstraction, le graphe topologique d'assemblage, est un graphique où les nœuds représentent les composants d'assemblage et les arcs représentent les intersections non-nulles entre des coques convexes de paires de composants. La seconde abstraction, les séries de composants limites, représente les

composants qui sont liés dans l'assemblage. Ces composants limites sont typiquement ceux les plus accessibles en ce qui concerne le désassemblage. Ils définissent un algorithme qui utilise le graphe topologique de désassemblage pour déterminer les paires de composants interloqués. Si de telles paires de composants sont absentes, alors la séquence de désassemblage pour l'enlèvement des composants dans l'assemblage est déterminée par l'analyse de la série de composants limites pour le désassemblage. Cette méthode est donc un outil pour faire face au produit ayant des liens complexes entre les composants. Elle ne vise pas nécessairement le désassemblage complet du produit mais permet de travailler avec des problèmes dus au désassemblage de composants avec géométrie complexe et une liaison complexe entre deux composants.

Hu, Li, et Hu [19] présentent une méthode basée sur les outils l'apprentissage et de raisonnement géométrique. Ils construisent un algorithme qui utilise l'information provenant des paramètres de liaisons entre les composants du produit pour trouver les composants candidats pour le désassemblage et déterminer la planification du chemin de désassemblage. Une approche complète et précise vérifiant l'interférence est utilisée pour assurer qu'il n'y est aucune collision lors du désassemblage d'un composant. Ils utilisent une série de critères et d'heuristiques basés sur l'apprentissage, les contraintes, les relations entre les composants et l'analyse quantitative du désassemblage. Cette méthode, particulière, permet les changements de directions pour le désassemblage d'un composant visé. Elle contourne les obstacles (autres composants sur le chemin). Il est possible de l'utiliser pour la maintenance et toutes les autres options de récupérations.

Torres, Puente, et Aracil [51] présentent un algorithme basé sur une méthode de représentation de produits, qui nous montre les relations hiérarchiques entre les composants. Cet algorithme est utilisable pour établir une séquence de désassemblage partielle non destructrice d'un produit.

5.1 Séquence de désassemblage avec ou sans composant défectueux

Gungor, et Gupta [11,12] ont comme objectif de développer la structure d'un plan de séquence de désassemblage pour la manipulation des incertitudes qui peuvent survenir durant le cheminement dans le produit ou dans les procédures de désassemblage.

5.1.1 Méthodologie de la stratégie

La méthodologie est présentée en deux parties : (1) l'évaluation du temps total de désassemblage, et (2) la génération de la séquence de désassemblage.

Calcul du temps total de désassemblage

Le temps total de désassemblage (TTD) donne une mesure de rapidité d'une séquence de désassemblage donnée. Quelques facteurs affectent le TTD, le type de joint ou fixation et les changements de direction de désassemblage. La formule de calcul pour le temps total de désassemblage est la suivante :

$$TTD = \sum_{i=1}^n (T_{X_{i-1}, X_i}) \quad \text{où :} \quad T_{X_{i-1}, X_i} = t_{X_i} \times (1 + \alpha + \beta), \quad (i = 1, 2, \dots, n)$$

Où :

T_{X_{i-1}, X_i} : Temps ajusté de désassemblage de X_i désassemblé après X_{i-1} quand
 $i = 1$, alors $T_{X_{i-1}, X_i} = t_{X_i}$

t_{X_i} : Temps de désassemblage de X_i

α : Facteur de direction, qui est déterminé comme suit :

$$\alpha = \begin{cases} 0.1 - 1.0, & \text{si la direction de désassemblage change pour } X_{i-1} \text{ à } X_i \\ 0 & , \text{ autrement} \end{cases}$$

β : Facteur de changement de type de joint, qui est déterminé comme suit :

$$\beta = \begin{cases} 0.1 - 1.0, & \text{si le joint de type } X_i \text{ est différent du joint de type } X_{i-1} \\ 0 & , \text{ autrement} \end{cases}$$

n : Nombre de sous assemblage dans le produit

L'information nécessaire pour effectuer le calcul est la séquence de désassemblage du produit, le temps de désassemblage de chaque composant du produit, les directions de désassemblage et les types de joints des composants du produit.

Génération de la séquence de désassemblage

L'algorithme requière l'information suivante : les relations de priorité entre les composants du produit et le niveau de difficulté moyen pour chaque composant du produit qui définit la difficulté d'enlèvement des composants.

Étape 1: Générer la matrice de priorité de désassemblage (DP)

La matrice de priorité représente la base physique de la relation de priorité entre les composants du produit. Elle est générée par un dessin conçu par ordinateur (CAD) initial du produit par l'évaluation des composants d'obstruction dans la direction possible de désassemblage. Quand la matrice de priorité est générée, nous prenons en considération les trois types de relations suivantes : ET, OU, et ET/OU complexe. Dans la DP : $dp_{ij} = 1$, si le composant i précède en ET, j ; $dp_{ij} = d$, si le composant i précède en OU, j , où d est la direction de mouvement de désassemblage ; et $dp_{ij} = 0$, autrement.

Étape 2: Générer le plan de séquence de désassemblage optimale

Utiliser la DP et une sélection de critères d'optimisation, un Plan de séquence de désassemblage optimal (DSP) ou presque-optimal peut être généré. Nous utilisons l'heuristique qui génère un DSP presque optimal qui est utilisé pour la procédure de désassemblage. Nous définissons le DSP optimal comme un ordre sélectionné de composants, dénoté par Φ , et $|\Phi| = k$ est le nombre de composants dans le produit.

Étape 3: Exécuter le désassemblage avec présence d'incertitude

Dans cette étape, la procédure actuelle de désassemblage est exécutée en utilisant le DSP optimal généré dans la seconde étape. Quand une situation inattendue arrive, nous modifions le DSP et continuons la procédure de désassemblage. La notation suivante est utilisée dans le développement de l'étape 3.

Φ : Série de composants dans leur ordre de désassemblage ;

Ω : Série partielle de composants qui sont affectés par la situation inexplicée courante ;

Ψ : Série de tous les composants qui ne peuvent pas être désassemblés en utilisant un désassemblage non-destructif ;

DV : Vecteur binaire de composants défectueux, $dv_i = 1$ si le composant dans la i^e position de DSP est défectueux ou déformé, $dv_i = 0$ autrement. DV est initialement inconnu, et il est développé durant le désassemblage, cependant, pour illustrer la méthodologie, on suppose qu'il est inconnu.

Heuristique pour la génération d'une séquence de désassemblage

L'heuristique requière l'information suivante : les relations de priorité des composants du produit selon la considération et le rang de difficulté pour chaque composant du produit qui définit le niveau de difficulté d'enlèvement des composants (figure 5.1).

```
Étape 1 :    Créer deux liste, DS = {}, et UDS
              DS = {liste de composants désassemblés} (liste triée)
              UDS = {liste de composants non désassemblés} (liste non triée)
              Créer m tables, P(1)...P(m) en ordre ascendant de rangs de difficultés. (Où m est
              le nombre de type de joints dans le produit)

              i = 1
              WHILE UDS ≠ {} (où n est le nombre de sous assemblages dans le produit)
Étape 2 :    j = 1 (c.-à.d. début pour la plus petite table indexée)
              WHILE P(j)≠{}
                  Trouver le premier x dans P(j) qui n'a pas de prédécesseur
                  DS(i) = x, UDS = UDS - x, et mettre à jour des tables P(j) à travers P(m)
                  (c.-à.-d., éliminer la ligne de x et mettre à jour les relations de priorité
                  dans toutes les tables)

                  i = i + 1

                  IF chaque autre x trouvé tel que k = j + 1 aller à l'étape 3
              END (de WHILE)
              IF p(j)={ } tel que j=j+1 (max j = m) et répéter l'étape 2
Étape 3 :    IF P(k)≠ {}
                  Trouver le premier x dans P(k) qui n'a pas de prédécesseur
                  DS(i) = x, UDS = UDS - x, et mettre à jour les tables P(j) à travers P(m), tel que
                  i = i + 1 et aller à l'étape 2

                  IF chaque autre x trouver dans P(k) THEN k=k+1 (max k=m). Refaire l'étape 3
              END (de WHILE)
PRINT la séquence de désassemblage du produit par lecture DS de gauche à droite. STOP.
```

Figure 5.1 : Génération d'une séquence de désassemblage

Source : Gungor, et Gupta [11]

La figure 5.2 ajoute des détails pour l'étape 3 dans le cas de composants défectueux.

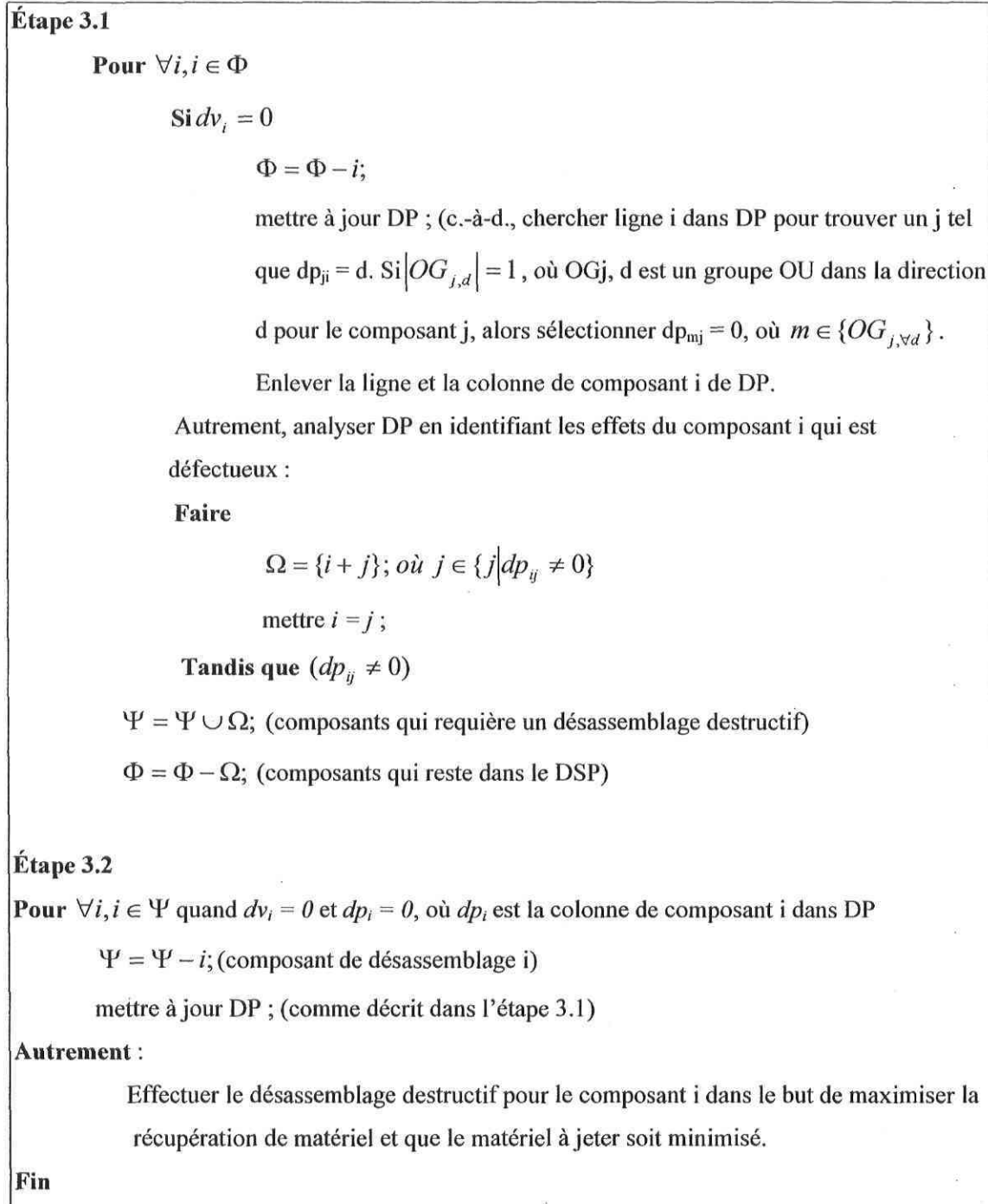


Figure 5.2 : Étape 3 avec présence de composants défectueux

Source : Gungor, et Gupta [12]

5.1.2 Exemple lié à la méthode

Pour illustrer l'application, nous utilisons une base d'ordinateur IBM PS/2 Model 30. Le tableau 5.1 fait le sommaire de l'information de la base d'ordinateur. Cet exemple n'inclut pas de composant défectueux.

Tableau 5.1 : Sommaire de la collecte de données (IBM PS/2 Model 30) [11]

# de sous-ensemble	Nom du sous-ensemble	Sous-ensembles prédécesseurs	Direction de désassemblage	Type de joint	Taux moyen de difficulté
1	Couvercle	-	-X	Vis (4)	2.7
2	Couvercle du lecteur de disquette	1	X	Fixation à pression (2)	2.0
3	Lecteur de disquette	2 \wedge 13	X	Fixation à pression (1)	3.6
4	Couvercle du disque dur	1	X	Fixation à pression (2)	2.0
5	Disque dur	4 \wedge 13	X	Fixation à pression (1)	3.8
6	Panneau arrière	1	-X	Fixation à pression (6)	3.9
7	Couvre-câbles	1	Y	Fixation à pression (2)	1.0
8	Carte réseau	1	Z	Vis (1)	2.3
9	Carte d'extension	7 \wedge 8	Y	Fixation à pression (embrochage)	2.0
10	SIMM RAM	1	Z	Fixation à pression (2x2)	3.3
11	Serrure principale	1	Y	Vis (2)	2.2
12	Unité de puissance	6 \wedge 7	Y	Vis (3) + Fixation à pression (2)	2.9
13	Câbles en plastique de raccordement	1	Y	Fixation à pression (embrochage)	1.3
14	Panneau avant	3 \wedge 5	X	Fixation à pression (7)	3.8
15	Support HDD et FDD	14	Y	Vis (1) + Fixation à pression (2)	2.2
16	Carte-mère	9 \wedge 12 \wedge 14	Z	Vis (7) + Fixation à pression (2)	3.1
17	Interrupteur	12 \wedge 14	X	Fixation à pression (2)	2.5

Quoi qu'il y a seulement 17 composants dans cet exemple, le nombre de séquences est très élevé. Nous sélectionnons aléatoirement une séquence de désassemblage faisable. Par exemple, la séquence de désassemblage SD = (1-11-7-8-6-12-9-4-13-5-2-3-14-15-10-16-17), où α et β sont 0.2 et 0.4, respectivement, et où TTD = 464.2 secondes en utilisant l'équation l'évaluation du temps total de désassemblage.

Nous utilisons l'heuristique de la méthode pour déterminer la séquence de désassemblage optimale. Nous avons deux types de joints dans cet exemple, $m = 2$. Comme décrit dans

les étapes de l'algorithme, la table P(1) contient la liste des composants avec des joints de fixation à pression, et la table P(2) contient la liste des composants fixés par une vis et les deux, vis et fixation à pression (voir Étape 1). Après l'exécution de l'algorithme, la séquence de désassemblage présenté dans le tableau 5.2, est trouvée. Nous utilisons l'équation de TTD pour déterminer le nouveau temps total de désassemblage, qui est de 405 secondes. La séquence de désassemblage de départ est 59 secondes plus longue que la séquence trouvée par l'utilisation de l'heuristique de séquence de désassemblage.

Tableau 5.2 : Résultat de l'exemple (IBM PS/2 Model 30) [11]

SD	DD	TJ	t_{X_i}	α	β	T_{X_{i-1}, X_i}
1	-X	JV	38	0	0	38
7	Y	FP	3	0.2	0.4	4.8
13	Y	FP	4	0	0	4
2	X	FP	8	0.2	0	9.2
4	X	FP	6	0	0	6
10	Z	FP	16	0.2	0	19.2
3	X	FP	20	0.2	0	24
6	-X	FP	35	0.2	0	42
5	X	FP	22	0.2	0	26.4
14	X	FP	35	0	0	35
11	Y	JV	20	0.2	0.4	32
8	Z	VFP	27	0.2	0.4	43.2
9	Y	FP	3	0.2	0.4	4.8
12	Y	VFP	33	0	0.4	13.2
17	X	FP	6	0.2	0.4	9.6
15	Y	VFP	15	0.2	0.4	24
16	Z	VFP	58	0.2	0	69.6

Légende : SD : séquence de désassemblage, DD : direction de désassemblage, TJ : type de joint, JV : joint vis, FP : fixation à pression, VFP : les deux, vis et fixation à pression

5.2 Les abstractions géométriques

Shyamsundar, Gadh [41] présentent deux abstractions pour identifier une séquence de désassemblage possible : le graphique topologique d'assemblage (ATG), et la série de composants limites.

Nomenclature

- C : Composant
- A : Assemblage
- G^A : Graphique topologique d'assemblage
- V : $\{C_i\} \in A$
- E : Intersection non nulle de surface convexe entre deux nœuds C_i et C_j dans G^A
- β_A : Composants limites de A
- H_A : Surface convexe de A

Hypothèses

- 1) Les $\{C_i\}$ doivent avoir des faces planes. Plusieurs modeleurs solides contiennent des fonctions intégrées pour aplanir les faces non-linéaires d'un C_i .
- 2) L'analyse est restreinte au désassemblage dans un contexte d'une translation de mouvement simple (dans une direction seulement).
- 3) Séparation simultanée d'assemblages de composants éloignés l'un de l'autres avec les deux mains, ne peut pas être considéré (enlèvement d'un composant à la fois).
- 4) Pour tous C_i et C_j , $i \neq j$, $C_i \cap C_j = \emptyset$.
- 5) L'entrée est supposée contenir les modèles solides de C_i et la location de chaque C_i dans un système de coordonnées global. Lequel peut être une information additionnelle d'assemblage disponible mais l'existence de cette information n'est pas garantie.

Construction du graphique topologique d'assemblage

Tout d'abord nous devons construire le graphique topologique d'assemblage (figure 5.3). Il va être utilisé pour définir la séquence d'assemblage. L'utilisation de ce graphique remplace l'utilisation des matrices d'interférence et de désassemblage. L'assemblage A comprend l'information suivante :

- (i) une liste de composants $C_i, i = 1 \dots n$;
- (ii) le nombre de composants dans l'assemblage, n ;
- (iii) une liste de surface convexe de $C_i, H_i, i = 1 \dots n$;
- (iv) le graphique ATG, G^A ;
- (v) la série de composants limites, β_A ;
- (vi) la surface convexe de A, H_A .

Cette l'information est obligatoire pour la construction et la compréhension du graphique et du reste de l'algorithme.

```
Construction_assemblage_topologie_graphe (assemblage A)
{
    Graphique  $A \times G^A = NUL$ 
    Pour  $i = 1 \text{ à } A \times n$   $A \times H_i = CoqueConvexe(A \times C_i)$ ;
        Pour  $(j = 1 \text{ à } A \times n)$ {
            Pour  $(i = 1 \text{ à } A \times n)$ {
                Si  $(i \neq j)$ {
                     $résult = Intersect(A \times H_i, A \times H_j)$ ;
                    Si (résult est non nul et contient un polyèdre ou plus)
                        Insérer i-côté dans  $A \times G^A$  ;
                    autrement si (résult est un face ou côté ou vertex)
                        insérer t-côté dans  $A \times G^A$  ;
                }
            }
        }
}
```

Figure 5.3 : Algorithme pour construire le graphique topologique d'assemblage

Source : Shyamsundar, Gadh [41]

La facilité de séparation DD_i de C_i en contact avec C_a, C_b, C_c, \dots en A est donnée par :

$$DD_i = S_{i,a} \cap S_{i,b} \cap S_{i,c} \dots$$

Si $DD_i = \text{NUL}$, C_i n'est pas désassemblé du reste A . L'algorithme pour calculer DD_i est défini dans la figure 5.4.

Séparabilité_Composants (C_i, A) {

$DD_i =$ toutes directions possibles ;

Pour (chaque C_k partage un côté avec C_i , $i \neq k$, dans $A \times G^A$) {

$S_{i,k} =$ Séparabilité (C_i, C_k) ;

$DD_i = DD_i \cap S_{i,k}$; }

Retourne DD_i }

Figure 5.4 : Algorithme pour calculer la facilité de séparation

Source : Shyamsundar, Gadh [41]

L'identification des composants limites (voir figure 5.6) implique la construction de la surface convexe de A et la détermination de tous $\{C_i\} \in A$ qui ont en commun un sommet, une arête ou une face avec la limite de A . En premier, les termes suivants ont besoin d'être définis :

- Assemblage limite : défini comme la limite de la face convexe H_A de l'assemblage A et est dénoté comme ∂H_A .
- Surface limite : la surface limite de C_i est défini comme la série de toutes faces, arêtes et sommets de C_i défini par ∂C_i .
- Composant limite : A composant est un composant limite s'il y a un $K_{i,A}$, où $K_{i,A}$ est défini comme une intersection de la surface limite de C_i , ∂C_i et limite d'assemblage, $\partial H_A : K_{i,A} = \{\partial C_i \cap \partial H_A\}$.

Pour déterminer l'ordre dans β_A , les composants limites sont groupés en trois séries :

- (i) série β_A^F : qui sont les $\{C_i\}$, $i = 1$ à n_F tel que chaque $K_{i,A}$ contient au moins une face. Donc, chaque ∂C_i à en commun au moins une face avec ∂H_A ;
- (ii) série β_A^{EE} qui sont les $\{C_i\}$, $i = 1$ à n_{EE} tel que chaque $K_{i,A}$ contient au moins une arête. Donc, chaque ∂C_i à en commun au moins une arête avec ∂H_A ;
- (iii) β_A^{VV} : qui sont les $\{C_i\}$, $i = 1$ à n_{VV} tel que chaque $K_{i,A}$ contient au moins un sommet. Donc, chaque ∂C_i à en commun au moins un sommet avec ∂H_A .

Le diagramme de Venn de la figure 5.5 permet de bien visualiser le classification de β_A .

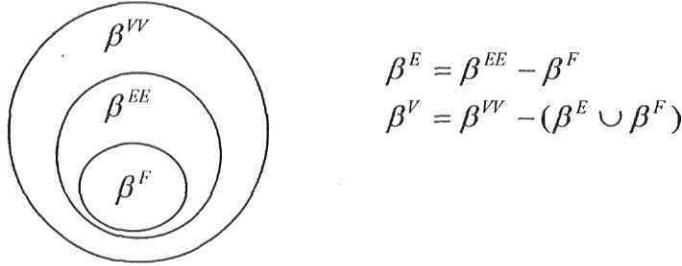


Figure 5.5 : Classification des composants limites

Source : Shyamsundar et Gadh [41]

$C_i \in \beta_A^F (= C_i^F, i = 1 \text{ à } n_F)$ sont ordonnancés en se basant sur le nombre de faces que chaque composant à en commun avec ∂H_A , dans l'ordre décroissant du nombre de faces en commun. $C_i \in \beta_A^E (= C_i^E, i = 1 \text{ à } n_E)$ et $C_i \in \beta_A^V (= C_i^V, i = 1 \text{ à } n_V)$ sont définis de la même façon que β_A^F excepté que β_A^E est basé sur le nombre d'arêtes et β_A^V est basé sur le nombre de sommets. Donc une série β_A est définie comme l'union ordonnancé ($U^\#$) des séries β_A^F , β_A^E , et β_A^V comme suit :

$$\begin{aligned} \beta_A &= \beta_A^F U^\# \beta_A^E U^\# \beta_A^V \\ &= \{C_1^F, \dots, C_{n_F}^F, C_1^E, \dots, C_{n_E}^E, C_1^V, \dots, C_{n_V}^V\} \\ &= \{b_A^1, b_A^2, b_A^3, \dots, b_A^{n_F+n_E+n_V}\} \end{aligned}$$

```

Identification_Limites_Composants (Assemblage A) {
Sélection_des_Composants_Limites  $A \times \beta_A = NUL$ ;
//la sélection des composants limites est initialisée à NUL//
 $\beta_A^F = NUL, \beta_A^{EE} = NUL, \beta_A^{VV} = NUL$ ;
 $\xi_A = CoqueConvexe(A)$ ;
Pour ( $i = 1$  à  $A \times n$ ) {
     $\kappa_{i,A} = NUL$ ;
     $\kappa_{i,A} = \{\delta(A \times C_i) \cap \delta(A \times \xi_A)\}$ ;
    Si ( $\kappa_{i,A} \neq NUL$ ) {
        Si ( $\kappa_{i,A}$  contient face(s))  $\beta_A^F = \beta_A^F \cup A \times C_i$ ;
        Si ( $\kappa_{i,A}$  contient côté(s))  $\beta_A^{EE} = \beta_A^{EE} \cup A \times C_i$ ;
         $\beta_A^{VV} = \beta_A^{VV} \cup A \times C_i$ ; //  $\kappa_{i,A}$  contient des surfaces
        convexes
    }
}
}
 $\beta_A^E = \beta_A^{EE} - \beta_A^F$ 
 $\beta_A^V = \beta_A^{VV} - (\beta_A^E \cup \beta_A^F)$ ;
L'ordre  $\beta_A^F$  dans le nombre décroissant de faces de chaque
composant partagé avec  $\delta(A \times \xi_A)$ ;
L'ordre  $\beta_A^E$  dans le nombre décroissant de arêtes de chaque
composant partagé avec  $\delta(A \times \xi_A)$ ;
L'ordre  $\beta_A^V$  dans le nombre décroissant de sommet de chaque
composant partagé avec  $\delta(A \times \xi_A)$ ;
 $A \times \beta_A = \beta_A^F \cup^\# \beta_A^E \cup^\# \beta_A^V$ ; // où  $\cup^\#$  est l'ordre des opérations
}

```

Figure 5.6 : *Algorithme de détermination des composants limites*

Source : Shyamsundar, Gadh [41]

Dans l'algorithme précédent, $K_{i,A}$ consiste seulement en des faces, arêtes ou sommets mais non à des volumes. Alors les données résultantes ne sont pas des volumes, une structure d'information qui considère l'intersection efficace est discuté si-dessous. En premier, la procédure pour déterminer $\partial C_i \cap \partial H_i$ est discutée.

Nous considérons C_1 et H_1 de la figure 5.9. Les faces et les arêtes dans H_1 et C_1 ne sont pas identiques. Aussi, la série de sommets de H_1 est nécessairement une sous-série de sommet de C_1 . Le sommet V_1 de C_1 est le sommet V_1' de H_1 . Les faces de C_1 qui sont présentes en H_1 sont nommées faces-personnelles, faces qui appartiennent à H_1 et n'ont pas de correspondances avec les faces de C_1 sont nommées faces-générées. Une terminologie similaire est utilisée pour les arêtes. E_1' est une arête-personnelle et E_3' est une arête-générée. La structure de données de la figure 5.7 est nécessaire pour stocker toute l'information des coques convexes.

<p>Coque Convexe H_i {</p> <p>Liste des sommets, $V_i, i = 1 \dots p.$</p> <p>Liste des boucles de sommets $L^i, i = 1 \dots s,$ qui forme une face ; }</p> <p>Vertex V_i {</p> <p>Coordonnées du point représentant le sommet ;</p> <p>Équations des arêtes incidentes du sommet ;</p> <p>Équations des faces incidentes du sommet ;</p> <p>Pointeur du sommet du composant ; }</p>
--

Figure 5.7 : *Algorithme de détermination des composants limites*

Source : Shyamsundar, Gadh [41]

Pour déterminer les arêtes-personnelles et les faces-personnelles de la coque convexe, l'algorithme de la figure 5.8 est utilisé.

Propre_Arêtes_et_Faces (H_i)

Pour (chaque arête (E) incident en V_k)

Si (équation d'incidence E est = équation de E incident au sommet correspondant de C_i) **Puis** E est identifiée comme une arête-personnelle.
autrement E est une arête-générée.

Pour (chaque arête (F) incident en V_k)

Si (équation d'incidence F est = équation de F incident au sommet correspondant de C_i) **Puis** E est identifiée comme une face-personnelle.
autrement E est une face-générée.

Figure 5.8 : Algorithme de détermination des arêtes et faces personnelles

Source : Shyamsundar, Gadh [41]

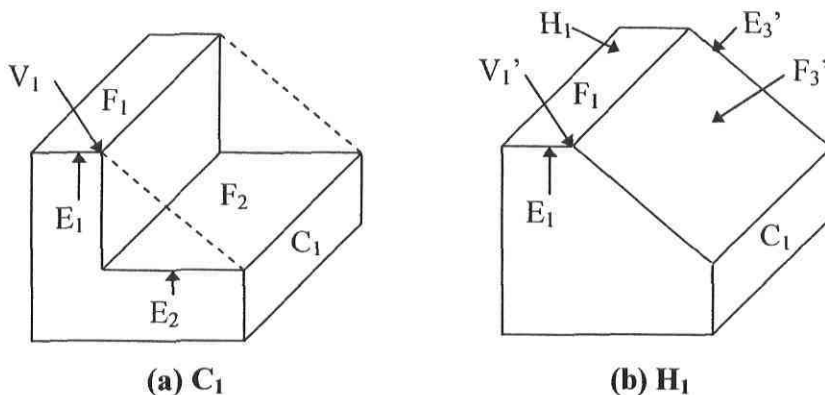


Figure 5.9 : Faces personnelles et générées

Source : Shyamsundar, Gadh [41]

Après avoir expliqué la procédure pour $\partial C_i \cap \partial H_i$, la procédure pour déterminer $K_{i,A} = \{\partial C_i \cap \partial H_A\}$ est maintenant à discuter. Pour déterminer H_A , les sommets de $\{C_i\}$ sont requis par l'algorithme **CoqueConvexe**. La base de données stockant H_A contient une liste de V_K . Chaque V_K de H_i a un pointeur de V_i dans un C_i ayant la même coordonnée que V_K . Au travers ce pointeur, l'information géométrique des arêtes et faces incidentes du sommet correspondant peuvent être déterminées. Une fois le H_A construit en utilisant **CoqueConvexe**, **Propre_Arêtes_et_Faces** est invoqué avec H_A comme argument.

L'algorithme de la figure 5.10 est utilisé pour calculer $K_{i,A}$.

```

Calcul_ $K_{i,A}$  ( $C_i, H_A$ ) {
Poser  $K_{i,A}$  ;
Pour ( $i = 1$  à  $n$ )  $K_{i,A} = \text{NUL}$  ;
Pour (tout sommet  $V_j$  de  $H_A$ ) {
 $C_i$  est le composant ayant le sommet correspondant de  $V_j$ .
//  $C_i$  peut être déterminé en utilisant le pointeur stocké
// dans la structure de données du sommet  $V_j$ .
Ajouter  $V_j$  à  $K_{i,A}$  .
Ajouter tous les arêtes-personnelles incidentes de  $V_j$  à  $K_{i,A}$ 
Ajouter tous les faces-personnelles incidentes de  $V_j$  à  $K_{i,A}$  }
}

```

Figure 5.10 : Algorithme pour calculer $K_{i,A}$

Source : Shyamsundar, Gadh [41]

Pour déterminer si A est valide, l'algorithme demande l'assemblage A comme donnée d'entrée. Premièrement, le G^A est construit (ligne 1). Si A a un ou plusieurs PIC⁴, A est invalide (ligne 5) et l'on sort du programme. S'il n'y a pas présence de PIC, β_A est déterminé (ligne 8). La capacité de séparation de b_A^k , dans β_A est déterminé (ligne 12). Si $DD^k \neq \text{NUL}$ peut être trouvé, alors le composant actuel est enlevé. Subséquemment, G^A est mis à jour et β_A est à redéfinir (ligne 15-20). Par cet algorithme, si $C_i \in A$ peut être désassemblé, alors A est valide (ligne 17). Si cependant, durant toute l'itération, tous les composants limites sont encore emboîtés alors A est invalide (ligne 25).

⁴ PIC : Composant avec contact d'emboîtement

```

Identifier_Valide_Assemblage (Assemblage A)
{
    1. Construction_Assemblage_Topologique_Graphe(A) ;
    2. Pour (chaque i-côté dans  $A \times G^A$  entre Ci et Cj){
    3. retour_valeur = Séparabilité(Ci,Cj) ;
    4. Si (retour_valeur = NULL){
    5.     return "Assemblage Invalide";
    6. }
    7. }
    8. Identification_Limites_Composants(A);
    9. size = Cardinal_Nombre( $A \times \beta_A$ );
    10. k=1 ;
    11. Pendant que ( $k \leq size$ ){
    12.     résultat = Composant_Séparabilité( $\zeta_A^k, A$ );
    13.     //  $\zeta_A^k$  : ke composant dans  $\beta_A$ 
    14.     Si (résultat  $\neq$  NUL) { // non enclenché
    15.         Décrémenter  $A \times n$  par 1;
    16.         Éliminer le nœud contenant Ci et tous les
            arêtes incidentes dans  $A \times G^A$  ;
    17.         Si ( $A \times n = 0$ )
    18.             retour "Assemblage valide";
    19.         Identification_Limites_Composants(A);
    20.         k = 1 ;
    21.     }
    22.     autrement
    23.         k = k + 1 ;
    24.     }
    25.     retour "Assemblage Invalide";
}

```

Figure 5.11 : Algorithme pour l'identification de la validité d'assemblage

Source : Shyamsundar, Gadh [41]

5.3 L'apprentissage et le raisonnement géométrique

Hu, Li, et Hu [19] proposent un modèle basé sur l'assemblage pour la planification de la séquence de désassemblage, et l'établissement d'un chemin de désassemblage pratique et correct pour le composant du produit en se basant sur le raisonnement et l'apprentissage géométrique. Voici donc la méthode proposée.

Hypothèses

1. Si l'assemblage ne peut être désassemblé, certaines étapes préalables doivent être réalisées pour rendre l'assemblage possible à désassembler.
2. A chaque étape de planification, seulement un composant est désassemblé de l'assemblage.
3. Durant la procédure de désassemblage, l'assemblage est stable (ne doit pas bouger d'aucune façon). Cela implique que quand un composant est enlevé de l'assemblage, il n'y a pas d'instabilité des composants restants. La méthode peut être utilisée en présence de composants naturellement stables ou que le reste de l'assemblage est bien fixé.
4. Considération seulement du modèle nominal. Supposition que le modèle est parfaitement assemblé et les composants parfaitement fabriqués. En réalité les composants diffèrent et cela implique des imperfections dans le modèle d'assemblage. Pour cela les variations dans le modèle d'assemblage ont une très petite influence dans la planification du désassemblage.
5. Considérer un premier mouvement transitionnel pendant le désassemblage du composant. Seulement dans le cas où il est impossible de désassembler le composant par mouvement transitionnel, la rotation du composant peut être considérée.

Procédure de planification du désassemblage

En ce basant sur un modèle d'assemblage, un système pour la séquence de désassemblage et le chemin de planification est développé. L'algorithme de planification de la séquence est défini dans la figure 5.12.

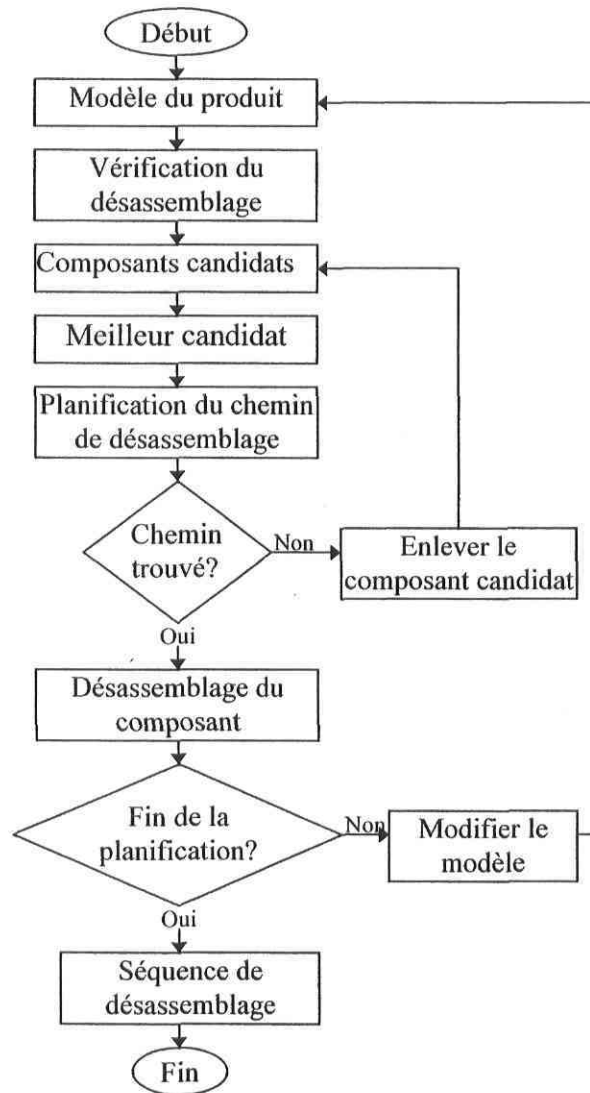


Figure 5.12 : *Procédure de planification du désassemblage*

Source : Hu, Li, et Hu [19]

Sélection du meilleur candidat

Pour la sélection du meilleur candidat le critère le plus important est le temps de désassemblage (étant directement lié au coût). La fonction de temps de désassemblage est basée sur quelques paramètres qui influencent la facilité de désassemblage du composant. De plus le temps de désassemblage indique la difficulté de l'opération de désassemblage correspondant. La fonction de temps de désassemblage proposée se définit comme suit :

$$D_{time} = h_{time} + w_c \times (c_{time} + s_{time})$$

où

D_{time} = temps de désassemblage h_{time} = temps de manipulation du composant

c_{time} = temps pour l'enlèvement des fixations w_c = poids pour l'accessibilité

s_{time} = temps pour l'enlèvement du composant

Les temps de manipulation, d'enlèvement, de déconnexion sont définis comme suit :

$$h_{time} = \left(1 + \sum_{i=1}^7 zw_i \right) \times sh_{time}$$

$$s_{time} = (1 + sw) \times ss_{time}$$

$$c_{time} = sc_{time} \times n$$

où :

sh_{time} : temps de manipulation

zw_5 : poids pour la rigidité

ss_{time} : temps d'enlèvement

zw_6 : poids pour la symétrie

sc_{time} : temps standard de déconnexion

zw_7 : poids pour stabilité

zw_1 : poids pour la masse

sw : poids pour la résistance de désassemblage

zw_2 : poids pour le volume

zw_3 : poids pour les irrégularités

n : nombre de fixations

zw_4 : poids pour la fragilité

L'algorithme de planification du chemin du composant est très important pour la planification de la séquence de désassemblage. Durant la planification du chemin, le système va trouver automatiquement un chemin de désassemblage pour le composant ou l'interaction complète par l'utilisateur, et aussi faire le test global s'assurant qu'il n'y a

aucune collision. L'algorithme permet de trouver la chemin optimal pour l'extraction du composant choisi. La procédure de planification est donnée dans la figure 5.13.

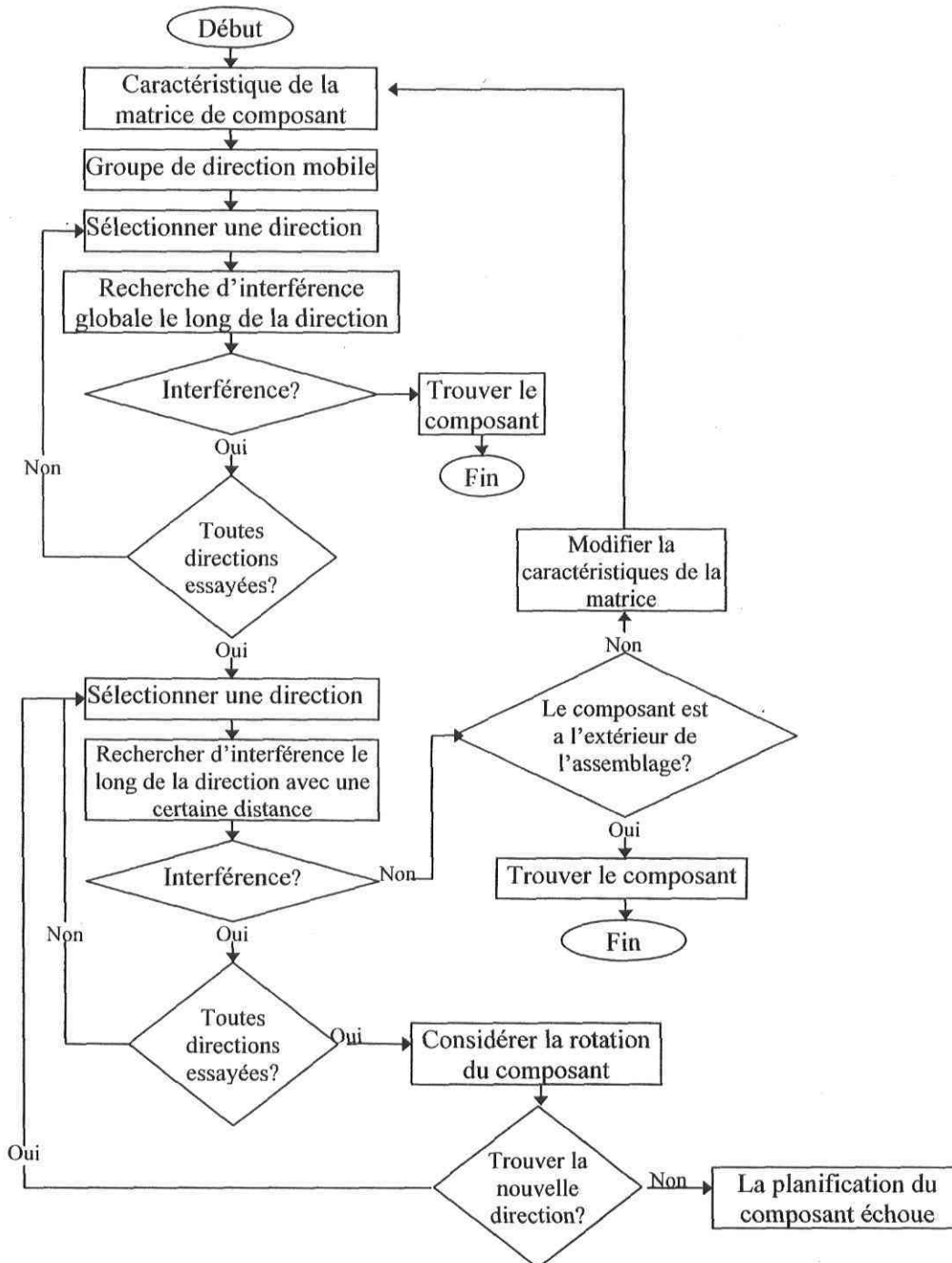


Figure 5.13 : Planification du désassemblage du composant

Source : Hu, Li, et Hu [19]

L'algorithme précédent permet d'extraire complètement chaque composant de l'assemblage. Il n'est cependant pas toujours nécessaire d'enlever tous les composants, lors de la maintenance ou de recyclage du produit par exemple. Hu, Li, Hu [19] ont développé un algorithme complémentaire permettant de faire un désassemblage partiel des composants. Ce dernier utilise les mêmes hypothèses énoncées précédemment dans cette section. L'algorithme de planification du chemin de la figure 5.14. Le composant que l'utilisateur vise à désassembler est nommé le composant cible final. Les composants ou sous-assemblages qui doivent être désassemblés avant le composant cible final à désassembler sont stockés dans la série de composants visés. La difficulté rencontrée ici c'est l'organisation de la série de composants visés et la recherche de la meilleur séquence de désassemblage pour désassembler le composant cible final efficacement. Les composants visés vont directement influencer la séquence de désassemblage générée.

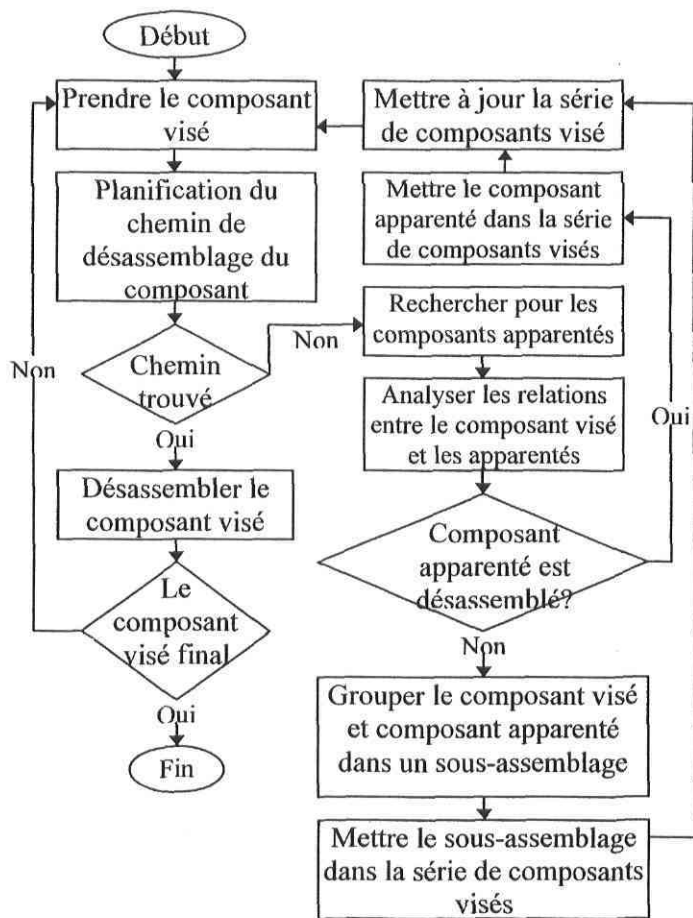


Figure 5.14 : Procédure de désassemblage partiel d'un composant

Source : Hu, Li, et Hu [19]

Cette méthode permet de trouver le chemin pour faire le désassemblage d'un composant. Il détermine étape par étape le déplacement du composant en prenant en considération toutes les contraintes dues à l'emplacement des autres composants. Les translations et les rotations peuvent être réalisées dans un mode 2D ou 3D. Le modèle prend aussi en considération les temps de désassemblage de chaque composant.

5.4 Les relations de priorité entre les composants

Torres, Puente, et Aracil [51] présentent un algorithme qui établit une séquence optimale de désassemblage partiel pour un produit en utilisant des techniques non destructrices et dérivé d'une nouvelle représentation intuitive du produit.

Séquence de désassemblage

Après la construction de l'arbre de désassemblage (section 4.2.2). Il est maintenant possible de calculer la séquence partielle correcte de désassemblage du produit par les étapes suivantes.

L'algorithme, qui sélectionne la séquence de désassemblage d'un composant c_j ou d'un sous-assemblage m_i du produit, est défini comme suit :

Étape 1 : Si un composant C_j est à désassembler, dans cette étape de l'algorithme rien ne doit être fait. Quand un sous-assemblage m_i est à désassembler, cependant, il est considéré comme un nouveau composant. Néanmoins, un composant peut être une partie de quelques sous-assemblages. Si les composants n qui sont une partie du sous-assemblage m_i et ils sont aussi une partie d'un autre sous-assemblage, le graphique va devoir être construit. Le sous-assemblage entier m_i est considéré comme un nouveau composant et forme une partie de l'ordre des sous-assemblages à lequel ce composant appartient. Par exemple, si le désassemblage m_1 dans le graphique de la figure 5.15(a) a été

désassemblé, le graphique est restructuré pour finalement exprimer la figure 5.15(b). Toutefois, pour chacun des composants du sous-assemblage m_i , il est nécessaire de regarder, s'il y a aussi formation d'une partie de l'ordre $m - 1$ sous-assemblages dans le graphique.

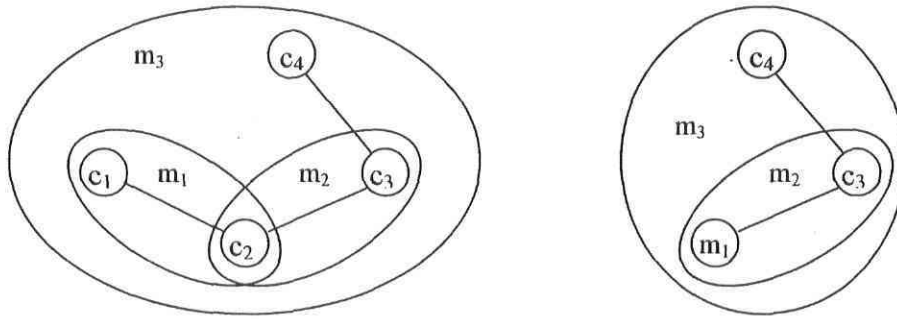


Figure 5.15 : *Reconstitution d'un sous-assemblage*

Source : Torres, Puente et Aracil [51]

Étape 2 : Identifier les sous-assemblages auxquels le composant c_j désassemblé appartient et identifier l'action d'enlèvement de ce composant. Toutefois, il est nécessaire de trouver le m sous-assemblage dans le graphique le composant c_j qui lui appartient, à un groupe S de sous-assemblages auquel le composant c_j appartient. Identifier des actions associées avec l'enlèvement du composant c_j à chacun des sous-assemblages du groupe S . Comme le composant c_j est dans tous les sous-assemblages au même niveau hiérarchique, les actions de désassemblage obtenues peuvent être données en parallèle ou dans l'ordre d'exécution pouvant être inter-changé. Cette étape a un coût de $O(m)$.

Étape 3 : Exécuter l'étape 2, considérant tous les sous-assemblages du groupe S comme un nouveau composant à être désassemblé. En exécutant cette étape, un nouveau groupe S' doit avoir été généré, et celui-ci est composé de sous-assemblages appartenant au groupe S . Cependant, il est nécessaire de répéter l'étape 2 jusqu'à un groupe composé uniquement de l'assemblage final.

Étape 4 : Cette procédure sert à sélectionner le chemin à être suivie dans l'ordre inverse, enlever les unions de fermeture (composants attachés entre eux) trouvées jusqu'à ce que nous arrivions au nœud source.

Cet algorithme permet de générer une séquence de désassemblage d'un produit automatiquement, basé seulement par l'information introduite par un opérateur connaissant le produit. Cette information peut être obtenue avec la procédure d'assemblage.

La méthode veut être utilisée sur un nouveau produit comme elle est utilisée pour le recyclage et la maintenance de produits existants qui ne sont pas conçus pour le désassemblage. Elle réalise une séquence de désassemblage permettant un désassemblage non destructif des produits.

Exemple d'utilisation de la méthode

La figure 5.16(a) présente le schéma d'une télécommande et son graphique. Elle est constituée de cinq composants et d'un élément de fixation. Le composant c_4 est le clavier de caoutchouc, c_3 est le circuit imprimé, c_2 et c_5 prennent en sandwich les deux précédent et une vis dans le haut de c_2 garde les quatre composants ensemble. Le composant c_1 est le couvercle pour la batterie. Les unions entre c_2 - c_3 , c_3 - c_4 , et c_4 - c_5 sont des unions de contact, et il existe seulement une union de fermeture entre c_2 - c_5 .

Dans la figure 5.16(b) nous présentons le graphique de liaison en les différents sous-assemblages du produit. Le sous-assemblage m_1 est constitué de l'union des composants c_2 , c_3 , c_4 et c_5 qui sont joints par l'union fermée entre c_2 et c_5 . Le sous-assemblage m_2 comprend le sous-assemblage m_1 et l'élément de fixation s_6 . L'assemblage final du produit est composé du sous-assemblage m_2 et du composant c_1 .

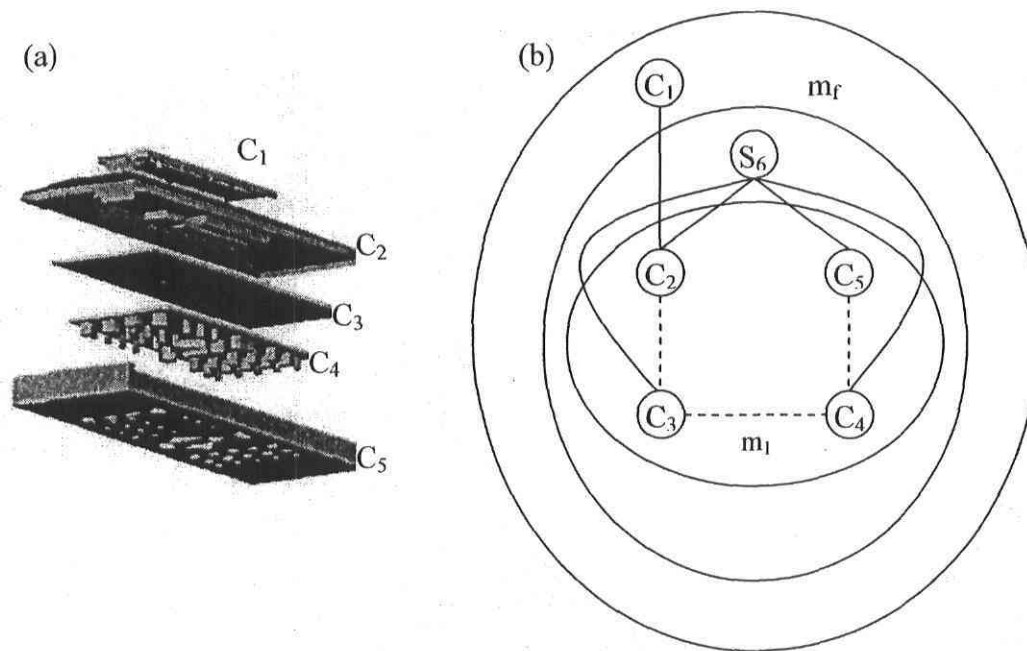


Figure 5.16 : Schéma et graphique d'une télécommande

Source : Torres, Puente et Aracil [51]

Pour trouver la séquence de désassemblage nous utilisons les quatre étapes définies précédemment. En premier, la séquence de désassemblage pour le composant c_3 est calculée, et donne la liste d'action présentée à la figure 5.17. Les actions sont trouvées comme suit :

- Étape 1 n'est pas utilisée parce que nous essayons de désassembler seulement un composant et pas un sous-ensemble. Le résultat après l'itération des étapes 2 et 3 de l'algorithme est :
 - $c_3 \in m_1 \Rightarrow$ L'action résultante peut être exprimée comme suit : enlever la fixation et séparer c_2 et c_5 et utiliser l'action : séparer c_2 et c_3 ou séparer c_5 et c_4 et alors c_4 et c_3 .
 - $m_1 \in m_2 \Rightarrow$ Enlever s_6 .
 - $m_2 \in m_f \Rightarrow$ Défaire l'union et séparer c_1 et m_2 .
- Finalement, l'étape 4 de l'algorithme établit les actions pour le désassemblage de c_3 ;
 - Défaire l'union et séparer c_1 et m_2 .

- Enlever s_6 .
- Défaire l'union et séparer c_2 et c_5 .
- Séparer c_2 et c_3 .
- Désassembler c_3 .

La procédure pour le désassemblage du composant 3 est donc réalisée.



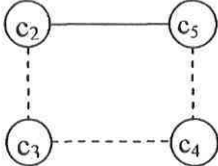
	<u>Représentation</u>	<u>Action</u>
$m_f = \{c_1, m_2\}$		(1) Défaire le joint et séparer c_1 et m_2
$m_2 = \{s_6, m_1\}$		(4) Enlever s_6
$m_1 = \{c_2, c_3, c_4, c_5\}$		(1) Défaire le joint et séparer c_2 et c_5 (2b) Prendre le composant cible

Figure 5.17 : *Actions pour chacun des sous-ensembles*

Source : Torres, Puente et Aracil [51]

Il est à noter que cette méthode ne prend pas en considération les temps de désassemblage et la présence de composants complexes et/ou liaisons complexes. Pour cela la méthode est utilisable facilement pour des produits simples mais elle est très rapidement arrêté par des produits plus complexes.

6 Stratégies avec les réseaux de Petri

Les méthodes des réseaux de Petri sont très répandues [24, 25, 34, 35, 36, 39, 47, 50 et 56]. Il en existe plusieurs versions et améliorations. Dans la présente section nous allons présenter deux versions du réseau de Petri. Tout d'abord le Réseau de Petri normal (6.2), et le Réseau de Petri Expert Stochastique Coloré et Augmenté (6.3). Chacune des deux méthodes comportent leurs propres hypothèses. Le niveau de complexité des deux méthodes est très différent tout comme le niveau d'efficacité. Elles permettent d'atteindre la séquence de désassemblage optimale mais de façon différente. Il existe plusieurs niveaux de complexité et d'efficacité entre les différentes utilisations (comme l'illustre le tableau 6.1 (A) et (B)).

Analyse des stratégies utilisant les réseaux de Petri

Tiwari, Sinha, Kumar, Rai et Mukhopadhyay [50] présentent un heuristique d'analyse de coûts pour l'assemblage d'une carte mère. Les différents composants du produit et leurs relations d'assemblage sont représentés par un diagramme du réseau de Petri. La transition du désassemblage du Réseau de Petri est intégrée avec les indices de coûts en développant une stratégie de désassemblage efficace. La méthodologie discutée simplifie les procédures de décision dans la planification du désassemblage. Les auteurs présentent des analyses permettant de choisir les stratégies de fin de vie des produits à utiliser. Ils se réfèrent aux travaux de Johnson et Wang [20, 21] pour ce qui est de l'importance économique associée avec les séquences de désassemblage. Les auteurs utilisent les cinq sous procédés de fin de vie de produits (réutilisation, désassemblage, démontage, recyclage, rejet) déterminés par Penev et Ron [37]. Ils présentent un système de décision pour le désassemblage avec un réseau de Petri.

Moore, Gungor et Gupta [36] présentent l'algorithme pour la génération de la procédure de désassemblage optimal du Réseau de Petri pour des produits avec relations de priorités ET/OU complexes. Ils travaillent essentiellement avec les matrices de désassemblage, la

méthode du réseau de Petri normal et les arbres de désassemblage. Chacun des outils sont modifiés pour permettre le désassemblage de produit avec relations complexes. Ils se basent sur les travaux de Lambert [28], Pnueli et Zussman [38], et leurs travaux [34,35]. Une évaluation des coûts est effectuée.

Kumar, Kumar, Shankar et Tiwari [24] présentent l'outil du Réseau de Petri Expert Stochastique Coloré et Augmenté. L'outil développé est plus complexe et complet que l'outil développé par Tiwari, Sinha, Kumar, Rai et Mukhopadhyay [50] et Moore, Gungor et Gupta [35]. Cette version du Réseau de Petri est utilisée autant dans une optique d'assemblage que de désassemblage tandis que le Réseau de Petri de Tiwari, Sinha, Kumar, Rai et Mukhopadhyay [50] est appliqué pour une stratégie de désassemblage.

Tang, Zhou et Caudill [47] présentent la planification du désassemblage et la méthode de gestion du désassemblage pour un système intégré flexible de démontage. Le poste de travail du réseau de Petri et le produit du réseau de Petri sont proposés pour ce modèle modulaire et hiérarchique dans l'ordre du chemin de désassemblage avec une valeur de fin de vie maximale. Ils détaillent la méthode du réseau de Petri pour le désassemblage, de la réception de produits, à la gestion des postes de travail. Ils analysent de façon très complète toutes les phases du désassemblage. Ils utilisent les matrices de désassemblage et les arbres de désassemblage.

Tableau 6.1: Comparaison des méthodes du réseau de Petri

Propriétés	Réseau de Petri normal (NPN)	Réseau de Petri Expert (EPN)	[24] Réseau de Petri stochastique généralisé (GSPN)	Réseau de Petri Stochastique de Haut Niveau (SHLPN)	Neuro Réseau de Petri Expert (NEPN)	Réseau de Petri Expert Stochastique Coloré et Augmenté (EECSPN)
<i>Élimination de transition</i>	Transitions est permise dès l'entrée des places de marquage	Transition permise est éliminée quand les évènements liés à la transition se produisent	Transitions est permise dès l'entrée des places de marquage	Temps d'élimination sont exponentiellement distribués	Transition permise est éliminée seulement quand les évènements liés à la transition se produisent	Permission d'élimination de transition seulement quand les règles liées avec sont satisfaites
<i>Contrôle de l'élimination</i>	Pas de contrôle	Contrôle par le moyen de contrôle de places	Pas de contrôle	Pas de contrôle	Pas de contrôle	Contrôle par le moyen de contrôle de places
<i>Temps comme paramètre</i>	Non-considéré	Considéré en terme de temps de transitions	Seulement applicable en temps de transitions	Temps exponentiel d'élimination	Non-considéré	Considéré en terme de temps de transitions
<i>But de Places</i>	Non-défini	Défini	Défini	Défini	Défini	Défini
<i>Probabilité d'occurrence</i>	Non-considéré	Non-considéré	Considéré	considéré	Non-considéré	Considéré
<i>Identification de couleur</i>	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Considéré
<i>Étiquettes basées sur les règles</i>	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Considéré
<i>Empêchement de réfraction</i>	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Considéré
<i>Considération des évènements négatifs</i>	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Non-considéré	Considéré
<i>Apprentissage</i>	Non-considéré	Considéré	Non-considéré	Non-considéré	Considéré	Considéré

6.1 Méthodes des réseaux de Petri

Tout d'abord nous allons voir ce qu'est un réseau de Petri (PN) et sa théorie de base. Il est présenté dans plusieurs articles dont celui de Moore, Gungor, et Gupta [35], Tiwari, Sinha, Kumar, Rai, et Mukhopadhyay [50], et Rai, Rai, Tiwari et Allada [39]. C'est une méthode très efficace et qui n'a que très peu de restrictions. Pour son utilisation nous avons besoin des matrices de désassemblage (4.2.1) et le résultat est donné sous forme d'arbre de désassemblage (4.2.2). Le réseau de Petri normal (NPN) est un quintuple :

$$\text{NPN} = (\mathbf{P}, \mathbf{T}, \mathbf{A}, \mathbf{W}, \mathbf{M}_0)$$

Où :

$$\mathbf{P} \cap \mathbf{T} = \emptyset \text{ et } \mathbf{P} \cup \mathbf{T} \neq \emptyset$$

\mathbf{P} : $\{p_1, p_2, p_3, \dots, p_n\}$ série avec une infinité de position

\mathbf{T} : $\{t_1, t_2, t_3, \dots, t_n\}$ série avec une infinité de transition

\mathbf{A} : série d'arcs (A_e : arcs déclencheurs, A_i : arcs d'inhibition) ($A \subseteq (\mathbf{P}\mathbf{T}) \cup (\mathbf{T}\mathbf{P})$)

\mathbf{W} : $\mathbf{A} \rightarrow \{1, 2, 3, \dots\}$ série de fonctions avec les poids des arcs

\mathbf{M} : $\mathbf{P} \rightarrow \{0, 1, 2, 3, \dots\}$ série de marquage

Un marquage M_j dénote le état courant du PN. Nous définissons la série de fonctions I des places d'entrées et O de places de sorties pour une transition t , comme $I(t) = \{p \mid (p,t) \in A\}$ et $O(t) = \{p \mid (t,p) \in A\}$, respectivement. Une transition t est permise pour un marquage M_i si $M_i(p) \geq w(p,t) \forall p \in I(t)$, où $w(p,t)$ est le poids d'un arc de p à t . Quand une transition est éliminée, le nouveau marquage est défini comme $M_{i+1} = \{M_i - w(p,t) \forall p \in I(t) ; M_i + w(t,p) \forall p \in O(t) ; M_i, \text{ autrement}\}$.

Les hypothèses du réseau de Petri normal sont les suivantes :

- En cas de ET/OU complexe, un composant ne peut pas être de priorité OU dans plus qu'une direction ;

- Nous ne pouvons pas considérer de direction dans le cas de XOU (Relation OU complexe) ;
- Nous supposons qu'il y a un nombre infini de composants ;
- Les composants peuvent être enlevés sans désassemblage destructif ;
- Les composants sont enlevés individuellement ;
- Les composants sont enlevés dans une seule direction.
- Ne considère pas le temps, qui est le cœur des paramètres et de chaque opération ;
- Ne compte pas les cases négatives ;
- Ne donne pas les relations entre les composantes à assembler et désassembler ;
- Ne fais pas de différence entre les variations d'outils.

Dans les différentes utilisations du réseau de Petri nous tentons le plus possible d'éliminer ces hypothèses très restrictives. Les deux méthodes suivantes permettent d'éliminer certaines hypothèses.

6.2 Réseau de Petri avec relations de priorité ET/OU complexes

L'approche du Réseau de Petri avec relations de priorité ET/OU complexes est présenté dans l'article de Moore, Gungor, Gupta [34, 35, 36]. C'est une version améliorée du réseau de Petri normal. Il permet le désassemblage de composants avec liaisons ET/OU complexes.

Hypothèses

1. La rotation de la table de désassemblage ou du produit durant le désassemblage n'est pas permis.
2. L'enlèvement du composant est effectué dans une seule direction, $d \in D$.
3. Les composants sont rigides et inflexibles; c'est-à-dire des composants ne peuvent être déformés pour le désassemblage.
4. Quand un composant est enlevé, les composants restants maintiennent leur position originale.
5. Les composants sont enlevés individuellement et tous les autres.
6. Chaque composant peut être enlevé par la satisfaction des relations de priorités.
7. Il y a un nombre fini de composants dans le produit.

Nomenclature : Voir Annexe C.

Plan d'utilisation de la méthode

Cette méthodologie se résume en 3 étapes (figure 6.1) :

1. Analyse du produit pour générer une matrice de priorité de désassemblage (DPM) représentant une base physique des contraintes de désassemblage;
2. Génération du désassemblage avec réseau de Petri de la DPM ;
3. Génération du plan de procédure de désassemblage (DPP) optimal du réseau de Petri.

Tout au long de la présentation de cette méthode nous allons utiliser le produit présenté à la figure 6.2. Cet exemple de produit comprend sept composants (cinq composants et deux éléments de joint). Ici nous limitons les mouvements à deux dimensions mais le réseau de Petri peut travailler avec des modèles en trois dimensions. Les directions de mouvement possibles pour cet exemple sont $d \in D$, où $D = \{x,-x,y,-y\}$.

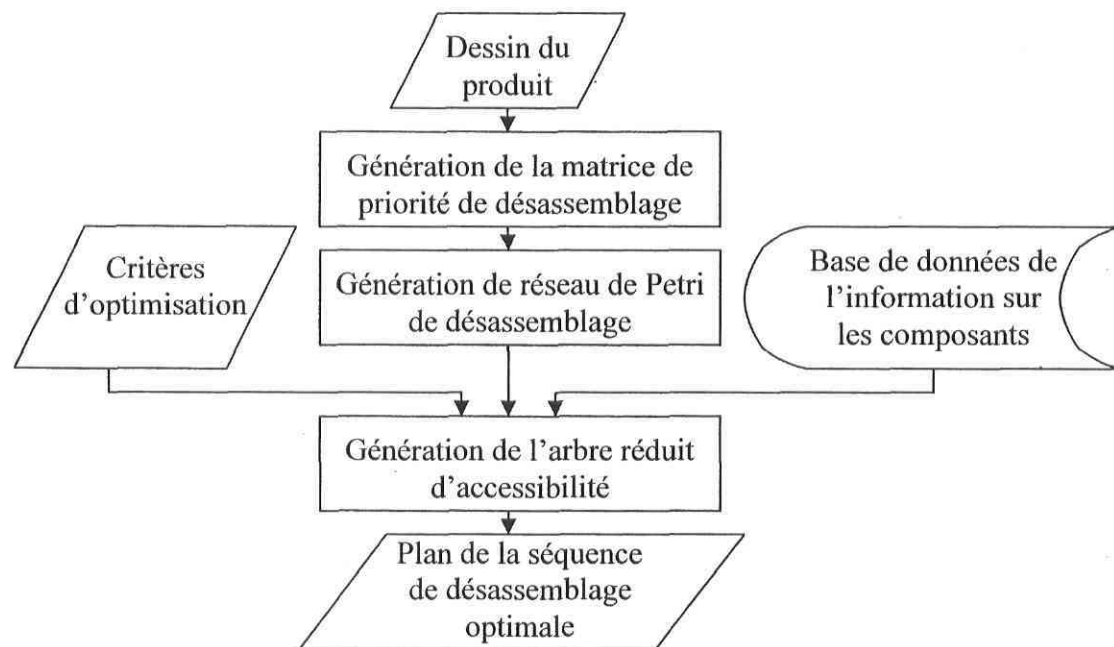


Figure 6.1 : Diagramme pour la génération de la séquence de désassemblage

Source : Moore, Gungor, Gupta [36]

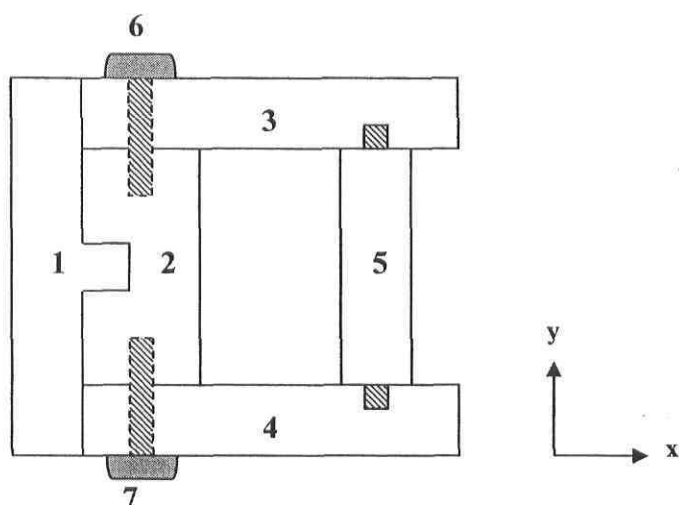


Figure 6.2 : Exemple de produit

Source : Moore, Gungor, Gupta [35]

Étape 1 : Génération de la matrice de priorité de désassemblage (DPM)

La matrice de priorité de désassemblage représente la base géométrique des relations de priorités entre les composants d'un produit. Ces relations définissent les contraintes de faisabilité pour la génération de la procédure de désassemblage optimum. Moore, Gungor, Gupta [14] utilisent leur propre algorithme pour la génération des matrices de désassemblage. Les relations physiques de priorités introduisent les déplacements transitoires de la manière suivante : « c_1 précède c_2 dans la direction x » ce qui veut dire que c_1 doit être enlevé dans la direction x afin d'enlever c_2 .

Nous identifions la série d'éléments d'obstruction comme suit. Pour chaque composant i , nous déterminons les chemins pour chaque direction de mouvement possible, $d \in D$. Les composants d'obstruction pour le composant i sont les composants qui sont dans la ligne de mouvement du composant i dans la direction d ; cette information géométrique est la seule nécessaire pour la génération de la matrice de priorité de désassemblage. Par exemple, dans la figure 6.2, c_3 et c_4 bloquent c_5 dans la direction x , c_2 , c_3 et c_4 bloquent c_5 dans le direction $-x$; c_3 bloque c_5 dans le direction y ; et c_4 bloque le composant c_5 dans la direction de $-y$. Alors, $e_{5,x} = \{3,4\}$, $e_{5,-x} = \{2,3,4\}$, $e_{5,y} = \{3\}$ et $e_{5,-y} = \{4\}$. En continuant de cette manière, nous générons E pour notre exemple. De E , nous pouvons générer \hat{E} avec l'algorithme donné dans la figure 6.3. Donc, nous initialisons les variables qui sont les cardinalités de $e_{5,d}$: $|e_{5,x}| = 2$; $|e_{5,-x}| = 3$; $|e_{5,y}| = 1$ et $|e_{5,-y}| = 1$. Le nombre minimum d'éléments bloquant le composant 5 sont $e_{5,y}$ et $e_{5,-y}$. Ainsi, nous écrivons la ligne 5 de la matrice \hat{E} du nombre minimum d'éléments d'obstruction, comme suit : $\hat{e}_{5,x} = \{\infty\}$, $\hat{e}_{5,-x} = \{\infty\}$, $\hat{e}_{5,y} = \{3\}$ et $\hat{e}_{5,-y} = \{4\}$. Nous répétons la procédure pour tout les lignes de E pour trouver \hat{E} . Nous pouvons maintenant construire la matrice de priorité de désassemblage, B , avec l'étape 4 de l'algorithme de la figure 6.3. Nous considérons la ligne 5 de \hat{E} pour notre composant; v , le nombre de $\hat{e}_{id} \neq \{\emptyset, \infty\}$, est deux. Donc $v > 1$ ce qui nous permet d'obtenir le vecteur de la 5^e colonne de B : $[0,0,y,-y,0,0,0]^T$. En répétant l'Étape 4 pour toutes les lignes de \hat{E} nous obtenons la matrice de priorité de désassemblage B :

$$B = \begin{matrix} k & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & \left[\begin{array}{ccccccc} 0 & -x & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & y & 0 & 0 \\ 0 & 0 & 0 & 0 & -y & 0 & 0 \\ 0 & x & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{array} \right] \\ 2 & & & & & & & \\ 3 & & & & & & & \\ 4 & & & & & & & \\ 5 & & & & & & & \\ 6 & & & & & & & \\ 7 & & & & & & & \end{matrix}$$

Étape 1 : Initialisation des variables

E et Ê sont $k \times |D|$, où

$$e_{id} = \hat{e}_{id} = \phi, i = 1 \text{ à } k, d \in D$$

B = 0

Étape 2 : Générer E

Pour chaque composant i :

Si i est bloqué par j dans la direction d ($i \neq j$)

$$e_{id} = e_{id} \cup j$$

Si i ne peut est enlevé dans la direction d

Dû aux fixations ou type de composant

$$e_{id} = \infty$$

Étape 3 : Générer Ê

Pour i^e ligne de E

$$g_i = \min |e_{id}|, d \in D$$

Si $\exists \{d | e_{id} = \phi\}$, quand $|e_{id}| = 0$

$$G_i = \{e_{id}\}, |e_{id}| = g_i$$

Série de priorités minimums

Si $g_i > 0$

$$G_i^A = e_{id}, e_{id} \in G_i$$

Série de ET contenant la série de priorité minimale

Pour chaque paire d'éléments dans G_i

Répéter cette boucle pour toutes paires dans G_i

$$\text{Mettre } G_c = e_{id} \cap e_{if}, d \neq f$$

$$\text{Si } G_i^A - G_c \neq \emptyset$$

Si vrai, il existe un conflit

$$\text{Mettre } g_c = e_{id} \text{ ou } e_{if}$$

Une sélection aléatoire pour l'éliminer de G_i

$$\text{Mettre } G_i = G_i - g_c$$

Une élimination de paire de composants en conflit

$$\hat{e}_{id} = \begin{cases} e_{id}, e_{id} \in G_i \\ \infty, \text{ autrement} \end{cases}$$

Figure 6.3(a) : Algorithme pour générer la matrice de priorité de désassemblage

Source : Moore, Gungor, Gupta [36]

Étape 4 : Génération de BPour la ligne i de \hat{E} v étant le $\min(g_i, |G_i|)$ v = nombre de $\hat{e}_{id} \neq \{\emptyset, \infty\}$ Si $v = 0$ $b_{ji} = 0, \forall j, j = 1 \text{ à } k$ $\hat{e}_{id} = \{\emptyset, \infty\}$ Si $v = 1$ $b_{ji} = 1, \forall j, j = \{\hat{e}_{id} | \hat{e}_{id} \neq \infty\}$ S'il y a seulement un $\hat{e}_{id} \neq \{\emptyset, \infty\}$ et aucun = \emptyset Si $v > 1$

$$b_{ji} = \begin{cases} 1, & j \in G_i^A \\ d, & j \in \hat{e}_{id} - G_i^A \\ 0, & \text{autrement} \end{cases} \quad \text{S'il y a plus que un } \hat{e}_{id} \neq \{\emptyset, \infty\} \text{ et aucun} = \emptyset$$
Figure 6.3(b) : Algorithme pour générer la matrice de priorité de désassemblage

Source : Moore, Gungor, Gupta [36]

Étape 2 : Génération du réseau de Petri de désassemblage

Nous allons maintenant développer l'algorithme pour générer un réseau de Petri de désassemblage de B. Cet algorithme est présenté dans la figure 6.4. Le réseau de Petri pour le désassemblage pour notre exemple est défini comme suit :

$$P = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_b, p_f, p_{o_2}, p_{o_5}\},$$

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_b, t_f\},$$

$$M_0 = \{0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0\},$$

$$A = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Étape 1 : Initialisation des variables

$$P = \{p_b, p_i, p_j\}, T = \{t_b, t_j, t_f\}, \text{ où } i, j = 1 \text{ à } k$$

$$A = (k + 2) \times (k + 2) \text{ matrice, où } a_{ii} = -1 \text{ pour } i = 1 \text{ à } k + 2, a_{k+1, k+2} = 1$$

$$AG_j = O_j = OG_j, d = \{\phi\}, j = 1 \text{ à } k, d \in \{D\}, nz_i = 1, i = 1 \text{ à } k$$

Étape 2 : Compléter T, P, et A**Étape 2.1 : Scruter B_j pour les groupes de priorités ET et OU**Si $b_{ij} = 1$

$$AG_j = \{AG_j \cup i\} \quad \text{Ajouter } i \text{ à la série de priorités ET à } j$$

$$nz_i = 0 \quad \text{Série } nz_i \text{ à } 0 \text{ (} i \text{ a un antécédent)}$$

Si $b_{ij} = d$

$$OG_{j,d} = \{OG_{j,d} \cup i\} \quad \text{Ajouter } i \text{ au groupe de priorité OU pour } j \text{ dans la direction } d$$

$$O_j = \{O_j \cup i\} \quad \text{Garder la trace des directions du groupe de priorités OU à } j$$

$$P = \{P \cup po_j\} \quad \text{Ajouter les places au manipulation de priorités complexes OU}$$

$$nz_i = 0 \quad \text{Mettre } nz_i \text{ à } 0 \text{ (} i \text{ a un antécédent)}$$

Étape 2.2 : Générer des arcs dans A pour les places et les transitions associées avec B_j **Étape 2.2.1: Examiner AG_j**

$$\text{Si } |AG_j| = 0 \quad \text{Vérifier pour l'absence de priorité ET}$$

$$a(p_j, t_b) = 1 \quad \text{Ajouter l'arc permettant de faire cette tâche}$$

$$\text{Si } |AG_j| > 0 \quad \text{Vérifier la présence de priorités ET}$$

$$a(p_j, t_j) = -|AG_j| \quad \text{Ajouter l'arc pour compléter les priorités ET}$$

$$a(p_j, t_i) =, \text{ pour } i \in AG_j \quad \text{Ajouter l'arc pour les priorités ET}$$

Figure 6.4(a) : Algorithme pour générer réseau de Petri de désassemblage

Source : Moore, Gungor, Gupta [36]

Étape 2.2.2: Examiner $O_{j,d}$ et O_j

Si $ O_j > 0$	Vérifier pour les arcs les conditions OU
$a(po_j, t_j) = -1$	Ajouter l'arc pour les priorités OU à j
$a(po_j, t_f) - (O_j - 1)$	Ajouter l'arc à t_f pour le reste des OU à j
Si $ O_{j,d} = 1$, pour $d \in O_j$	Vérifier pour la réduction
$a(po_j, t_i) = 1$, pour $i \in O_{j,d}$	Ajouter l'arc pour la condition OU simple à j
Si $ O_{j,d} > 1$, pour $d \in O_j$	Vérifier les arcs dans les conditions complexes ET/OU
$T = \{T \cup ta_{j,d}\}$	Ajouter une transition qui représente les ET avec $OG_{j,d}$
$P = \{P \cup pa_{j,d} \cup po_j\}$	Ajouter les places de manipulation dans les priorités complexes ET/OU
$a(po_j, ta_{j,d}) = 1$	Ajouter l'arc pour le groupe complet OU
$a(pa_{j,d}, ta_{j,d}) = - O_{j,d} $	Ajouter l'arc qui collecte les ET dedans le groupe OU
$a(pa_{j,d}, t_i) = 1$ pour $i \in O_{j,d}$	Ajouter les arcs pour les ET dedans le groupe OU

Étape 3 : Finaliser A

Étape 3.1 : Scruter nz_i aux arcs générés pour les composants sans antécédent

$a(p_f, t_i) = nz_i$ Ajouter l'arc pour les composants sans antécédent

Étapes 3.2 : Sommer nz_i aux arcs générés pour la place finale

Si $\sum_{i=1}^k nz_i > 0$, $a(p_f, t_f) = -\sum_{i=1}^k nz_i$ Ajouter l'arc d'accumulation de composants sans antécédent

Si $\sum_{i=1}^k nz_i = 0$, $a(p_f, t_b) = 1$ Case de manipulation où ils sont sans ligne nul en B

Figure 6.4(b) : Algorithme pour générer réseau de Petri de désassemblage

Source : Moore, Gungor, Gupta [36]

La représentation graphique de A avec le marquage définit dans M_0 est illustrée dans la figure 6.5.

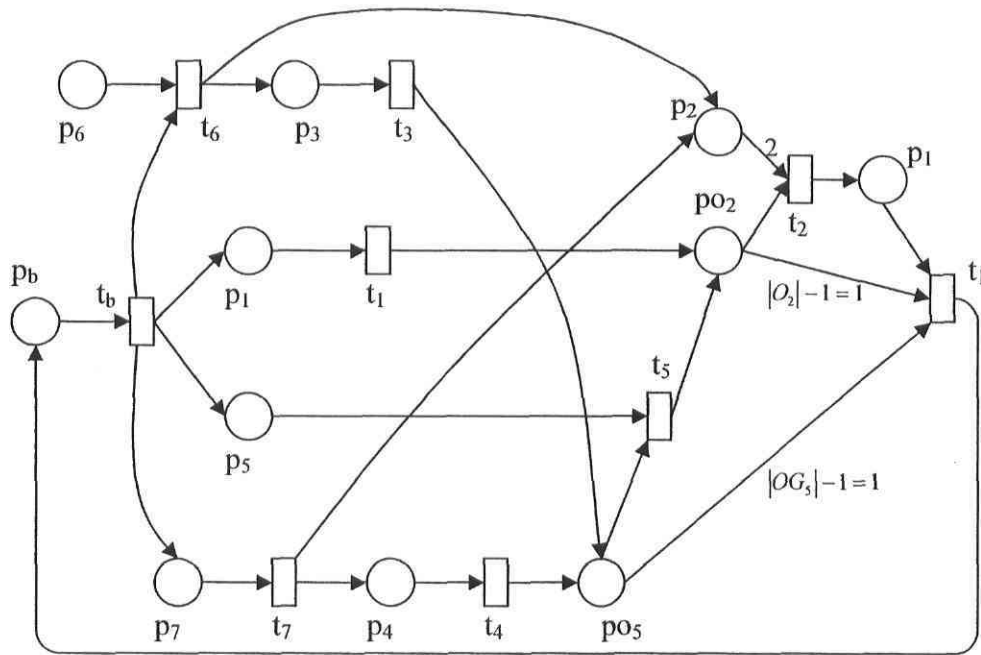


Figure 6.5 : *Désassemblage avec réseau de Petri (DPN) du produit*

Source : Moore, Gungor, Gupta [36]

La résolution de l'algorithme demande quelques lemmes et théorèmes. Ils sont les suivants :

Lemme 1 : Chaque DPM contient au moins une colonne de zéro, c'est-à-dire

$$\exists B_{i-} = \{b_{ij} | b_{ij} = 0\}.$$

Lemme 2 : Chaque DPM contient au moins une ligne qui ne contient pas de 1 ; c'est-à-dire

$$\exists B_{-j} = \{b_{ij} | b_{ij} = 0 \text{ ou } d\}.$$

Théorème 1 : S_{NP} , S_{NA} , S_A et S_{AO} sont libres, en phase, et réversibles.

Théorème 2 : S_M est libre, en phase, et réversible.

Étape 3 : Génération du plan de séquence de désassemblage optimale

L'arbre d'accessibilité est une énumération de tous les marquages accessibles dans un réseau de Petri, c'est-à-dire de $M \in R(M_0)$. En donnant un réseau de Petri avec marquage M_0 , nous pouvons obtenir plusieurs nouveaux marquages comme transitions possibles, $|E(M_0)|$. Pour chacun de ces marquages, nous pouvons obtenir de nouveaux marquages,

et ainsi on produit un arbre de marquages où M_0 est la racine, les nœuds représentant les marquages générés de M_0 et ses successeurs, et les arcs représentant le lien d'un parent à un nœud successeur. Nous pouvons utiliser l'arbre d'accessibilité pour générer tous les changements de transitions possibles, ou séquences de désassemblage. En ayant les séquences faisables de désassemblage nous pouvons utiliser les fonctions de coûts pour déterminer la séquence optimale de désassemblage. Il est à noter que l'arbre d'accessibilité est impraticable pour les produits avec beaucoup de composants.

Fonctions de coûts

Les considérations de coûts commun dans le désassemblage incluent les changements d'outils et de directions de mouvement [12,13,35]. Le changement d'outil demande un coût et un temps d'installation additionnel, pendant le changement de direction demande un temps additionnel pour réaliser la procédure. Un autre facteur de coût doit être mentionné, les caractéristiques spécifiques d'un ou plusieurs composants du produit qui doivent être désassemblés. L'enlèvement de chaque composant doit minimiser la possibilité de dommage d'un composant fragile ou la contamination de la zone de travail d'un composant hasardeux, ou minimiser le temps avant l'extraction d'un composant ayant une grande valeur. Les facteurs additionnels doivent être identifiés dépendamment des besoins individuels du système de désassemblage et des produits. Nous utilisons trois facteurs déterminants, lesquels sont interprétés comme des pénalités de temps de base : pénalité de changement d'outil (α), pénalité de changement de direction de désassemblage (β), et une pénalité pour le délais d'enlèvement d'un composant hasardeux (δ).

C_n est le coût associé avec le marquage M_n (en unité de temps), nous définissons :

$$C_{n+1} = C_n + \Delta t_j + a\alpha + b\beta + (|H - H_r|)\delta,$$

$$a = \begin{cases} 0 & \text{outil pour } (t_j | M_{n-1} [t_j] M_n) = \text{outil pour } t_j \in E[M_n] \\ 1 & \text{autrement;} \end{cases}$$

$$b = \begin{cases} 0 & d \text{ pour } (t_j | M_{n-1} [t_j] M_n) = d \text{ pour } t_j \in E[M_n] \\ 1 & \text{autrement.} \end{cases}$$

Génération de l'arbre réduit d'accessibilité (RRT)

L'algorithme RRT définit ici par les entrées suivantes : A, M₀, H, la série de temps de changement de transition, et l'information sur les changements d'outils et de directions. L'algorithme RRT emploie une recherche multi-hypothèses construite dynamiquement, les « v coût le plus bas espérer » arcs de l'arbre. Chacun de ces arcs représentent une séquence d'élimination de transition, $\sigma_i, i = 1 - v$, qui guide un désassemblage complet avec le coût de chaque séquence de désassemblage. Dans la notion d' « arc v avec le plus bas coût espéré », v décrit le nombre du DPP presque optimum, que nous pouvons trouver. En d'autres mots, v définit comment le coût le plus bas est sélectionné dans le niveau de recherche. Une fois que les v DPP sont générés, nous sélectionnons la meilleur de toute comme séquence de désassemblage préférée. L'algorithme pour construire le RRT est donné à la figure 6.6.

Étape 1 : Initialisation de l'arbre

Générer $L_e, e = 1 \text{ à } z ; q = 0 ; zc = 0$

Laisser M_0 être un nœud ; c'est le composant .

$M' = M_0 ; IM_q = M_0 ;$

$q = q + q ; zc = zc + 1 ;$

Étape 2 : Construire l'arbre du nœud initial.

Étape 2.1 : SI $q > |T|$, aller à l'étape 3,

Autrement CONTINUER.

Figure 6.6(a) : Algorithme de l'arbre réduit d'accessibilité

Source : Moore, Gungor, Gupta [36]

<p>Étape 2.1.1 : Si $q = Lzc$</p> <p>Générer ET_q du marquage de IM_q-1</p> <p>Pour $\forall f \in ET_q$</p> <p>Transition éliminée f;</p> <p>Générer un nouveau nœud d'ordre, M_{new} ;</p> <p>Créer une flèche d'entrée du nœud qui permet f et étiquette de f;</p> <p>Calculer la coût de M_{new} ;</p> $IM_q = IM_q \cup M_{new}$ <p>Générer IM ;</p> <p>$M'_e \in IM, e = 1 \text{ à } v$;</p> <p>$IM_q = M'$;</p> <p>$nc + nc + 1 ; q = q + 1$;</p> <p>Aller à l'étape 2.1.</p> <p>Étape 2.1.2 : Si $q \neq Lzc$</p> <p>Générer ET_q de marquage de IM_{q-1}</p> <p>Pour $\forall f \in ET_q$</p> <p>Transition éliminée f;</p> <p>Générer un nouveau nœud d'ordre, M_{new} ;</p> <p>Créer une flèche d'entrée du nœud qui permet f et étiquette de f ;</p> <p>Calculer la coût de M_{new} ;</p> $IM_q = IM_q \cup M_{new}$ <p>$q = q + 1$;</p> <p>Aller à l'étape 2.1.</p> <p>Étape 3 : Imprimer l'arbre d'accessibilité.</p> <p>Print RRT. ARRÊT.</p>	<p>Niveau de recherche</p> <p>Trouver la transition permise</p> <p>Pour toutes les transitions permises</p> <p>Transition éliminée</p> <p>Créer une flèche entre le marquage parent et le nouveau marquage</p> <p>Nouveau nœud marqué comme IM</p> <p>Arranger IM_q</p> <p>Trouver les meilleurs nœuds v</p> <p>Les meilleurs marquages sont les nouvelles marques intermédiaires</p> <p>La profondeur n'est pas un niveau de recherche</p> <p>Trouver les transitions permises</p> <p>Pour toutes les transitions permises</p> <p>Transition éliminée</p> <p>Créer une flèche entre le marquage parent et le nouveau marquage</p> <p>Calcul du coût</p> <p>Nouveau nœud est inséré dans IM_q</p> <p>Arbre est complètement construit</p>
--	--

Figure 6.6(b) : Algorithme de l'arbre réduit d'accessibilité

Source : Moore, Gungor, Gupta [36]

Résultats pour notre exemple

Nous illustrons l'approche en utilisant le produit présenté à la figure 6.2. Le désassemblage avec réseau de Petri pour le produit est développé dans la figure 6.5. Les temps de désassemblage, les types d'outils, et les directions sont donnés dans le tableau 6.2. Nous déterminons $\alpha = \beta$, $\delta > \alpha$, et $H = \{5\}$, (c'est-à-dire : le composant 5 contient du matériel hasardeux). Note, toutes les directions possibles de désassemblage ne sont pas présentées. Nous présentons seulement les directions de désassemblage préférable. Les directions sont choisies préalablement par le désassembleur pour différentes raisons. Par exemple, en lien avec le DPM et DPN, le composant 2 peut être enlevé dans la direction $-x$ (dû au limitation de l'outil), pour cela nous mettons $-x$ comme opération de désassemblage préférable pour le composant 2 dans le tableau 6.2. En utilisant l'algorithme RRT, avec $w = 1$ et $v = 2$, nous obtenons la séquence optimale de transition de $[t_b t_6 t_3 t_5 t_7 t_4 t_1 t_2 t_f]$; le RRT contenant 35 nœuds. Donc t_b et t_f , n'incluant pas de coût, elle peuvent être enlevées du plan de procédures de désassemblage (DPP). Le DPP résultant est $[6 3 5 7 4 1 2]$.

Tableau 6.2 : Temps de désassemblage, outils, directions pour l'exemple [36]

t_j	Δt_j	Type d'outil	Direction de désassemblage
t_b	0.0	Aucun	Aucun
t_f	0.0	Aucun	Aucun
t_1	10.0	2	-X
t	10.0	3	-X
t	10.0	4	Y
t	10.0	4	-Y
t	10.0	3	-Y
t_6	10.0	1	Y
t_7	10.0	1	-Y

Finalement cette méthode permet de générer la séquence de désassemblage optimale. L'algorithme de l'arbre réduit d'accessibilité explore dynamiquement le produit en trouvant les arcs les moins dispendieux. La fonction de coût incorpore le changement d'outil, le changement dans les directions de mouvement, et les caractéristiques individuelles des composants donnant priorité à l'enlèvement de certains composants plus critiques. L'approche peut être utilisée pour des produits ayant des relations ET, OU, et ET/OU complexes.

6.3 Le réseau de Petri expert stochastique coloré et augmenté

Kumar, Kumar, Shankar et Tiwari [24] présentent de façon détaillée cette méthode. Ils se concentrent sur la présentation du Réseau de Petri Expert Stochastique Coloré et Augmenté (EECSPN) et de son application dans l'assemblage/désassemblage, la planification de l'assemblage/désassemblage, la modélisation et représentation, les composants et séquences d'opérations et optimisation d'un plan détaillé. La section suivante comprend les éléments suivants : le travail avec le modèle EECSPN et ses composants ; la présentation de la planification des tâches d'assemblage/désassemblage ; l'analyse de l'assemblage versus le désassemblage utilisé avec le EECSPN ; et la présentation de la planification de séquences de tâches d'assemblage versus de désassemblage. Les matrices de désassemblage (4.2.1) et les arbres de désassemblage (4.2.2) sont utilisés.

6.3.1 Description de la méthode EECSPN

Nomenclature

- R : série de probabilité
- L : série d'arcs étiquetés (L_g : arcs étiquetés verts, L_r : arcs étiquetés rouge)
- U : série de constantes et variables (N : série de constantes, V : série de variables)
- C : série de couleurs associées avec les étiquettes (g : vert, r : rouge)
- K : série d'apprentissage
- P_N : série de places normales,
- P_I : série de places d'inhibition
- P_C : série de places de contrôle
- P_T : série de places de transition
- I : série d'arcs d'entrées (I_e : arcs d'entrées déclencheurs, I_i : arcs d'entrées d'inhibition)
- O : série d'arcs de sorties (O_e : arcs de sorties déclencheurs, O_i : arcs de sorties d'inhibition)
- a_{ij} : probabilité pour le changement dans le marquage M_i à M_j

Définition du EECSPN

EECSPN est un outil graphique et mathématique avec la capacité de performer dans une analyse en profondeur et permet de travailler avec des procédures de système ayant des problèmes plus complexes. Le Réseau de Petri Expert Stochastique Coloré et Augmenté est un 9-tuple :

$$\text{EECSPN} = (\mathbf{P}, \mathbf{T}, \mathbf{A}, \mathbf{M}, \mathbf{L}, \mathbf{U}, \mathbf{C}, \mathbf{K})$$

Où

P est une série de places qui satisfont la relation :

$$P = P_N \cup P_I \cup P_C \cup P_T$$

$$P_N \cap P_I = \phi, P_C \cap P_T = \phi, P_I \cap P_T = \phi$$

$$P_N \cap P_C = \phi, P_C \cap P_T = \phi, P_I \cap P_C = \phi$$

$$P_N \cup P_I \neq \phi, P_C \cup P_T \neq \phi, P_I \cup P_T \neq \phi, \text{ et}$$

$$P_N \cup P_C \neq \phi, P_N \cup P_T \neq \phi, P_I \cup P_C \neq \phi$$

Place normale : représente les composants faisables impliqués dans les procédures d'assemblage/désassemblage : $P_N = \{p_1, p_2, p_3, \dots\}$

Place d'inhibition : représente les composants faisables impliqués dans les procédures d'assemblage/désassemblage qui ne sont pas un succès, c'est-à-dire qui ne sont pas prévus dans les procédures mais peuvent être accidentellement produits : $P_I = \{p_{i1}, p_{i2}, p_{i3}, \dots\}$

Place de contrôle : places associées avec le flot de contrôle des procédures d'assemblage/désassemblage. Les places de contrôle sont appliquées quand il y a plus d'un chemin faisable. Elles sélectionnent le chemin optimal au travers des chemins faisables : $P_C = \{p_{c1}, p_{c2}, p_{c3}, \dots\}$

Place de transition : places associées avec des transitions qui ont au moins un arc d'entrée. Leur fonction est d'éviter les réfractions (un phénomène de boucle infinie entre une transition et une place). Les arcs associés avec les places de transition sont toujours dirigés de la transition de place à sa transition respective :

$$P_T = \{p_{T1}, p_{T2}, p_{T3}, \dots\}$$

T est une série de transitions telle que : $T = \{t_1, t_2, t_3, \dots\}$

A est une série d'arcs, telle que :

$$A = I \cup O$$

$I \cap O = \phi, I \cup O \neq \phi$, où une série d'arcs d'entrées

$$I = I_e \cup I_i$$

I_e est une série d'arcs d'entrées déclencheurs, et **I_i** est une série d'arcs d'entrées de maîtrise : $I_e \cup I_i \neq \phi, I_e \cap I_i = \phi$.

O est une série d'arcs de sorties : $O = O_e \cup O_i$.

O_e est une série d'arcs d'entrées déclencheurs, **O_i** est une série d'arcs d'entrées de maîtrise : $O_e \cup O_i \neq \phi, O_e \cap O_i = \phi$.

Arcs déclencheurs : Arcs de croisement de places et de transitions, où la faisabilité des opérations prend place :

$$A_e = I_e \cup O_e$$

$$I_e \cup O_e \neq \phi, I_e \cap O_e = \phi,$$

où A_e est une série d'arcs déclencheurs.

Arcs de maîtrise : Arcs de croisement de places et de transitions, où les opérations prennent place ne sont pas fructueuses. En d'autres mots, ce sont des arcs reliant des transitions aux endroits de maîtrise :

$$A_i = I_i \cup O_i \quad \text{où} \quad A_i \text{ est une série d'arcs de maîtrise.}$$

$$I_i \cup O_i \neq \phi, I_i \cap O_i = \phi,$$

Série d'arc A_g est une série associée avec des places contenues une pièce verte et A_r est une série d'arcs associés qui sont considérés comme une pièce rouge.

M est une série de marquages :

$$M : P \rightarrow \{0,1,2,\dots\}$$

$$M = \{N_T(p_1), N_T(p_2), \dots\}$$

$$\text{Également, } M = M_g \cup M_r$$

Une série d'arcs M_g est le marquage de places contenant des pièces marquées en verts et M_r est une série de places contenant des pièces marquées en rouges. La marque M pour une série de m places est représenté par une matrice $m \times 1$.

Équation d'état :

M_k est marqué après k renvois, M_{k-1} est marqué après $(k-1)$ renvois et A est la matrice d'incidence démontrant le changement de marquage, alors :

$$M_k = M_{k-1} + A^T U_k, \forall k = 1, 2, 3, \dots,$$

Où : U_k est une matrice de $n \times 1$ colonne, et comprend un vecteur de contrôle. Cela consiste en $n - 1$ et un 1 la t^e position. 1 indique que la transition 't' éliminée à la k^e élimination.

Condition d'atteinte :

Si M_k est une marque d'atteinte de M_0 , au travers d'une élimination de séquence $\{u_1, u_2, u_3, \dots\}$, alors la condition pour l'atteinte est :

$$M_k = M_0 + A^T \sum_{k=1}^k u_k$$

$$\Rightarrow M_k - M_0 = A^T \sum_{k=1}^k u_k$$

$$\Delta M = M_k - M_0 \equiv \text{changement en marquage}$$

$$\Delta M = A^T \sum_{k=1}^k u_k$$

R est la série de probabilité : $R = \{R_0, R_1, R_2, R_3, \dots, R_N\}$

Et $R_i = \{a_{i0}, a_{i1}, a_{i2}, \dots\}$, $i = 1, 2, 3, \dots$, où, $\sum a_{ij=1}$ et $0 \leq a_{ij} \leq 1$.

Alors a_{ij} est la probabilité pour le changement dans le marquage M_i à M_j . Probabilité de série R donne la probabilité de chaque configuration à occurrence dans l'opération d'assemblage/désassemblage.

L est une série d'arcs étiquetés utilisés pour les arcs de désignation.

L'étiquetage peut aussi être utilisé pour la représentation entre les relations de composants de tous les niveaux d'opération.

Aussi, $L = L_g \cup L_r$, où, L_g est un niveau de l'arc associé avec les places contenant des étiquettes vertes et L_r est un niveau de l'arc associé avec les places contenant les étiquettes rouges.

Aussi, $L_g \cup L_r \neq \emptyset$

$$L_g \cap L_r = \emptyset$$

U est une série de constantes et variables : $U = N \cup V$,

Où : N est une série de constantes

V est un série de variables

N est une série K-tuple

$$N = \langle n_1, n_2, n_3, \dots \rangle$$

V est une série de k-tuple et

$$V = \langle v_1, v_2, v_3, \dots \rangle$$

C est une série de couleurs associées avec les étiquettes : $C = \{g, r\}$

Où 'g' représente la couleur verte et 'r' est la couleur rouge.

Une transition $t \in T$ permet le marquage $M = M_g \cup M_r$ si :

$$\text{Couleur } (L_g(p, t)) \in M_g(p), \forall p \in I, (p, t) \in A_g$$

$$\text{Couleur } (L_r(p, t)) \in M_r(p), \forall p \in I, (p, t) \in A_r.$$

Le nouveau marquage $M'_g(p)$ est donné par :

$$M'_g(p) = M_g(p) = \text{Couleur } (L_g(p, t)) + \text{Couleur } (L_g(t, p))$$

$$M'_r(p) = M_r(p) = \text{Couleur } (L_r(p, t)) + \text{Couleur } (L_r(t, p)).$$

K représente la série d'apprentissage, telle que : $K = K(P) \cup K(T)$,

Où K est une application de PXT séries non-vides, K(P) est une place d'apprentissage, où K(T) représente l'apprentissage de transition :

$$K(P) = \cup \{K(P_T), K(P_C), K(P_M), K(P_I)\}$$

$$K(T) = \cup \{K(t_j)\}$$

$K^P : P \rightarrow K(P), K^P$ série appliqué P de série de place d'apprentissage $K(P)$

$K^T : T \rightarrow K(T), K^T$ série appliqué T de série de place d'apprentissage $K(T)$

Modélisation de règles du système expert avec le EECSPN

En supposant une règle de base du système (RBS) comprenant 'n' transitions de places ($p_{T1}, p_{T2}, p_{T3}, p_{T4}, \dots, p_{Tn}$). La série contenant 'p' conditions positives, 'q' conditions négatives, 'r' conclusions positives, et 's' conclusions négatives. Dans une opération, les conditions sont les entrées, c'est-à-dire ils sont compris en avance, où les conclusions sont des sorties. Les places consistent en des sorties, qui comprennent les transitions de places, conditions positives, et conditions négatives, c'est-à-dire :

$$I(t_i) = \{p_{ci}, p_{Ti}, cd_{i1}^+, cd_{i2}^+, \dots, cd_{ip}^+, cd_{i1}^-, cd_{i2}^-, \dots, cd_{iq}^-\}$$

Où $I(t_i)$ est une série de places de transition d'entrées « t_i »,

p_{Ti} est la place transition associée avec la transition « t_i »,

$cd_{i1}^+, cd_{i2}^+, \dots, cd_{ip}^+$ sont les « p » conditions positives,

$cd_{i1}^-, cd_{i2}^-, \dots, cd_{iq}^-$ sont les « q » conditions négatives.

Similairement, les places de sortie consistent en des conditions positives et négatives liées avec une conclusion positive ou négative c'est-à-dire :

$$O(t_i) = \{cl_{i1}^+, cl_{i2}^+, \dots, cl_{ir}^+, cl_{i1}^-, cl_{i2}^-, \dots, cl_{is}^-\}$$

Où $O(t_i)$ est une série de sortie de transition « t_i ».

$cl_{i1}^+, cl_{i2}^+, \dots, cl_{ir}^+$ sont les « r » conditions positives,

$cl_{i1}^-, cl_{i2}^-, \dots, cl_{is}^-$ sont les « s » conditions négatives.

Évidemment :

$$P = \bigcup_{i=1}^n I(t_i) \cup O(t_i) \quad \text{et} \quad T = \{t_1, t_2, t_3, \dots, t_n\}.$$

Arc A_g associé avec les occurrences positives et lié avec des conditions négatives et des conclusions positives :

$$A_g = \bigcup_{i=1}^n \{(cd_{ij}^-, t_i), (t_i, cd_{ij}^+) | 1 \leq j \leq p\} \cup \bigcup_{i=1}^n \{(t_i, cl_{ij}^+) | 1 \leq j \leq r\} \cup \bigcup_{i=1}^n \{p_{Ti}, t_i\}$$

Arc A_r associé avec les occurrences négatives et lié avec des conditions négatives et des conclusions négatives :

$$A_r = \bigcup_{i=1}^n \{(cd_{ij}^-, t_i), (t_i, cd_{ij}^-) | 1 \leq j \leq q\} \cup \bigcup_{i=1}^n \{(t_i, cl_{ij}^-) | 1 \leq j \leq s\}$$

Aussi,

$$L_g(cd_{ij}^+, t_i) = L_g(t_i, cd_{ij}^+) = \{n_{ij}^+\},$$

$$\text{Où } n_{ij}^+ \text{ dénote une constante positive : } L_g(t_i, cl_{ij}^+) = \{v_{ij}^+\},$$

$$\text{Où } v_{ij}^+ \text{ dénote une variable positive : } L_r(cd_{ij}^-, t_i) = L_r(t_i, cd_{ij}^-) = \{n_{ij}^-\},$$

$$\text{Où } n_{ij}^- \text{ est une constante négative : } L_r(t_i, cl_{ij}^-) = \{v_{ij}^-\},$$

$$\text{Où } v_{ij}^- \text{ dénote une variable négative : } L_g(p_{Ti}, t_i) = \{v_1, v_2, v_3, \dots, v_k\} \equiv \text{séries de variables}$$

Coloration du modèle EECSPN

Afin de tracer les procédures qui sont incluses dans le EECSPN, les étapes associées de coloration sont données comme suit :

- Initialement, chaque transition est assignée à une pièce verte.
- Chaque endroit autre que l'endroit de transition est assigné à un marquage rouge par défaut.
- Chaque autre place que une transition de place est assignée par défaut à une pièce rouge.
- Avec chaque évènement positif, une pièce verte est assignée.
- Avec chaque évènement négatif prouvant que c'est faux, nous l'assignons par défaut à une pièce rouge.
- Avec chaque évènement négatif qui est assumé comme étant faux, est assigné par défaut à une pièce rouge.

M^0 est le marquage initial, M_g^0 le marquage vert initial, M_{dd}^0 le marquage déduit en rouge, et M_{df}^0 le marquage initial rouge par défaut.

Clairement :

$$\begin{aligned}M_g^0(p_{Ti}) &= C(t_i) \\M_{dd}^0(pTi) &= M_{df}^0(p_{Ti}) = \phi, \forall p_{Ti} \in P_T \\M_g^0(p) &= M_{dd}^0(p) = \phi \\M_{df}^0(p) &= C(p), \forall p \in P - P_T.\end{aligned}$$

Il y a changement de marquage quand une transition est éliminée. Nous posons une transition « t » à « m » places négatives, certaines sont mises par défaut en rouge et les autres sont par la couleur verte. Quand « t » s'éliminée, le marquage « M » se change en marquage M' .

Clairément :

$$M^l = M_g^l \cap (M_{dd}^l \cap M_{df}^l)$$

$$M_g^l(p) = M_g(p) - \text{Couleur}(L_g(p,t)) + \text{Couleur}(L_g(t,p))$$

$$M_{dd}^l(p) = M_{dd}(p) - \text{Couleur}(L_r(p,t)) + \text{Couleur}(L_r^l(p,t)) + \text{Couleur}(L_r(t,p))$$

$$M_{df}^l(p) = M_{df}(p) - \text{Couleur}(L_r(p,t)) - \text{Couleur}(L_g(p,t)) + \text{Couleur}(L_r^l(t,p)) \\ - \text{Couleur}(L_g^-(t,p)) - \text{Couleur}(L_g(p,t)).$$

Notez que $\text{Couleur}(L_r(t,p))$ représente les couleurs rouges traversant les arcs d'entrées, et $(L_r^l(p,t))$ dénote les réactions correspondant aux couleurs rouges. Similairement, $M_{df}^l(p,t)$, $\text{Couleur}(L_r(p,t))$ signifie par défaut les couleurs rouges traversant les arcs d'entrées, et $\text{Couleur}(L_r^l(t,p))$ dénote les réactions correspondant aux couleurs rouges. Donc :

$$\text{Couleur}(L_r(p,t)) = \text{Couleur}(L_r^l(p,t)) \text{ pour les deux } M_{dd}^l(p) \text{ et } M_{df}^l(p),$$

$$\text{Où : } \text{Couleur}(L_g(p,t)) \subseteq \text{Couleur}(L_g(t,p)).$$

Donc, l'équation précédemment mentionnée peut être simplifiée par :

$$M_g^l(p) = M_g(p) - \text{Couleur}(L_g(p,t)) + \text{Couleur}(L_g(t,p))$$

$$M_{dd}^l(p) = M_{dd}(p) + \text{Couleur}(L_r(t,p))$$

$$M_{df}^l(p) = M_{df}(p) - \text{Couleur}(L_g(t,p)) - \text{Couleur}(L_r(t,p))$$

$$M_{df}(p) \cap M_{dd}(p) = \phi$$

$$M_{df}(p) \cap M_g(p) = \phi$$

Ainsi, le chemin pour le désassemblage est défini.

6.3.2 Exemple complet dans le cas d'une monture de lunette

Pour définir la performance du EECSPN en terme d'assemblage/désassemblage, l'exemple du système d'assemblage d'un monture de lunette est considéré pour le modelage et l'analyse. L'exemple comprend trois composants de base : Doublet-1, Espaceur et Doublet-2, qui sont représentés respectivement par 1, 2, et 3. 12 et 23 représentent les sous-assemblages Doublet-1 et Espaceur, et Espaceur et Doublet-2, respectivement, ainsi que 123 représente l'assemblage désiré. 3' signifie l'état du Doublet-2 après une transition d'état interne arrivant après avoir réalisé la procédure d'assemblage/désassemblage.

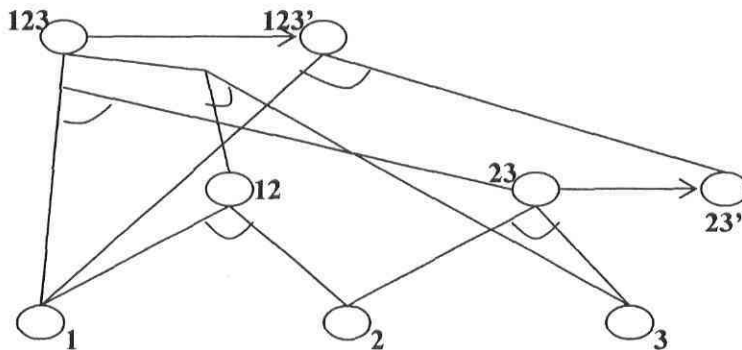


Figure 6.7 : Graphique ET/OU pour l'assemblage d'une monture de lunette

Source : Kumar, Kumar, Shankar et Tiwari [24]

La séquence d'assemblage/désassemblage désirée peut être présentée par un arbre ET/OU. La figure 6.7 illustre les combinaisons/désignations variables des étapes en utilisant l'arbre ET/OU. Une description des composants ou sous-assemblages utilisés dans la figure 6.7 est donnée dans le tableau 6.3.

Tableau 6.3 : Description d'un arbre ET/OU [24]

Composant/sous-assemblages	Désignation
Doublet-1	1
Espaceur	2
Doublet-2	3
Sous-assemblage de Doublet-1 et Espaceur	12
Sous-assemblage de Doublet-2 et Espaceur	23
Sous-assemblage de Espaceur et Doublet-2 (considérant l'état interne de transition)	23'
Assemblage du Doublet-1, Espaceur et Doublet-2	123
Assemblage du Doublet-1, Espaceur et Doublet-2 (considérant l'état interne de transition)	123'

Assemblage/désassemblage des réseaux de Petri

La séquence d'assemblage désirée peut être illustrée par un graphique du réseau de Petri. Ce dernier, pour ce cas en particulier, comprend huit transitions pour les opérations d'assemblage/désassemblage faisables, quatre transitions signifiant l'état interne de transition et une transition pour l'assemblage et une pour le désassemblage incluant un état interne du composant avec transition (Doublet-2).

Tableau 6.4 : Description du réseau de Petri [24]

Place	Désignation	Composant/sous-assemblages
p ₁	1	Doublet-1
p ₂	2	Espaceur
p ₃	3	Doublet-2
p ₄	12	Sous-assemblage de Doublet-1 et Espaceur
p ₅	23	Sous-assemblage de Doublet-2 et Espaceur
p ₆	23'	Sous-assemblage de Espaceur et Doublet-2 (considérant l'état interne de transition)
p ₇	123	Assemblage du Doublet-1, Espaceur et Doublet-2
p ₈	123'	Assemblage du Doublet-1, Espaceur et Doublet-2 (considérant l'état interne de transition)

La figure 6.8 représente bien ces transitions et la description est donnée dans le tableau 6.4. Les séquences d'assemblage sont $t_1 \rightarrow t_{10}$ et $t_4 \rightarrow t_7$. Si l'état interne de transition est considéré, alors la séquence d'assemblage est changé par $t_4 \rightarrow t_6 \rightarrow t_{14}$. Similairement, une séquence de désassemblage peut être $t_8 \rightarrow t_2$ ou $t_9 \rightarrow t_3$. Dans le cas d'un état interne de transition, nous avons $t_{11} \rightarrow t_5 \rightarrow t_3$, $t_{12} \rightarrow t_8 \rightarrow t_2$ ou $t_{12} \rightarrow t_9 \rightarrow t_3$.

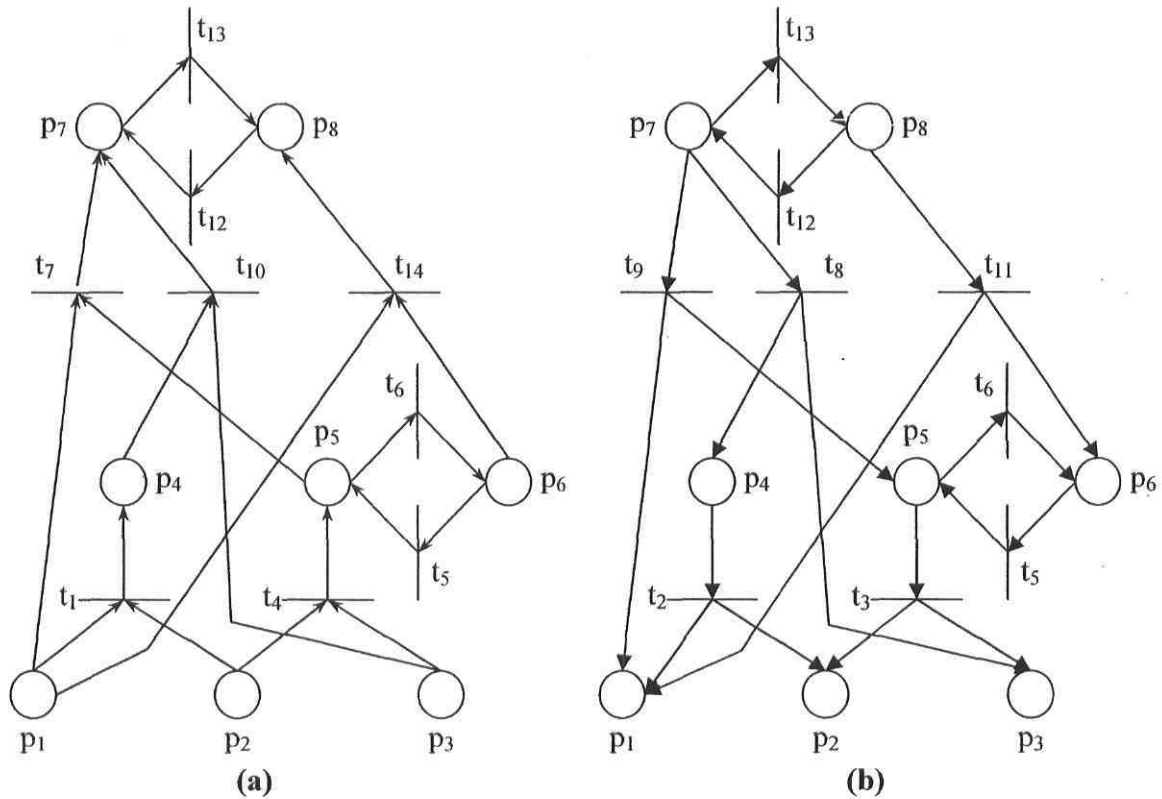
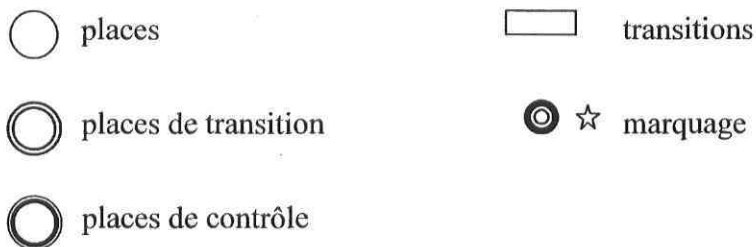


Figure 6.8 : Procédure d'assemblage/désassemblage avec un réseau de Petri

Source : Kumar, Kumar, Shankar et Tiwari [24]

La légende pour les figures 6.9 à 6.14 est :



Assemblage/désassemblage en utilisant le EECSPN

La figure 6.9 illustre le modèle EECSPN pour l'assemblage de la monture de lunette. Le marquage initial et les futurs changements de marquage ont un rôle important dans la description complète de la procédure d'assemblage/désassemblage.

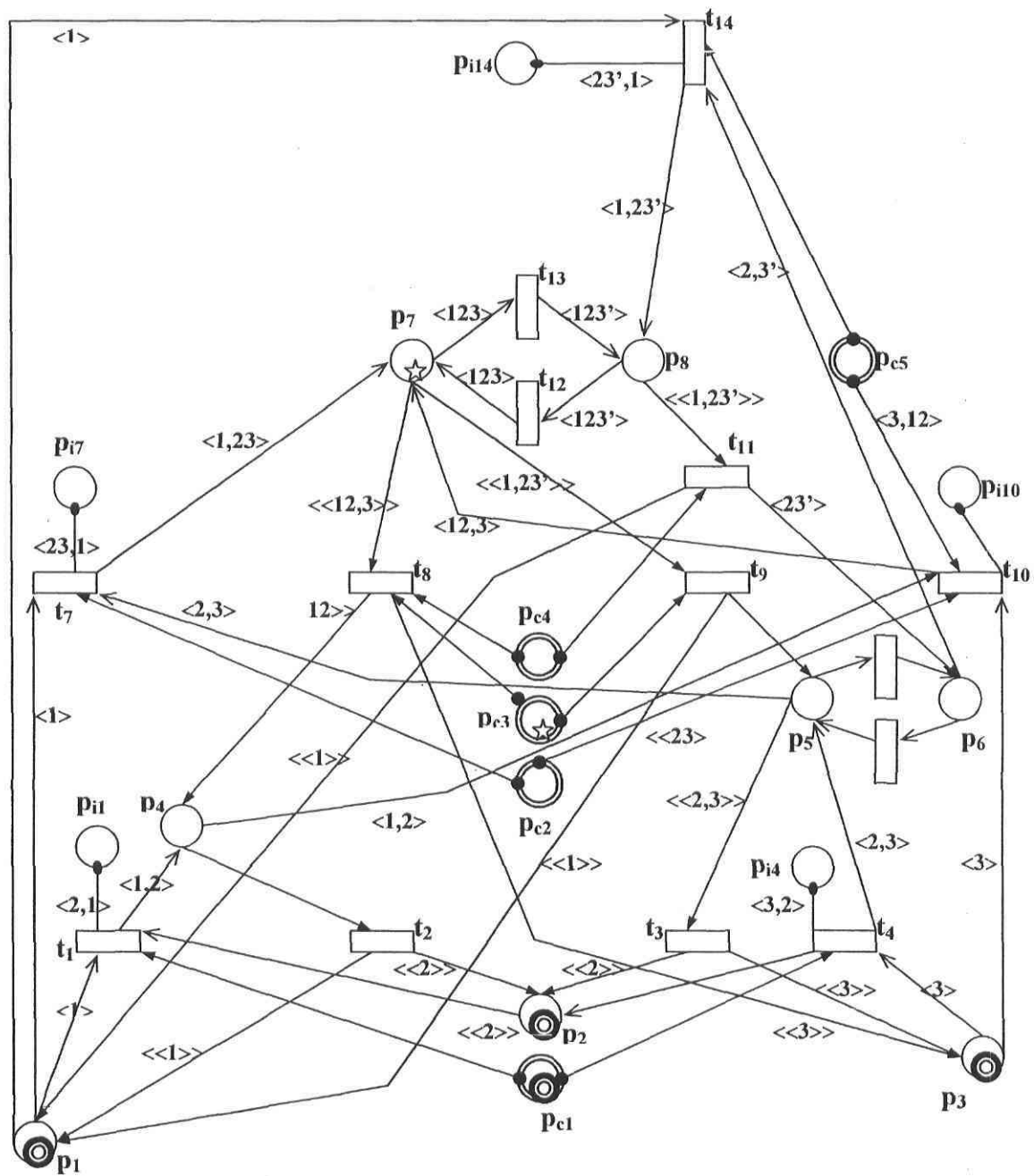


Figure 6.10 : Position des pièces dans le modèle EECSPN à la première étape

Source : Kumar, Kumar, Shankar et Tiwari [24]

Pour l'assemblage/désassemblage, la figure 6.10 montre le statut initial de marquage M^0 , la figure 6.11 montre le statut intermédiaire de marquage M^1 , obtenu après le changement de la transition t_4 . La figure 6.12 représente le statut final de marquage M^2 après le

Les arcs et niveaux sont associés à la relation entre les places et les transitions. Des attributs relatifs impliqués dans de tels rapports sont indiqués par des marquages

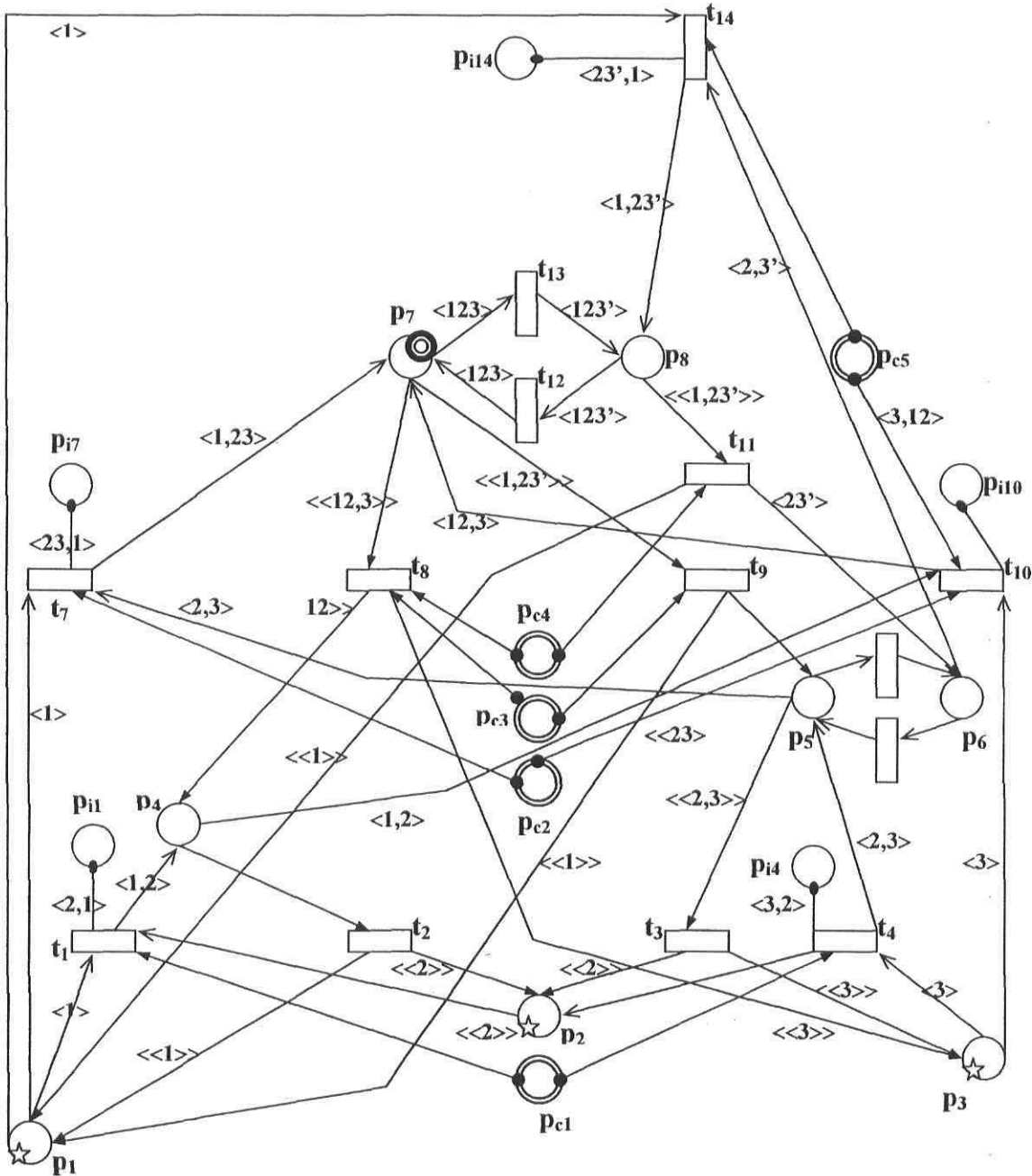


Figure 6.12 : Position des pièces dans le modèle EECSPN à la troisième étape

Source : Kumar, Kumar, Shankar et Tiwari [24]

Par exemple, la couleur verte $\langle 1,23 \rangle$ dans la figure 6.10 montre que l’assertion « 1 est au dessus de 23 » est vrai, aussi la couleur rouge $\langle 2,1 \rangle$ signifie que l’affirmation « 2 est au dessus de 1 » est fausse. Par analogie, dans un réseau de Petri normal, la marquage démontre un événement positif, tandis que dans le EECSPN, le marquage vert décrit un événement positif, ainsi que le marquage rouge désigne un événement négatif (non-désiré).

Les niveaux des arcs sont donnés dans le tableau 6.6. Les séries de couleurs pour l’assemblage et le désassemblage sont données dans le tableau 6.7.

Tableau 6.5 : Évolution des places dans le modèle EECSPN [24]

Places	Désignation	Places	Désignation
p ₁	1	p _{C3}	Place de contrôle entre t ₈ et t ₉
p ₂	2	p _{C4}	Place de contrôle entre t ₈ et t ₁₁
p ₃	3	p _{C5}	Place de contrôle entre t ₁₀ et t ₁₄
p ₄	12	p _{T1}	Place de transition pour la transition t ₁
p ₅	21	p _{T2}	Place de transition pour la transition t ₂
p ₆	23	p _{T3}	Place de transition pour la transition t ₃
p ₇	32	p _{T4}	Place de transition pour la transition t ₄
p ₈	23'	p _{T5}	Place de transition pour la transition t ₅
p ₉	123	p _{T6}	Place de transition pour la transition t ₆
p ₁₀	231	p _{T7}	Place de transition pour la transition t ₇
p ₁₁	312	p _{T8}	Place de transition pour la transition t ₈
p ₁₂	123' (excepté la sortie avec l'état interne de transition)	p _{T9}	Place de transition pour la transition t ₉
p ₁₃	231'	p _{T10}	Place de transition pour la transition t ₁₀
p _{C1}	Place de contrôle entre t ₁ et t ₄	p _{T11}	Place de transition pour la transition t ₁₁
p _{C2}	Place de contrôle entre t ₇ et t ₁₀	p _{T14}	Place de transition pour la transition t ₁₄

Critère d'évaluation pour le sous-assemblage

La détermination du chemin optimal est en premier lieu l'étape vers une planification réussit de la procédure d'assemblage/désassemblage. À cette fin, les facteurs impliqués, leurs effets et la manière de trouver le chemin optimal sont donnés ci-dessous. Les facteurs affectant l'optimisation des assemblages/désassemblages sont : la stabilité, la direction et le niveau de manipulation. Les facteurs mentionnés ci-dessus fournissent une balance indirecte au coût mesuré encouru dans des opérations d'assemblage/désassemblage.

Stabilité :

La stabilité des sous-assemblage, s_{ai} , peut être définie sur la base d'un ensemble de groupes flottants de composants, $F_c(s_{ai})$, et des groupes discontinus de composants, $D_c(s_{ai})$, inclus dans s_{ai} . Le groupe de composant de s_{ai} , $F_c(s_{ai})$, est défini comme flottant s'il est connecté au reste de s_{ai} par une liaison flottante seulement. Un groupe de composant de s_{ai} , $D_c(s_{ai})$, est défini comme étant discontinué s'il n'y a pas de liaison connectée aux composants restants de s_{ai} . Le sous-graphe de s_{ai} est $G(s_{ai})$, La série flottante coupée est C_S et la liberté de mouvement locale de C_S est $LFM(s_{ai})$. Ce dernier peut être défini comme la liberté de mouvement locale de $F_C(s_{ai})$. Donc :

$$LFM(C_S) = \cap LFM(C_S; FC(s_{ai}), s_{ai} - F_C(s_{ai})), \forall p \text{ tels que } p \in C_S.$$

Tableau 6.6 : Évolution des symboles dans le modèle EECSPN [24]

Places	$P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, PC_1, PC_2, PC_3, PC_4, PC_5, PT_1, PT_2, PT_3, PT_4, PT_5, PT_6, PT_7, PT_8, PT_9, PT_{10}$.
Transitions	$t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{14}$.
A_g (assemblage) pour les places normales	$g_1=(p_1,t_1), g_2=(p_2,t_2), g_3=(p_1,t_{14}), g_4=(p_2,t_1), g_5=(p_2,t_4), g_6=(p_3,t_4), g_7=(p_3,t_{10}), g_8=(p_4,t_{10}), g_9=(t_4,p_1), g_{10}=(t_4,p_6), g_{11}=(p_6,t_7), g_{12}=(p_6,t_6), g_{13}=(t_5,p_6), g_{14}=(t_6,p_8), g_{15}=(p_8,t_5), g_{16}=(p_8,t_{14}), g_{17}=(p_9,t_{13}), g_{18}=(t_{12},p_9), g_{19}=(t_7,p_9), g_{20}=(t_{10},p_{20}), g_{21}=(t_{14},p_{12}), g_{22}=(t_{13},p_{12}), g_{23}=(p_{12},t_{12})$.
A_g (assemblage) pour les places de contrôles	$g_{24}=(PC_1,t_1), g_{25}=(PC_1,t_4), g_{26}=(PC_2,t_7), g_{27}=(PC_2,t_{10}), g_{28}=(PC_5,t_{10}), g_{29}=(PC_5,t_{14})$.
A_g (assemblage) pour les places de transition	$g_{30}=(PT_1,t_1), g_{31}=(PT_4,t_4), g_{32}=(PT_5,t_7), g_{33}=(PT_8,t_8), g_{34}=(PT_{10},t_{14})$.
A_g (désassemblage) pour les places normales	$g_{35}=(t_{11},p_1), g_{36}=(t_9,p_1), g_{37}=(t_2,p_1), g_{38}=(t_2,p_2), g_{39}=(t_3,p_2), g_{40}=(t_3,p_3), g_{41}=(t_8,p_3), g_{42}=(t_8,p_4), g_{43}=(p_4,t_2), g_{44}=(p_6,t_3), g_{45}=(t_9,p_6), g_{46}=(t_{11},p_8), g_{47}=(p_9,t_8), g_{48}=(p_9,t_9), g_{49}=(p_{12},t_{11})$.
A_g (désassemblage) pour les places de contrôles	$g_{50}=(PC_3,t_8), g_{51}=(PC_3,t_9), g_{52}=(PC_4,t_8), g_{53}=(PC_4,t_{11})$.
A_g (désassemblage) pour les places de transition	$g_{54}=(PT_2,t_2), g_{55}=(PT_3,t_3), g_{56}=(PT_6,t_8), g_{57}=(PT_7,t_9), g_{58}=(PT_9,t_{11})$.
A_r (assemblage)	$r_1=(p_1,t_5), r_2=(t_4,p_7), r_3=(t_7,p_{10}), r_4=(t_{10},p_{11}), r_5=(t_{14},p_{13})$.
$L(g)$ (assemblage) pour les places normales	$L(g_1)=\langle 1 \rangle, L(g_2)=\langle 1 \rangle, L(g_3)=\langle 1 \rangle, L(g_4)=\langle 2 \rangle, L(g_5)=\langle 2 \rangle, L(g_6)=\langle 3 \rangle, L(g_7)=\langle 3 \rangle, L(g_8)=\langle 12 \rangle, L(g_9)=\langle 1,2 \rangle, L(g_{10})=\langle 2,3 \rangle, L(g_{11})=\langle 23 \rangle, L(g_{12})=\langle 23 \rangle, L(g_{13})=\langle 23 \rangle, L(g_{14})=\langle 23' \rangle, L(g_{15})=\langle 23' \rangle, L(g_{16})=\langle 23' \rangle, L(g_{17})=\langle 123 \rangle, L(g_{18})=\langle 123 \rangle, L(g_{19})=\langle 1,23 \rangle, L(g_{20})=\langle 12,3 \rangle, L(g_{21})=\langle 1,23' \rangle, L(g_{22})=\langle 123' \rangle, L(g_{23})=\langle 123' \rangle$.
$L(g)$ (désassemblage) pour les places normales	$L(g_{35})=\langle \langle 1 \rangle \rangle, L(g_{36})=\langle \langle 1 \rangle \rangle, L(g_{37})=\langle \langle 1 \rangle \rangle, L(g_{38})=\langle \langle 2 \rangle \rangle, L(g_{39})=\langle \langle 2 \rangle \rangle, L(g_{40})=\langle \langle 1,2 \rangle \rangle, L(g_{41})=\langle \langle 3 \rangle \rangle, L(g_{42})=\langle \langle 12 \rangle \rangle, L(g_{43})=\langle \langle 1,2 \rangle \rangle, L(g_{44})=\langle \langle 2,3 \rangle \rangle, L(g_{45})=\langle \langle 23 \rangle \rangle, L(g_{46})=\langle \langle 23' \rangle \rangle, L(g_{47})=\langle \langle 12,3 \rangle \rangle, L(g_{48})=\langle \langle 1,23 \rangle \rangle, L(g_{49})=\langle \langle 1,23 \rangle \rangle$.
$L(r)$ (assemblage)	$L(r_1)=\langle 2,1 \rangle, L(r_2)=\langle 3,2 \rangle, L(r_3)=\langle 23,1 \rangle, L(r_4)=\langle 3,12 \rangle, L(r_5)=\langle 23',1 \rangle$.
Constantes (N)	1, 2, 3, 3', 12, 21, 23, 32, 23', 123, 123'.
Variable (V)	ϕ
Constantes et variables (U)	1, 2, 3, 3', 12, 21, 23, 32, 23', 123, 123'.

Direction :

La direction est un facteur important affectant la performance d'assemblage. Pour un meilleur résultat, l'évaluation des directions d'assemblage doit être basée sur les directions locales d'assemblage, s_{ai} et $A-s_{ai}$. Le coût relatif d'assemblage dû à la stabilité associée avec s_{ai} et $A-s_{ai}$ est illustré dans le tableau 6.8.

Tableau 6.7 : Étape initiale et son évolution [24]

Séries de couleurs dans l'étape initiale				Évolution les séries de couleurs dans l'étape initiale			
Séries de couleurs				Séries de couleurs			
Places	Vert	Rouge		Places	Vert	Rouge	
		Rouge par défaut	Rouge par déduction			Rouge par défaut	Rouge par déduction
P1	1	1	0	P1	1	1	0
P2	1	1	0	P2	1	1	0
P3	1	1	0	P3	1	1	0
P4	0	1	0	P4	0	1	0
P5	0	1	0	P5	0	1	0
P6	0	1	0	P6	0	1	0
P7	0	1	0	P7	0	1	0
P8	0	1	0	P8	0	1	0
P9	0	1	0	P9	0	1	0
P10	0	1	0	P10	0	1	0
P11	0	1	0	P11	0	1	0
P12	0	1	0	P12	0	1	0
P13	0	1	0	P13	0	1	0
PC1	1	1	0	PC1	1	1	0
PC2	1	1	0	PC2	1	1	0
PC3	0	1	0	PC3	0	1	0
PC4	0	1	0	PC4	0	1	0
PC5	1	1	0	PC5	1	1	0
PT1	1	0	0	PT1	1	0	0
PT2	0	0	0	PT2	0	0	0
PT3	0	0	0	PT3	0	0	0
PT4	1	0	0	PT4	1	0	0
PT5	1	0	0	PT5	1	0	0
PT6	0	0	0	PT6	0	0	0
PT7	0	0	0	PT7	0	0	0
PT8	1	0	0	PT8	1	0	0
PT9	0	0	0	PT9	0	0	0
PT10	1	0	0	PT10	1	0	0

Niveau de Manipulation :

Les sous-assemblages sujets à la réorientation et/ou à une transition avec un joint effectif doivent être facile à manipuler avec les mains ou des outils. Le niveau de manipulation de s_{ai} est utilisé pour quantifier l'efficacité dans l'orientation et la manipulation s_{ai} .

L'efficacité d'orientation peut être mesurée dans une base de symétrie et de polarité marquée dans la géométrie et le poids de s_{ai} , considérant que l'efficacité de manipulation peut être mesurée dans une base de régularité de la dimension, forme, poids, flexibilité et la fragilité de s_{ai} . Le niveau de manipulation γ peut être déterminé pour un sous-assemblage, s_{ai} , par :

$$\gamma(s_{ai}) = \frac{\Sigma(\text{les notes de l'orientation et de l'efficacité de manipulation pour } s_{ai})}{\Sigma(\text{les notes nominales de l'orientation et de l'efficacité de manipulation})}$$

Le haut et bas niveaux de la planification des procédures d'assemblage/désassemblage sont utilisés pour réduire la complexité de EECSPN dans une perspective de recherche de séquences. C'est deux niveaux sont détaillés dans la suite de l'exemple.

Tableau 6.8 : Détermination du chemin optimal [24]

Évaluation	Poid relatif
1. Instabilité de s_{ai} ou $A - s_{ai}$	∞ (rejet de la sélection)
2. Pas d'assemblage stable commun entre s_{ai} et $A - s_{ai}$	∞ (rejet de la sélection)
3. Nombre de dispositifs de fixations requis pour stabiliser $D_c(s_{ai})$ et $D_c(A - s_{ai})$, qui sont représentés par les nombres $D_c(s_{ai})$ et $D_c(A - s_{ai})$	15/dispositif
4. Nombre de dispositifs de fixations requis pour stabiliser $F_c(s_{ai})$ et $F_c(A - s_{ai})$, avec plus d'un degré de translation de mouvement. Ce nombre peut être calculé par les A-neuds de l'assemblage A, qui sont inclus dans $F_c(s_{ai})$ et $F_c(A - s_{ai})$ requérant une stabilisation.	15/dispositif
5. Réorientation de s_{ai} et $A - s_{ai}$	10/réorientation

Planification des procédures d'assemblage/désassemblage de Haut-Niveau

La procédure de planification d'assemblage consiste en une sélection de chemins appropriés et du guide opérationnel, qui dirige vers l'assemblage final désiré. Le plan de

la procédure pour notre exemple est présentée à la figure 6.13. Les hypothèses suivantes sont posées pour la planification des procédures :

- Durant une opération d'assemblage, exactement deux composants sont assemblés dans chaque opération d'assemblage.
- Tous les accessoires perdus durant les opérations d'assemblage sont récupérés dans les opérations de désassemblage.

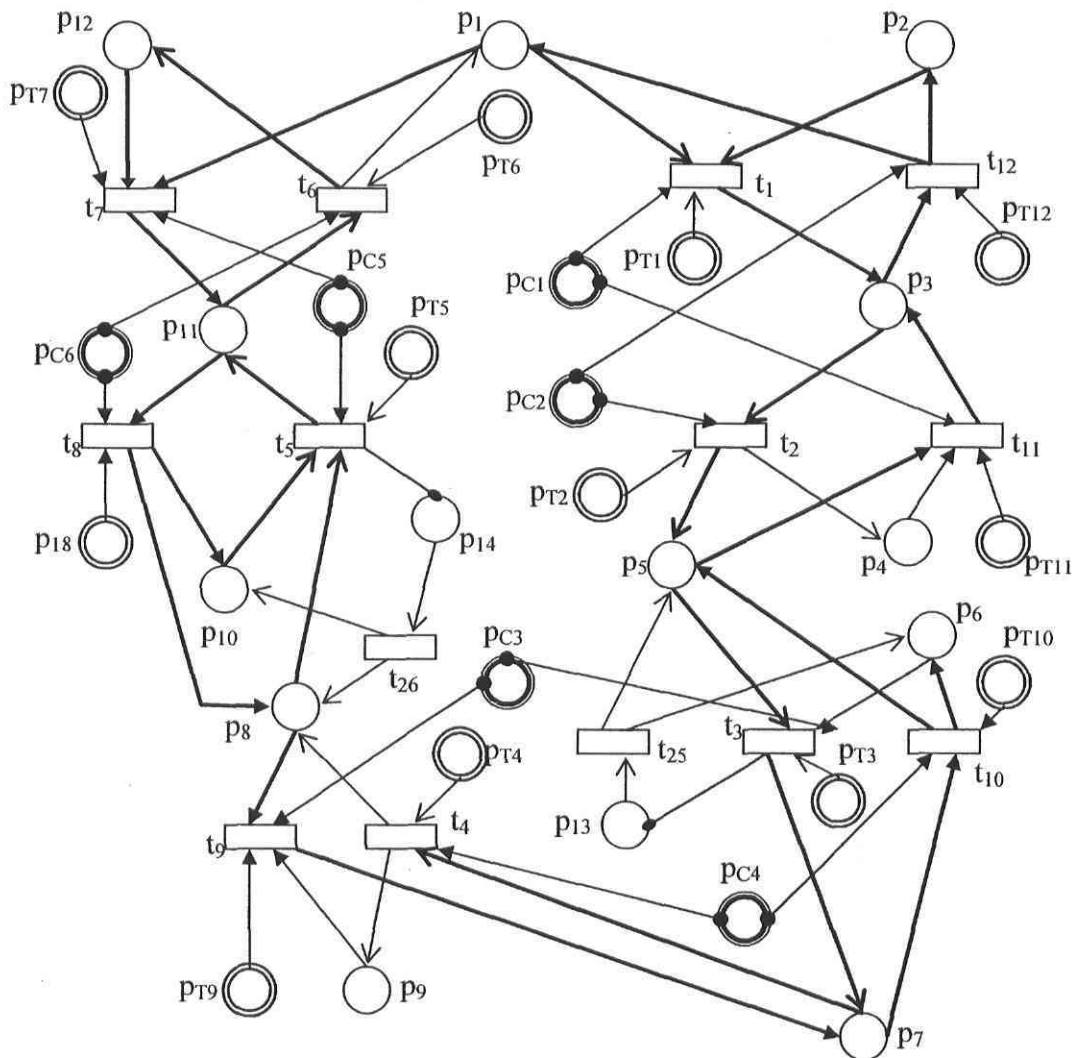


Figure 6.13 : *EECSPN de haut-niveau*

Source : Kumar, Kumar, Shankar et Tiwari [24]

Les procédures considèrent les sous-assemblages non-désirés formés durant les opérations d'assemblage et le désassemblage immédiat de ces composants de retour à

leurs positions pour prendre en considération les opérations d'assemblage futures. La probabilité de telles occurrences est prise en considération par la stochasticité du graphique du Réseau de Petri (figure 6.13). Les flux de contrôle de places prennent en considération les conflits aussi bien qu'une préférence de transition vers une autre. La description des différents places et transitions est donnée au tableau 6.9. B1, B2 et B3 servent comme tampon pour les composants Doublet-1, Espaseur et Doublet-2, respectivement. Nous supposons qu'ils ne sont jamais vide et qu'ils assurent que les procédures d'assemblage/désassemblage sont libre d'obstacle.

Les transitions du Réseau de Petri sont utilisées avec quatre opérations de base :

- Tenir(X,Y) implique tenir Y avec X.
- Séparer(X,Y) implique séparer Y de X.
- Lacher(X,Y) implique lacher Y et X.
- Retirer(X,Y) envoie les composants/désassemblage X et Y de leurs positions originales.
-

Tableau 6.9 : Description des places et des transitions pour le haut-niveau [24]

Transition	Opération	Places	Composant/sous-assemblage
t ₁	Tenir(R,D ₁)	P ₁	R
t ₂	Séparer(RD ₁ ,B ₁)	P ₂	D ₁ B ₁
t ₃	Tenir(RD ₁ ,S)	P ₃	RD ₁ B ₁
t ₄	Séparer(RD ₁ ,S,B ₂)	P ₄	B ₁
t ₅	Tenir(R,D ₁ S,D ₂)	P ₅	RD ₁
t ₆	Lacher(R,D ₁ SD ₂ B ₃)	P ₆	SB ₂
t ₇	Tenir(R,D ₁ SD ₂ B ₃)	P ₇	RD ₁ SB ₂
t ₈	Séparer(RD ₁ S,D ₂ B ₃)	P ₈	RD ₁ S
t ₉	Tenir(RD ₁ S,B ₂)	P ₉	B ₂
t ₁₀	Lacher(RD ₁ ,SB ₂)	P ₁₀	D ₂ B ₃
t ₁₁	Tenir(RD ₁ ,B ₁)	P ₁₁	RD ₁ SD ₂ B ₃
t ₁₂	Lacher(R,D ₁ B ₁)	P ₁₂	D ₁ SD ₂ B ₁
t ₂₅	Retirer(RD ₁ ,SB ₂)	P ₁₃	RSD ₂ B ₁
t ₂₆	Retirer(RD ₁ S,D ₂ B ₃)	P ₁₄	RD ₂ B ₃ D ₁ S
		P _{Ci}	Places de contrôle (i=1,2,...,6)

Pour l'assemblage, les places p₁, p₂, p₆, et p₁₀ ont les pièces initiales. La séquence d'assemblage la plus faisable trouvée durant l'assemblage robotisé est :

$$t_1 : \text{tenir}(R, D_1) \rightarrow t_2 : \text{séparer}(RD_1, B_1) \rightarrow t_3 : \text{tenir}(RD_1, S)$$

→ t₄ : séparer(RD₁S,B₂) → t₅ : tenir(RD₁S,D₂)
 → t₆ : lâcher(RD₁SD₂,B₂)

Pour le désassemblage, les places p₁₂, p₂, et p₄ ont les pièces initiales. La séquence de désassemblage la plus faisable trouvée durant le désassemblage robotisé est :

t₇ : tenir(R, D₁SD₂B₃) → t₈ : séparer(RD₁S,D₂B₃) → t₉ : lâcher(RD₁S,B₂)
 → t₁₀ : lâcher(RD₁,SB₂) → t₁₁ : tenir(RD₁,B₁)
 → t₁₂ : lâcher(R,D₁B₁)

Matrice de probabilité de transition : ayant comme référence la matrice suivante, il est clair que durant les déplacements du chemin d'assemblage, des transitions t₃ et t₅, deux produits peuvent être formés.

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & y_1/(x_1 + y_1) & x_1/(x_1 + y_1) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_2/(x_2 + y_2) & x_2/(x_2 + y_2) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Par conséquent, nous retrouvons un élément d'induction de probabilité dans la séquence :
 x₁ = chances en faveur de la formation du produit désiré après l'élimination de la transition t₃.

y₁ = chances en faveur de la formation du produit non désiré après l'élimination de la transition t₃.

x₂ = chances en faveur de la formation du produit désiré après l'élimination de la transition t₅. y₂ = chances en faveur de la formation du produit non désiré après l'élimination de la transition t₅.

a_{ij} est la probabilité pour le changement de marquage M_i en marquage M_j. Donc :

$$\begin{aligned}
a_{00} &= 0, a_{01} = 1, a_{02} = 0, a_{03} = 0, a_{04} = 0, a_{05} = 0, a_{06} = 0, \\
a_{10} &= 0, a_{11} = 0, a_{12} = 1, a_{13} = 0, a_{14} = 0, a_{15} = 0, a_{16} = 0, \\
a_{20} &= 0, a_{21} = 0, a_{22} = y_1 / (x_1 + y_1), a_{23} = x_1 / (x_1 + y_1), a_{24} = 0, a_{25} = 0, a_{26} = 0, \\
a_{30} &= 0, a_{31} = 0, a_{32} = 0, a_{33} = 0, a_{34} = 1, a_{35} = 0, a_{36} = 0, \\
a_{40} &= 0, a_{41} = 0, a_{42} = 0, a_{43} = 0, a_{44} = y_2 / (x_2 + y_2), a_{45} = x_2 / (x_2 + y_2), a_{46} = 0, \\
a_{50} &= 0, a_{51} = 0, a_{52} = 0, a_{53} = 0, a_{54} = 0, a_{55} = 0, a_{56} = 1, \\
a_{60} &= 0, a_{61} = 0, a_{62} = 0, a_{63} = 0, a_{64} = 0, a_{65} = 0, a_{66} = 1.
\end{aligned}$$

Planification des procédures de Bas-Niveau

La planification de l'opération d'assemblage/désassemblage de Bas-Niveau implique l'analyse des opérations de Haut-Niveau, en simplifiant les opérations. Chaque opération d'assemblage du haut niveau est remplacée par une séquence d'activités, nommée « dans la direction de », « tenir » et « joindre ». Similairement, chaque opération de désassemblage dans un haut niveau est remplacé par une autre séquence de mouvement « dans la direction de », « lâcher » et « détacher ». Conséquent, pour l'assemblage et le désassemblage, les places représentant les composants actives sont additionnées. Les hypothèses du haut niveau sont aussi applicables dans ce contexte. Par les règles suivantes, le haut niveau peut être transformé en réseau de Petri de bas-niveau. Ce dernier utilise les opérations de base suivante :

- Tenir(X,Y), implique, tenir Y avec X.
- Séparer(X,Y), implique, séparer Y de X.
- Lâcher(X,Y), implique, lâcher Y de X.
- Dans la direction(X,Y), implique, bouger X dans la direction de Y.
- Bouger(X,temp), implique, donner à X une opération temporaire pour obtenir la forme désirée.
- Bouger(X,in), implique, bouger X à son emplacement de repos.
- Retirer(X,Y), implique, envoyer composants/sous-assemblages X et Y dans sa position originale.

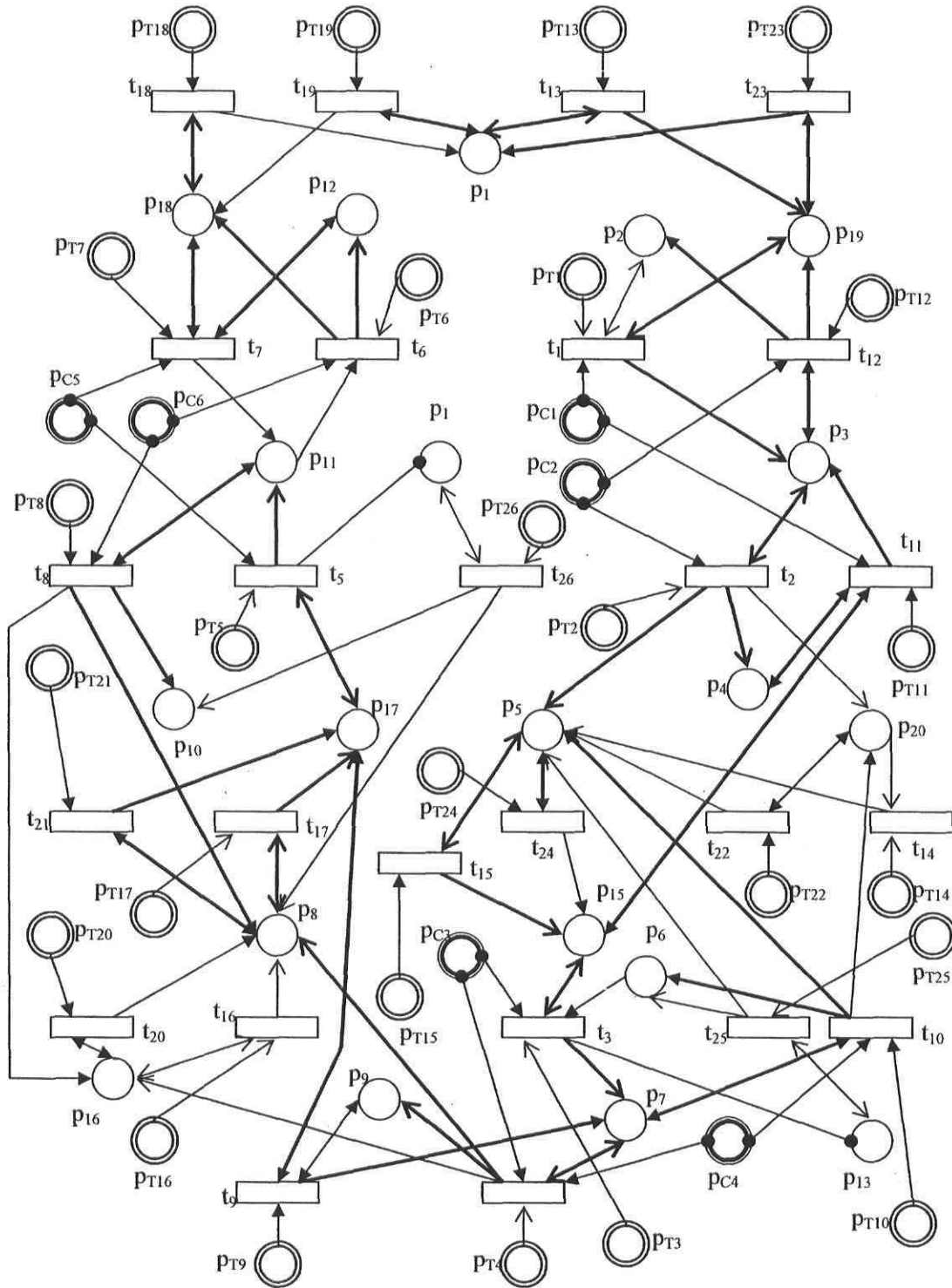


Figure 6.14 : EECSPN de bas-niveau

Source : Kumar, Kumar, Shankar et Tiwari [24]

Règles :

- Chaque opération d'assemblage est décomposé en opérations de mouvement « dans la direction », « tenir », et « séparer ».
- Chaque opération de désassemblage est décomposé en opérations de mouvement « dans la direction », « tenir », et « séparer ».
- Correspondant avec chaque opération générée, un nombre adéquat de places est aussi généré.
- L'opération temporaire « bouger » est aussi considérée durant l'assemblage/désassemblage cela correspondant à quelques conditions impliquées pendant l'opération et apportent le composant assemblé/désassemblé de nouveau dans la procédure d'assemblage/désassemblage.
- Les arcs dirigés sont additionnés selon la convention d'assemblage ou de désassemblage

Tableau 6.10 : Description des places et des transitions pour le bas-niveau [24]

Transition	Opération	Places	Composant/sous-assemblage
t ₁	Tenir(R,D ₁)	P ₁	R
t ₂	Séparer(RD ₁ ,B ₁)	P ₂	D ₁ B ₁
t ₃	Tenir(RD ₁ ,S)	P ₃	RD ₁ B ₁
t ₄	Séparer(RD ₁ ,S,B ₂)	P ₄	B ₁
t ₅	Tenir(R,D ₁ S,D ₂)	P ₅	RD ₁
t ₆	Lacher(R,D ₁ SD ₂ B ₃)	P ₆	SB ₂
t ₇	Tenir(R,D ₁ SD ₂ B ₃)	P ₇	RD ₁ SB ₂
t ₈	Séparer(RD ₁ S,D ₂ B ₃)	P ₈	RD ₁ S
t ₉	Tenir(RD ₁ S,B ₂)	P ₉	B ₂
t ₁₀	Lacher(RD ₁ ,SB ₂)	P ₁₀	D ₂ B ₃
t ₁₁	Tenir(RD ₁ ,B ₁)	P ₁₁	RD ₁ SD ₂ B ₃
t ₁₂	Lacher(R,D ₁ B ₁)	P ₁₂	D ₁ SD ₂ B ₁
t ₁₃	Dans la direction(R,D ₁)	P ₁₃	RSD ₂ B ₁
t ₁₄	Bouger(RD ₁ ,temp)	P ₁₄	RD ₂ B ₃ D ₁ S
t ₁₅	Dans la direction(RD ₁ ,S)	P ₁₅	RD ₁
t ₁₆	Bouger(RD ₁ S,temp)	P ₁₆	RD ₁ S
t ₁₇	Dans la direction(RD ₁ S,D ₂)	P ₁₇	RD ₁ S
t ₁₈	Bouger(R,IN)	P ₁₈	R
t ₁₉	Dans la direction(R,D ₂ SD ₂ B ₃)	P ₁₉	R
t ₂₀	Bouger(RD ₁ S,temp)	P ₂₀	RD ₁
t ₂₁	Dans la direction(RD ₁ S,B ₂)	P _{Ci}	Places de contrôle (i=1,2,...,6)
t ₂₂	Bouger(RD ₁ ,temps)		
t ₂₃	Bouger(R,IN)		
t ₂₄	Dans la direction(RD ₁ ,B ₁)		
t ₂₅	Retirer(RD ₁ ,SB ₂)		

Le réseau de Petri de Bas-Niveau obtenu, est illustré dans la figure 6.14.

La séquence d'assemblage la plus faisable durant l'assemblage robotisé est :

t_{13} : dans la direction(R,D₁) → t_1 : tenir(R,D₁) → t_2 : séparer(RD₁,B₁)
→ t_{15} : dans la direction(RD₁,S) → t_3 : tenir(RD₁,S)
→ t_4 : séparer(RD₁S,B₂) → t_{17} : dans la direction(RD₁S,D₂)
→ t_5 : tenir(RD₁S,D₂) → t_6 : lâcher(R,D₁SD₂B₃)
→ t_{18} : bouger(R,IN).

La séquence la plus faisable durant le désassemblage robotisé est :

t_{19} : dans la direction(R,D₁SD₂B₃)
→ t_7 : tenir(R,D₁SD₂B₃) → t_8 : séparer(RD₁S,D₂B₃)
→ t_{21} : dans la direction(RD₁S,B₂) → t_9 : tenir(RD₁S,B₂)
→ t_{10} : lâcher(RD₁,SB₂) → t_{24} : dans la direction(RD₁,B₁)
→ t_{11} : tenir(RD₁,B₁) → t_{12} : lâcher(R,D₁B₁)
→ t_{23} : bouger(R,IN).

Bien que l'exemple que nous venons de présenté soit très simple (pour facilité la visualisation), il illustre toute la procédure pour trouver la séquence optimale autant pour l'assemblage que pour le désassemblage. Ce qui fait la force de cette méthode c'est l'intégration de l'assemblage. Il est ainsi possible de faire une analyse complète de l'assemblage et du désassemblage d'un produit dans une même méthode complète. Le EESCPN, permet de contourner beaucoup d'obstacle tel que les composants et les configurations de produits complexes.

7 Stratégies basées sur l'aspect environnemental et économique

Les stratégies suivantes misent sur l'aspect environnemental et économique du désassemblage de produits, c'est-à-dire, la remise à neuf, la nouvelle fabrication et le recyclage du produit lors de sa récupération en fin de vie. Leur point fort c'est qu'elles utilisent les valeurs de fin de vie des produits et les introduisent directement dans leur méthode de désassemblage. Tout d'abord, l'évaluation économique des opérations de désassemblage pour le recyclage, la remise à neuf et la réutilisation, fait une analyse des procédures de récupération pour leur permettre de construire leur algorithme de désassemblage adapté à l'aspect environnementale. La méthode du réseau logique de désassemblage présente une résolution rapide dans un objectif de recyclage des produits. Et finalement, l'algorithme génétique présenté avec introduction de l'analyse environnementale, nous permet de faire l'analyse du désassemblage de façon différente.

Johnson et Wang [20] présentent une façon très intéressante de voir la conception pour le désassemblage. Ils structurent une méthodologie en quatre niveaux d'analyse du désassemblage. Ces quatre niveaux sont la base de plusieurs travaux intéressants comme la construction des matrices de liens et de désassemblage, les arbres de désassemblage et une analyse économique importante dans l'étude du désassemblage. Ils présentent un modèle économique et quatre étapes pour générer des séquences de désassemblages optimales. Ils regardent aussi la compatibilité des matériaux et les nœuds de disposition du graphique en arbre. Ces travaux sont à la base de plusieurs autres travaux des mêmes auteurs et d'auteurs différents dans les années suivantes. Johnson et Wang [21] développent des facteurs économiques dans les cédules des opérations de désassemblage pour les opportunités de récupération des matériaux (MRO). Ils développent d'une méthode quantitative pour l'analyse du désassemblage. Les buts sont d'avoir une procédure de planification du désassemblage efficace et de générer une séquence de désassemblage optimale avec une maximisation des profits. Donc, le but principal est de générer une séquence optimale de désassemblage en terme de considération économique

des opportunités de récupération des matériaux à la fin du cycle de vie. Les éléments pertinents expliqués sont la formulation du problème, l'analyse du désassemblage des produits, la première analyse du produit à désassembler, l'analyse économique, la décision à appliquer durant le désassemblage, la génération d'une séquence de désassemblage optimale, la formation de la matrice de marge profit/perte (PLM), la réduction de la matrice de marge profit/perte (PLM), les nœuds de disposition, le groupement des opérations de désassemblage, la formulation du problème, des objectifs, des variables de décisions et des contraintes. Les auteurs présentent une analyse de coûts pour l'évaluation des opportunités de récupération en fin de vie des produits. Ils apportent de nouveaux éléments par rapport à Johnson et Wang [20], sur l'arbre de désassemblage, la matrice de profit et sur l'algorithme d'optimisation des séquences de désassemblage.

Xu, Huang, Ling, Hu, Li, Hu [54] présentent une méthode pour désassembler un produit : le réseau logique de désassemblage. Elle utilise un dérivé des arbres de désassemblage pour illustrer le produit. C'est une méthode intéressante par sa facilité d'utilisation. Elle peut être utilisée pour effectuer de la maintenance et de la récupération de composants. L'algorithme de Dijkstra lui ressemble beaucoup par le fait qu'elle trouve le chemin le plus court en deux points. La présentation des résultats est simple à comprendre. Cette méthode est donc utilisable dans le cas de désassemblage du produit relativement simple et ne comprenant pas de relation trop complexe. La plupart des méthodes graphiques et géométriques permettent de réaliser le désassemblage de produit avec relation complexe. Donc pour trouver le coût minimum de désassemblage rapidement avec des produits simple, le réseau logique de désassemblage est intéressant.

Seo, Park, et Jang [40] présentent une approche basée sur l'algorithme génétique pour trouver la séquence de désassemblage optimale en considérant les aspects environnementaux et économiques. Les séquences de désassemblage faisables sont générées à partir de l'arbre de désassemblage (section 4.2.2). Ils utilisent les matrices de désassemblage (section 4.2.1) et les matrices avec valeurs économiques pour optimiser la séquence. Ils considèrent l'aspect économique et environnemental dans le contexte de récupération de produit, principalement pour le recyclage. Les facteurs économiques et

environnementaux peuvent être comparés par les mêmes mesures, donc les facteurs environnementaux sont convertis en coût. Par l'algorithme génétique, nous pouvons analyser dynamiquement le désassemblage qui nous permet d'avoir la meilleure valeur.

7.1 Évaluation économique des opérations de désassemblage

Cette section mise sur l'intégration de la conception de produit et les procédures de désassemblage pour faciliter une récupération économique des matériaux. Trois axes de corrélation du désassemblage et de la récupération des matériaux sont visés : (1) l'analyse économique ; (2) la génération des séquences de désassemblage ; (3) la conception pour le désassemblage. Le but principal de cette partie est de générer une séquence optimale de désassemblage en terme de considération économique des opportunités de récupération des matériaux à la fin du cycle de vie.

Trois critères pour réduire l'espace de recherche et pour faciliter les opportunités de récupération sont les suivantes :

1. La compatibilité des matériaux
2. Le regroupement pour l'évaluation
3. Les opérations courantes de désassemblage

7.1.1 Partie analyse environnementale

Les détails de l'optimisation des coûts sont présentés dans deux articles de Johnson et Wang [20, 21]. Les concepteurs peuvent être intéressés d'utiliser la conception pour le désassemblage (DFD) pour différentes fins de vie de vieux matériaux. (1) recyclage, (2) récupération et (3) réutilisation. Les éléments pertinents des pages suivantes sont, les quatre niveaux d'analyse du désassemblage, la valeur du matériel, les facteurs de coût de

la valeur des matériaux, l'analyse économique des séquences de désassemblage, et les nœuds de disposition.

Formulation du problème

- Cycle de vie environnementale : extraction de la matière première, transformation, transport, fabrication, utilisation et finalement le rejet.
- Les opportunités de récupération des matériaux (MRO) ne font pas juste réduire le rejet de matériaux, elles réduisent les besoins en matériaux neufs ce qui diminue la pollution potentielle due à l'extraction et à la transformation.
- Le désassemblage de produit pour opportunités de récupération des matériaux demandent un temps de désassemblage et un coût à minimiser et la valeur des pièces résultantes maximisées.

Analyse du désassemblage des produits

- Trois chemins pour analyser le désassemblage
- Une analyse basée sur les composants (visant un composant en particulier ou un groupe de composants d'un produit)
- Une analyse basée sur les matériaux (visant un type particulier de matériaux ou un groupe de types de matériaux d'un produit)
- Une analyse basée sur le produit (visant le produit en entier incluant tous les matériaux et composants)

La figure 7.1 illustre les procédures de désassemblage et la maximisation potentielle de l'investissement pour les opportunités de récupération des matériaux. Deux facteurs sont associés avec la maximisation des profits dans les procédures de désassemblage :

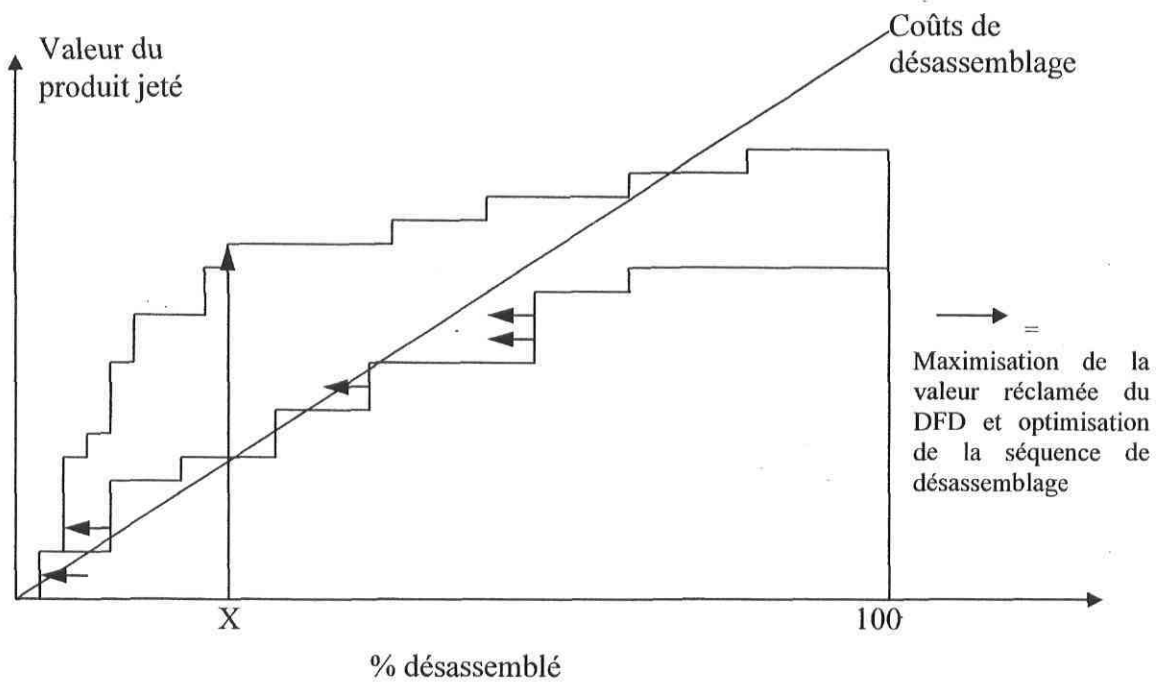


Figure 7.1 : Maximisation potentielle pour le MRO

Source : Johnson et Wang [21]

1. L'importance économique de concevoir des items de haute valeur (ayant une valeur résiduelle importante) qui sont enlevés en premier.
2. Les procédures de désassemblage sont optimisées, car nous devons trouver les séquences de désassemblage pour retrouver la valeur des composants dans un court délai de temps.

Nomenclature

Rv_k	valeur du composant k (\$)
mv_k	valeur du matériel du composant k (\$/unité de poids)
wt_k	poids du k^e composant
d_f	facteur de dépréciation entre 0 et 1
Cd_k	coût de désassemblage du k^e composant (\$)
Cd_T	coût total de désassemblage pour tous les composants de désassemblage (\$)
t_k	temps de désassemblage du k^e composant
C_L	ratio de travail (\$/unité de temps)
Cp_k	coût de rejet du k^e composant (\$)
Cp_R	ratio courant de rejet (\$/unité de poids)

C_{pT}	coût total de rejet pour m composants (\$)
PLM_T	marge profit/perte totale des réclamations de rejet (\$)
PLM_{MAX}	point dans la procédure de désassemblage où le PLM_T est maximisé (\$)
$PLM_{RECOVERY}$	un index de décision pour la récupération : marge profit/perte pour le composant k réclamé (\$)
$PLM_{DISPOSAL}$	un index de décision pour le rejet : marge profit/perte pour disposer du composant k (\$)
PLM_{FINAL}	le PLM résultant, généré par l'analyse du coût (\$), utilisant comme entrée dans la génération des séquences de désassemblage
n	Nombre total de composants réclamés.
m	Nombre total de composants à l'intérieur du produit
$d1$	Nombre total de composants désassemblés mais rejetés
$d2$	Nombre total de composants rejetés (sans désassemblage)

Hypothèses

1. La méthode d'analyse des coûts est plus appropriée pour inclure l'économie associée avec le désassemblage des produits pour la réclamation des matériaux. Deux groupes peuvent être intéressés par cette analyse (a) les recycleurs (b) les fabricants ;
2. En travaillant avec les composants de recyclage, certains paramètres à calculer qui sont identifiés, ont un grand impact sur les coûts et les bénéfices associés avec la récupération des matériaux ;
3. Pour calculer les paramètres, il faut s'assurer que certaines bases de données soient disponibles.

Première analyse du produit à désassembler

1. Les différentes opérations de désassemblage et les composants enlevés durant cette opération.
2. Les temps de désassemblage associé avec les opérations.
3. Le poids des composantes récupérées.
4. Pour le désassemblage manuel, il est important d'établir une courbe d'apprentissage et les compétences demandées dans les procédures de désassemblage.

5. Les variétés de produits sur la ligne de désassemblage demande du temps de tri. Il est important de diminuer la variété.
6. Une analyse est très souhaitable pour automatiser le désassemblage en utilisant des robots.

Quatre niveaux d'analyse du désassemblage

Niveau 1 : Étude de faisabilité des opportunités de récupération des matériaux. Dans ce niveau, une étude initiale est nécessaire pour déterminer :

1. Pourcentage de produit que nous pouvons récupérer ;
2. Estimation des coûts préliminaires : la récupération versus la disposition ;
3. Autres considérations d'échange : courbe d'apprentissage, automatisation, réponse du consommateur, les tendances de la législation ;
4. Analyse de décision : toutes les procédures quantitatives et qualitatives pour déterminer tous critères de faisabilité possibles des opportunités de récupération des matériaux.

Niveau 2 : Génération de la séquence optimale de désassemblage. Ce niveau génère une séquence de désassemblage préférable qui va maximiser les valeurs qui sont demandées dans la récupération. La séquence de désassemblage est analysée en considérant les caractéristiques courantes de conception.

1. Relations de priorités entre les composants ;
2. Caractéristiques des opérations individuelles de désassemblage tel l'outillage et l'accessibilité et comment telles caractéristiques se chevauchent entre les opérations ;
3. Groupement des matériaux compatibles ;
4. Maximisation des séquences en terme d'opérations de désassemblage courantes et les possibilités de récupération des matériaux.

Niveau 3 : Optimisation du désassemblage. Utilisation des séquences de désassemblage générées au niveau 2, le désassemblage est donc optimisé en terme de coût, bénéfique et en degré de désassemblage. De cette analyse, résulte l'information suivante :

2. Identification des composants qui adhèrent aux opportunités de récupération des matériaux et qui n'adhèrent pas ;
3. Identification si chaque point de la procédure de désassemblage est économique pour continuer la récupération ;
4. Identification des composants de conception spécifique qui contraignent la faisabilité économique de récupération des matériaux ;
5. Identification des composants qui vont apporter des bénéfices de la conception pour le désassemblage.

Niveau 4 : Guide la conception pour le désassemblage, ce niveau prend en considération tout ce qui a été déterminé dans les niveaux pour implanter les opportunités de récupération des matériaux dans les entreprises.

Valeur du matériel

Un avantage de cette perspective est une analyse quantitative pouvant être utilisée pour comparer le coût des matériaux vierges des manufacturiers avec les coûts pour l'amélioration du niveau de qualité des matériaux vierges.

Dans toutes les situations, la valeur des matériaux peut varier dépendant des opportunités de récupération des matériaux employées, le niveau de rejet des produits, le coût de rejet et autres coûts externes associés avec la transformation des matériaux.

Facteurs de coût de la valeur des matériaux

1. Total développement/amélioration des coûts incluant le transport, tri, nettoyage, désassemblage et procédures. Parfois la conception des produits peut rendre un composant de partie non récupérable ;
2. Produits/composants stade de rejet : la forme des produits rejetés doit être analysée ;
3. Possibilité de recyclage et transport des infrastructures ;
4. Prix de disposition.

Tous ces facteurs sont situés dans le niveau 1 d'analyse et peuvent être subséquemment utilisés pour l'analyse les valeurs de post-consommation des produits

D'autres considérations peuvent aussi être faites dans l'analyse de l'évaluation des coûts en faveurs ou non en faveurs d'une augmentation de la valeur des matériaux.

Analyse économique des séquences de désassemblage

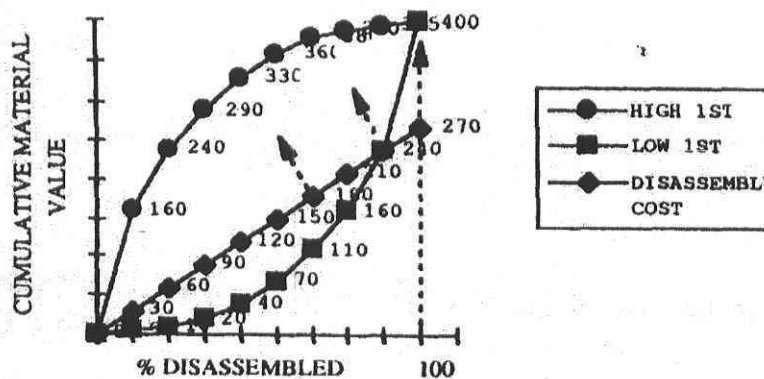


Figure 7.2 : Importance des séquences de désassemblage

Source : Johnson et Wang [20]

Le produit désassemblé dans la figure 7.2 présente les intervalles des matériaux de valeurs positives. Les deux courbes démontrent que le désassemblage est contraint par

l'arrangement des composants et peuvent atteindre l'objectif de désassemblage complet du produit au travers d'un large écart dans les séquences de désassemblage.

En fait, l'écart entre les deux courbes représente l'écart de faisabilité des séquences de désassemblage. La courbe supérieure est obtenue si les matériaux sont enlevés au travers des étapes successives des matériaux de plus haute valeur vers les matériaux de plus faible valeur. La courbe du bas est obtenue si les matériaux sont enlevés des valeurs les plus bas vers le plus haut. La ligne du milieu représente la relativité linéaire d'un ratio de travail (\$/h) et temps.

Dans ce mode, les règles du désassemblage peuvent être générées en analysant le point où la procédure de désassemblage doit arrêter pour minimiser le coût et maximiser le retour de réclamation de matériaux :

Règle 1 : Si la marge de perte (PLM_T) > 0 , on continue le désassemblage.

Règle 2 : Si la marge de perte (PLM_T) ≤ 0 , arrêt rejet à considérer.

Un composant spécifique k est défini comme ayant une valeur de réclamation (Rv_k) :

$$Rv_k = mv_k \cdot wt_k \cdot df$$

Où mv_k représente la valeur des matériaux du k^e composant et df est le facteur de dépréciation spécifique après utilisation étant situé entre 0 et 1. wt_k est le poids.

Le coût de désassemblage associé à l'enlèvement du k^e composant peut être représenté par :

$$Cd_k = t_k \cdot C_L$$

Où t_k est le temps de désassemblage pour le k^e composant et C_L est considéré comme le ratio de travail.

Le coût de disposition avec enfouissement associé au k^e composant peut être représenté par :

$$Cp_k = Cp_R \cdot wt_k$$

Où Cp_R est le ratio de coût et wt_k est le poids du k^e composant.

En rassemblant les paramètres retrouvés précédemment, la marge profit/perte (PLM) de récupération du k^e composant pour la récupération du matériel, peut être exprimée par :

$$PLM_{RECOVERY,k} = Rv_k - Cd_k + Cp_k \quad (k = 1, 2, \dots, n)$$

Où Cp_k est le coût sauvé en ne payant pas les coûts de disposition du k^e composant. Toutes les valeurs sont positives pour tous les m composants.

La marge profit/perte pour disposer du k^e composant peut être exprimée par l'une des deux expressions suivantes :

$$PLM_{DISPOSAL,k} = \begin{cases} -Cp_k & \text{coût de disposition seulement } (k = 1, 2, \dots, d_1) \\ -Cp_k - Cd_k & \text{coût de disposition et coût de désassemblage } (k = 1, 2, \dots, d_2) \end{cases}$$

- La première équation représente le coût de disposition du k^e composant.
- La deuxième équation représente le coût de disposition plus le coût de désassemblage du k^e composant.

L'analyse économique précédente nous permet de construire le tableau suivant qui représente les valeurs obtenues par cette analyse et la prise de décision pour la récupération et le rejet (disposition).

Tableau 7.1 : Table de décision pour la récupération et le rejet [21]

Opérations de désassemblage	Composant enlevé	Rv	Cd	Cp	PLM	PLM	PLM	Décision	PLM
					récupération	D1	D2		final
1	1A	0	10	0	-10	0	-10	D2	-10
2	Sub1	-							
	Sub2	-							
	Sub3	-							
	Sub4	-							
3(Sub1)	2A	1	7	2	-4	-2	-9	Récupération*/	-4
	3A	0	9	0	-9	0	-9	D1	0
4(Sub2)	3B	1	-	0	1	0	0	D1	0
	4A	0	6	0	-6	0	-6	D2 *	-6
5	5A	29	8	0	21	0	-8	Récupération	23
	5B	2	-	0	2	0	0	Récupération*/	-
6(Sub3)	6A	2	5	5	2	-5	-10	Récupération /	12
	6B	10	-	0	10	0	0	Récupération	-
7(Sub4)	7A	34	20	0	14	0	-20	Récupération	14
8	8A	11	7	3	7	-3	-10	Récupération	2
	8B	0	-	5	5	-5	-5	Rejet /	-
Somme de PLM									31

* Désassemblage requis pour avoir accès aux autres matériaux, marge perte ou gain : où l'on considère les autres facteurs de coûts : prix des matériaux vierges, coûts de traitement.

Source : Johnson et Wang [21]

7.1.2 Partie optimisation de la séquence de désassemblage

Matrices et arbre de désassemblage avec coûts

Étapes de construction des matrices :

1. Formation de matrice PLM : la matrice de profit initial est formée ;
2. Réduction de la matrice PLM : critère pour rechercher des espaces de réduction ;
3. Entrée la matrice PLM dans la formulation du réseau de deux commodités.

Étapes 1 : Formation de la matrice PLM

La génération des séquences de désassemblage peut être utilisée dans l'estimation de procédures de désassemblage pour, simultanément, minimiser les coûts et maximiser la

valeur de matériaux réclamés. Nous utilisons les valeurs de PLM_{FINAL} qui sont générées dans l'analyse économique, pour construire la matrice de profit. Par cette matrice nous analysons les séquences dépendantes entre l'économie et les opérations de désassemblage.

Les valeurs PLM_{ij} représentent la marge profit/perte du passage entre l'opération de désassemblage i et l'opération de désassemblage j . Ces valeurs de passage entre les opérations de désassemblages représentent les conséquences économiques pour déplacer le composant suivant dans la procédure de désassemblage.

Initialement la procédure de départ est de former la matrice de profit en considérant les valeurs de passage des séquences dépendantes.

$$D = [PLM_{ij}] = \begin{matrix} & \begin{matrix} \text{Opérations de désassemblage} \\ 3 & 4 & 5 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} \text{Opérations de désassemblage} \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{matrix} & \begin{bmatrix} - & -6 & 23 & 12 & 14 & 2 \\ 0 & - & 23 & 12 & 14 & 2 \\ 0 & -6 & - & 12 & 14 & 2 \\ 0 & -6 & 23 & - & 14 & 2 \\ 0 & -6 & 23 & 12 & - & 2 \\ 0 & -6 & 23 & 12 & 14 & - \end{bmatrix} \end{matrix}$$

Étape 2 : Réduction de la matrice PLM

Pour réduire l'espace de recherche nous utilisons les critères suivants :

- Regrouper les matériaux compatibles ;
- Regrouper les nœuds de PLM_i négatifs pour disposition ;
- Regrouper les étapes de désassemblage similaires.

Étape 3 : Construction de l'arbre de désassemblage

1. Initialement, une représentation du produit désassemblé est illustrée graphiquement par un arbre de désassemblage (DT), c'est la définition des éléments.
2. L'arbre est pleinement étendu incluant les opérations de désassemblage interchangeables.

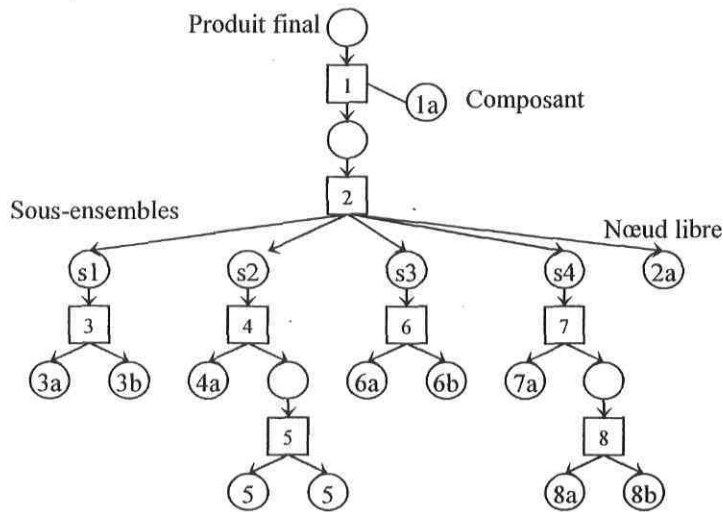


Figure 7.3 : Représentation de l'arbre de désassemblage

Source : Johnson et Wang [21]

3. La dimension de l'espace de recherche est réduite en utilisant les quatre critères suivants :
 - Regroupement des matériaux compatibles ;
 - Regroupement des PLM_i négatifs, nœuds pour rejet ;
 - Regroupement des opérations de désassemblage similaires ;
 - Maximisation du rendement.
4. Une fois l'arbre réduit, le modèle économique PLM est utilisé pour analyser les opérations de désassemblage ou les nœuds individuellement.

Compatibilité des matériaux

En terme relatif, nous diminuons le nombre de composants avec un regroupement des matériaux compatibles, et ainsi réduire le nombre d'opérations de désassemblage requises cela permet d'améliorer deux points :

1. Le temps de désassemblage est diminué pour une réduction de coût ;
2. Un grand volume de matériaux est regroupé pour le recyclage.

Évaluation de séquence

$$m_c = \sum_{i=1}^k R v_i + C p_i$$

- La valeur de m_c est ajoutée aux opérations de désassemblage « parentes »
- Élimine des opérations de désassemblage à l'intérieur des groupes

Amélioration MRO :

- Promouvoir le recyclage
- Diminuer le temps de désassemblage

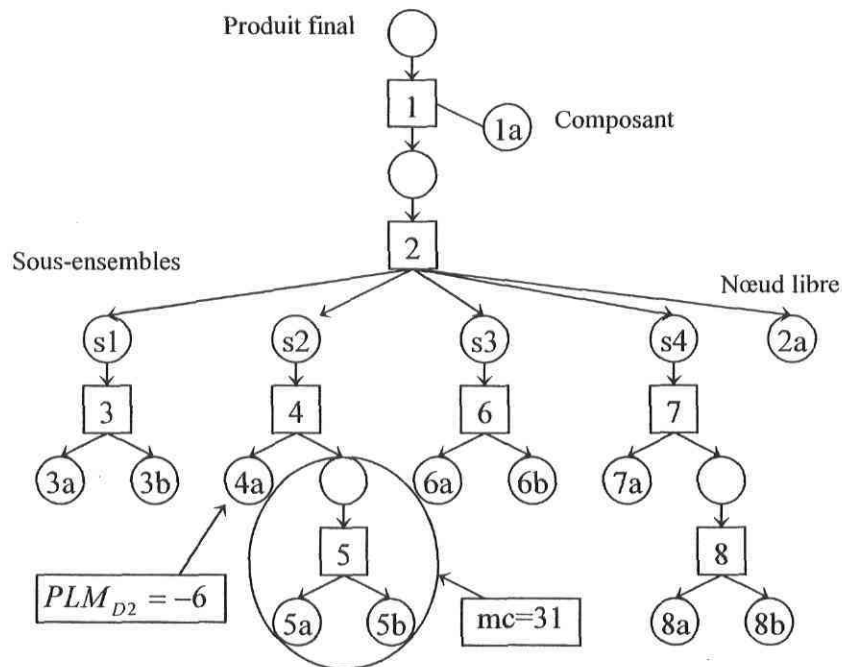


Figure 7.4 : Groupe pour la compatibilité des matériaux

Source : Johnson et Wang [21]

Le bénéfice associé avec le regroupement dans un ensemble de matériaux, ajouté aux opérations de désassemblage précédentes sont définies par la fonction :

$$\Delta PLM_{PARENT} = \begin{cases} m_c = \sum_{i=1}^k Rv_i + Cp_i & \text{si matériaux(parent) = matériaux(enfant)} \\ 0 & \text{si matériaux(parent) \neq matériaux(enfant)} \end{cases}$$

Où m_c est la valeur de réclamation totale des groupes de matériaux et k est le nombre total de composants à l'intérieur des groupes, i représente les composants individuels et leurs valeurs Rv et Cp correspondantes aux matériaux parents qui sont les types de matériaux de nœuds parents et aux matériaux enfants qui sont les types de matériaux de nœuds enfants.

Nœuds parents et enfants de matériaux compatibles et de groupement de valeurs PLM sont ajoutés aux nœuds périphériques. Les opérations de désassemblage et leurs coûts correspondants entre les matériaux compatibles sont éliminées de l'espace de recherche. Les nœuds parents et enfants qui ne sont pas des matériaux compatibles et la valeur sans extra sont ajoutées aux nœuds parents.

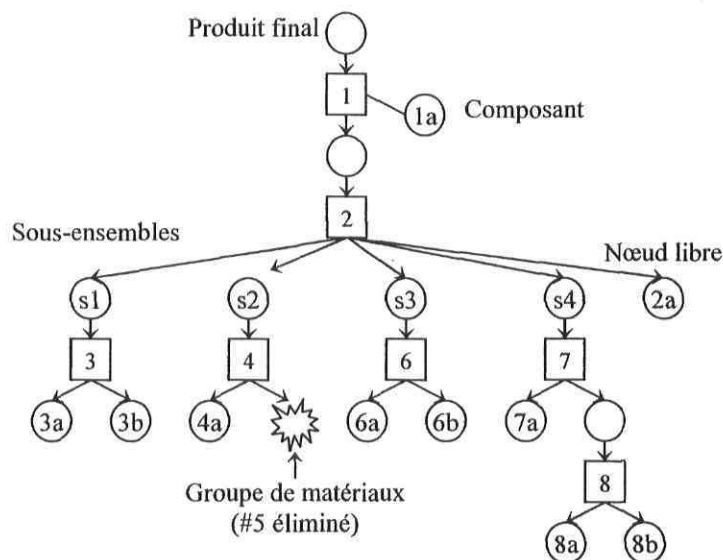


Figure 7.5 : Arbre réduit de désassemblage

Source : Johnson et Wang [21]

$$D = [PLM_{ij}] = \text{Opérations de désassemblage}$$

		<i>Opérations de désassemblage</i>				
		3	4	6	7	8
3	-	25	12	14	2	
4	0	-	12	14	2	
6	0	25	-	14	2	
7	0	25	12	-	2	
8	0	25	12	14	-	

Nœuds de disposition

Dans certaines circonstances, les concepteurs vont trouver que ce n'est pas économique de considérer le recyclage et le réusinage ou la réutilisation de certains composants ou matériaux et que le rejet est la meilleure alternative. Parfois, la meilleure solution est de grouper les matériaux qui sont destinés au rejet dans un nœud de disposition pour éliminer tous les coûts de désassemblage futur. Les groupes de matériaux compatibles similaires et les opérations de désassemblage à l'intérieur des nœuds de disposition sont éliminés de l'espace de recherche pour réduire la complexité future de la recherche. En utilisant l'exemple précédent, l'opération de désassemblage #3 peut faire encourir d'autres coûts seulement si elle est effectuée pour la récupération. Donc, elle devient un nœud de disposition et est enlevée de la matrice PLM :

$$D = [PLM_{ij}] = \text{Opérations de désassemblage}$$

		<i>Opérations de désassemblage</i>			
		4	6	7	8
4	-	12	14	2	
6	25	-	14	2	
7	25	12	-	2	
8	25	12	14	-	

Les règles suivantes sont utilisées pour l'analyse si le désassemblage futur avec un arc du DT doit continuer ou non.

Si , $PLM_i \geq 0$, alors

l'opération de désassemblage i est incluse dans les séquences de désassemblage.

Si $PLMi < 0$, alors

n est le nombre total de tâche de désassemblage enlevée de l'arc,

Si $n = 1$, alors

Arrêt de désassemblage et enlèvement de l'opération de désassemblage de l'espace de recherche. Retour à l'utilisateur : plus économique de disposer du composant qui va être désassemblé seulement si cela exige un coût élevé.

Si $n > 1$, alors

PLM_{\max} est la k^e opération qui a le maximum de PLM pour toutes les tâches n

Calculer PLM_{\max}

Si $PLM_{\max} > 0$, alors

Les k premières opérations de désassemblage sont incluses dans l'espace de recherche. Toutes $(n - k)$ opérations de désassemblage dans l'espace de recherche doivent être regroupées pour disposition.

Si $PLM_{\max} \leq 0$, alors

Arrêt du désassemblage et enlèvement de toutes les n opérations de désassemblage de l'espace de recherche.

Groupe d'opérations de désassemblage

La même analyse que celle utilisée dans la compatibilité des matériaux peut être appliquée dans le groupement des opérations de désassemblage. Le fait de sauver en regroupant les opérations similaires est défini par la fonction suivante :

$$\Delta PLM_{ij} = \begin{cases} do_r = t_s L & \text{si l'opération } i = \text{l'opération } j \\ 0 & \text{si l'opération } i \neq \text{l'opération } j \end{cases}$$

où ΔPLM_{ij} est le changement de la valeur PLM entre l'opération i et j, do_r vise la j^e opération de l'augmentation de la valeur PLM, l'opération i est l'opération de désassemblage du i^e nœud et l'opération j est l'opération de désassemblage du j^e nœud. Nous sommes en présence d'une réduction du coût de désassemblage parce que la réduction du temps de désassemblage correspond au temps de travail (L).

Pour bien illustrer cette utilisation, nous utilisons l'exemple précédant. Si les opérations de désassemblage 4, 6, et 7 demandent les mêmes outils nous obtenons des économies respectivement de 2, 3, et 9, cette économie étant appliquée à la matrice PLM :

do_r	Séquence	ΔPLM
$do_4 = 2$	6-4	+2
	7-4	+2
$do_6 = 3$	4-6	+3
	7-6	+3
$do_7 = 9$	4-7	+9
	6-7	+9

$$D = [PLM_{ij}] = \begin{matrix} & \text{Opérations de désassemblage} \\ & \begin{matrix} 4 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} 4 \\ 6 \\ 7 \\ 8 \end{matrix} & \begin{bmatrix} - & 15 & 23 & 2 \\ 27 & - & 23 & 2 \\ 27 & 15 & - & 2 \\ 25 & 12 & 14 & - \end{bmatrix} \end{matrix}$$

Formulation du modèle

N est le nombre d'opérations de désassemblage sur une machine ou par un travailleur. Les citations 1,2,..., n correspondent aux opérations de désassemblage Q_1, Q_2, \dots, Q_n qui doivent être planifiées. Les distances entre les citations i et j sont remplacées pour la séquence dépendante des valeurs PLM_{ij} qui fructus entre les opérations i et j. Nous utilisons le modèle réseau, le problème de génération d'une séquence optimale de désassemblage est formulé en terme de maximisation des séquences dépendantes PLM valeurs entre les opérations de désassemblage.

Objectifs

L'objectif primaire est de maximiser le profit associé avec le désassemblage du produit. Le problème de génération d'une séquence optimale de désassemblage est formulé en terme de maximisation de la valeur de la marge profit/perte des séquences dépendantes entre les opérations de désassemblage :

$$\text{Max } Z = \sum_i \sum_j PLM_{ij}(x_{ij}^P + x_{ij}^Q)$$

où

PLM_{ij} représente la marge profit/perte du passage entre l'opération de désassemblage i et j ;

x_{ij}^P représente le flux du réseau du produit P ;

x_{ij}^Q représente le flux du réseau du produit Q .

Variables de décisions

Les variables de décisions sont les séquences d'opérations de désassemblage. La notation x_{ij} dénote la i^{e} et la j^{e} opération de désassemblage sous considération que $x_{ij} \in (0,1)$. Quand $x_{ij} = 1$, l'opération i précède immédiatement l'opération j . Alternativement quand $x_{ij} = 0$, l'opération i ne précède pas l'opération j .

Variables auxiliaires

Le réseau est représenté par la notation $G = (N, A)$ où N est l'ensemble des nœuds et A l'ensemble des arcs du réseau G . Le réseau G est trouvé avec les profits PLM_{ij} des arcs $(i, j) \in A$. De même, $n = |N|$ nœuds qui sont comptés $1, 2, \dots, n$ et $m = |A|$ arcs.

La formulation des réseaux de deux produits est définie comme suit :

- Un flux faisable x_{ij}^P est requis avec égard du produit P. Nous sélectionnons un point de départ s, (n-1) unités de P sont disponibles, un pour chaque nœud, $i \neq s$.

$$\sum_{j=1}^n x_{ij}^P - \sum_{j=1}^n x_{ji}^P = n-1 \quad \text{pour } i = s$$

Pour chaque nœud, une demande d'au moins une unité du produit P est requise :

$$\sum_{j=1}^n x_{ij}^P - \sum_{j=1}^n x_{ji}^P = -1 \quad \text{Même chose pour tous les autres } i \in N$$

$$x_{ij}^P \geq 0 \quad \text{pour tous } (i, j) \in A$$

- De même, un flux réalisable x_{ij}^Q est requis avec égard du produit Q. En complémentarité P,Q demande $-(n-1)$ unités sélectionnées au point de départ s :

$$\sum_{j=1}^n x_{ij}^Q - \sum_{j=1}^n x_{ji}^Q = -(n-1) \quad \text{pour } i = s$$

Pour chaque nœud, une demande d'au moins une unité du produit Q est requise :

$$\sum_{j=1}^n x_{ij}^Q - \sum_{j=1}^n x_{ji}^Q = +1 \quad \text{même chose pour tous les autres } i \in N$$

$$x_{ij}^Q \geq 0 \quad \text{pour tous } (i, j) \in A$$

- Le flux combiné total $x_{ij}^P + x_{ij}^Q$ pour chaque arc est égal à n-1 pour chaque arc du tour :

$$\sum_{j=1}^n (x_{ij}^P + x_{ij}^Q) = n-1 \quad \text{pour chaque } i \in N$$

- De plus la condition intégrale $x_{ij}^P + x_{ij}^Q = 1$ ou $(n-1)$ doit être imposée pour tous les arcs du tour. Ainsi, les quatre contraintes déterminent un lien de formulation du réseau des deux produits de la variable de décision x_{ij} :

$$x_{ij}^P + x_{ij}^Q = (n-1)x_{ij} \quad \text{pour tous } (i, j) \in A$$

Les contraintes de priorités sont sous la forme d'opération de désassemblage O_u avant O_v ou $O_\mu \rightarrow O_\nu$. Ceci est représenté par :

$$\sum_{j=1}^n x_{\mu j}^P - \sum_{j=1}^n x_{j\nu}^P \geq 1 \quad \text{pour tous } O_\mu \rightarrow O_\nu$$

En utilisant l'exemple précédent et en supposant que la matrice D a été réduite et évaluée aussi loin que possible en utilisant les critères précédemment définis, une opération fictive est introduite en utilisant un vecteur ligne et colonne comme indiqué dans la matrice suivante. Cette opération fictive indique l'étape initial et final des opérations de désassemblage.

$$D = [PLMij] = \begin{matrix} & \text{Opérations de désassemblage} \\ & \begin{matrix} 0 & 4 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} 0 \\ 4 \\ 6 \\ 7 \\ 8 \end{matrix} & \begin{bmatrix} - & 25 & 12 & 14 & 2 \\ 0 & - & 15 & 23 & 2 \\ 0 & 27 & - & 23 & 2 \\ 0 & 27 & 15 & - & 2 \\ 0 & 25 & 12 & 14 & - \end{bmatrix} \end{matrix}$$

Nous résolvons le problème en se basant sur les contraintes ci-dessus et la matrice réduite D et trouvons la solution suivante : $x_{04} = 1, x_{47} = 1, x_{76} = 1, x_{68} = 1, x_{80} = 1$ pouvant être exprimé par la séquence de désassemblage : (0, 4, 7, 6, 8, 0).

7.2 Réseau logique de désassemblage

Plusieurs autres méthodes que celle de Johnson et Wang [20, 21] existent pour évaluer directement le coût de désassemblage minimum. Xu, Huang, Ling, Hu, Li, Hu [54] présentent une méthode récente et innovatrice, celle du réseau logique de désassemblage. Cette dernière fait l'analyse de la structure du réseau de désassemblage.

Terminologie pour cette méthode

Point de désassemblage : Durant la procédure de désassemblage du produit, les composants à être désassembler sont nommés, points de désassemblage.

Meta-problème : Un point de désassemblage et les activités de désassemblage composant une procédure de solution complète dans le problème de désassemblage.

Point initial et terminal : Point initial quand la série devant le composant est inexistant (premier composant) et point terminal quand la série derrière le composant est inexistante (dernier composant).

Hypothèses

1. Le réseau logique de désassemblage (D-L-N) peut avoir de multiple points initiaux et terminaux de désassemblage ;
2. Chaque point de désassemblage peut être enlevé seulement une fois durant la procédure de désassemblage, il ne peut y avoir aucun retour ou boucle ;
3. Entre deux points de désassemblage il doit exister une relation de priorité ;
4. Les relations logiques ET/OU entre les meta-problèmes peuvent exister seulement entre celles ayant le même point terminal.

Caractéristiques du D-L-N

1. Le meta-problème dans le D-L-N a un sous-problème concept. Un problème de désassemblage spécial peut consister quelques fois en un meta-problème.
2. La solution d'un problème de désassemblage peut-être sérielle ou simultanée.
3. Le D-L-N peut être exprimé comme une relation logique avec contrainte. En ce basant sur cela, la planification du désassemblage peut-être exprimé par un schéma multiple.
4. Il n'existe pas de concept d'activité virtuelle dans la méthode Activité sur arc (A-O-A), parce que O et \rightarrow sont combinés dans le meta-problème de la méthode D-L-N. Le meta-problème utilise le concept du nœud de la méthode A-O-A, si nous pouvons considérer comme un nœud avec arc dirigé.

Modèle optimal pour coût minimum de désassemblage

Par la construction correcte du D-L-N, le coût de désassemblage de chaque meta-problème est fixé. Pour différents problèmes de désassemblage qui comprennent la combinaison sérielle ou parallèle de quelques meta-problèmes, il est nécessaire de calculer le paramètre du coût de chaque meta-problème. En premier, nous définissons plusieurs paramètres de coût de base où le coût complet de désassemblage est présenté.

Coût d'activité, ($C_i \rightarrow j$) coût complet dans une activité de désassemblage

Coût du meta-problème, (C_i) coût total des activités dans un meta-problème

$$C_i = \sum_{j=1}^n C_{i \rightarrow j}$$

Où : n détermine la quantité d'activité de E_i .

Le coût d'activité immédiatement devant (C_i) coût total de toutes les activités de relations ET qui caractérise le meta-problème.

$$C_i = \sum_{k \in A} C_{k \rightarrow i}$$

Où, $A = \{k | k \in \cdot i, \text{ et chaque } e_{ki} \text{ constituant les relations ET}\}$

En raison de la relation OU, le coût d'activité immédiatement devant du meta-problème n'est pas toujours unique. Pour un réseau ET, c'est un coût d'activité immédiatement devant qui peut être écrit directement comme :

$$C_i = \sum_{k \in \cdot i} C_{k \rightarrow i}$$

Le coût compréhensif de désassemblage (\hat{C}_i), pour le problème de désassemblage d'un point de désassemblage i , débute à un ou quelques points initiaux de désassemblage dans la direction des activités de désassemblage. Nous entreprenons le travail de désassemblage entre chaque point de désassemblage, jusqu'à ce que le désassemblage i soit complété. Le coût total d'activité requis pour la procédure globale est comme suit :

$$\hat{C}_i = \sum_{k \in A} \hat{C}_k + C_i - C_{k \rightarrow j}$$

Où, $A = \{k | k \in \cdot i, \text{ et chaque } e_{ki} \text{ constitue des relations ET}\}$, $j \neq i$. Similairement, pour un réseau ET, le coût compréhensif de désassemblage peut être écrit comme :

$$\hat{C}_i = \sum_{k \in \cdot i} \hat{C}_k + C_i - C_{k \rightarrow j}$$

Solution du coût minimum de désassemblage

En raison des exigences de la maintenance et du recyclage, un composant ou quelques composants dans le produit doivent être désassemblés et le coût doit être réduit au minimum. Le réseau logique provient des méthodes de modélisations pour ce problème pratique. Il peut être divisé en quelques réseaux ET, qui correspondent aux quelques plans de désassemblage faisables, le plan de désassemblage avec le coût minimum étant le plan de désassemblage final.

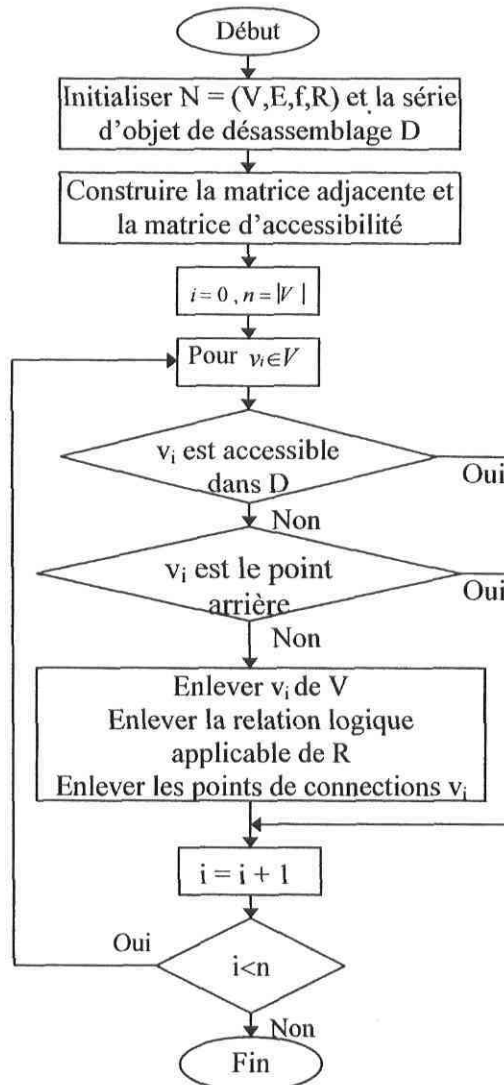


Figure 7.6 : *Algorithme pour le réseau incident directeur positif*

Source : Xu, Huang, Ling, Hu, Li, Hu [54]

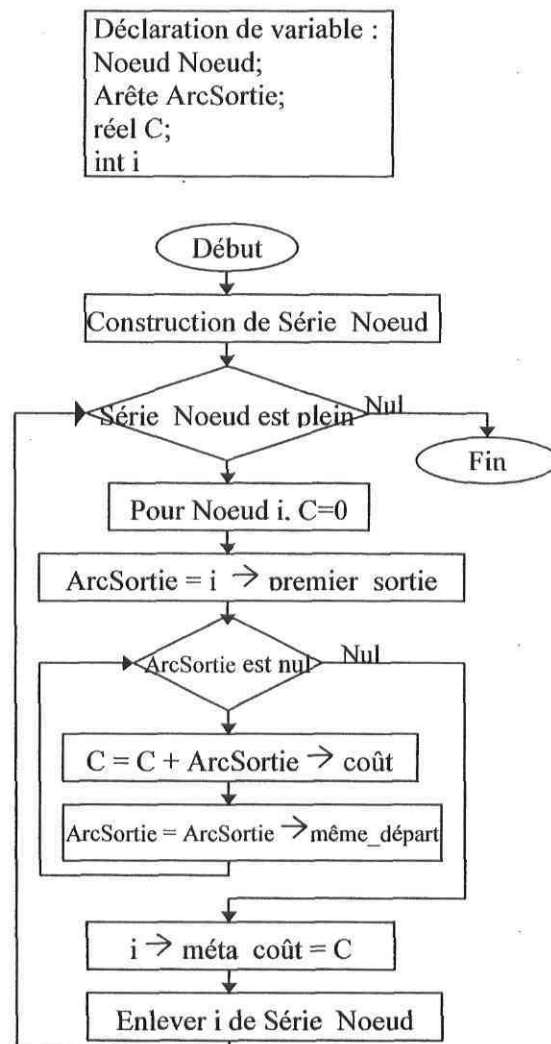


Figure 7.7 : *Algorithme pour le coût du méta_problème d'un noeud*

Source : Xu, Huang, Ling, Hu, Li, Hu [54]

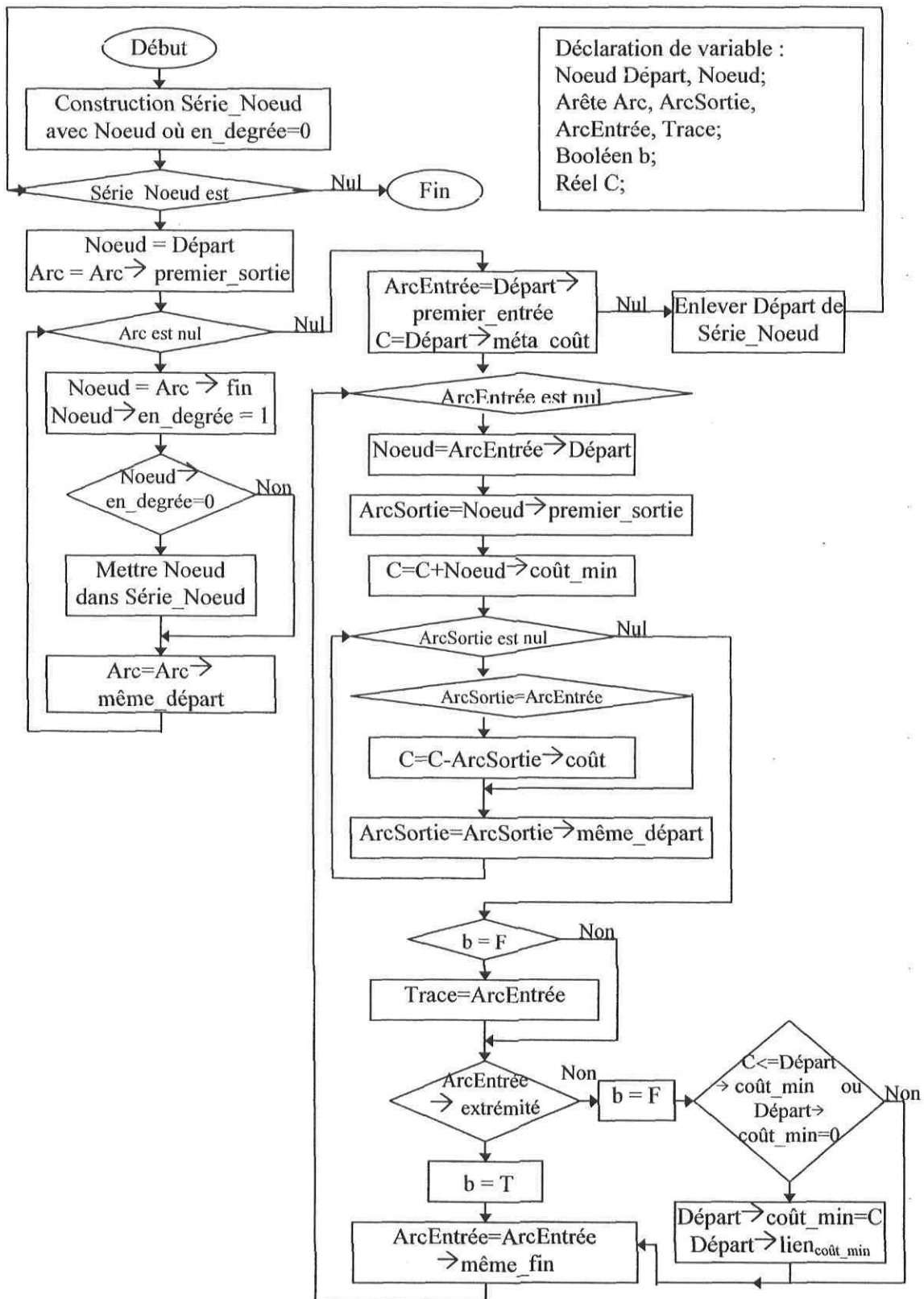


Figure 7.8 : Algorithme directeur positif pour le coût minimum d'un nœud

Source : Xu, Huang, Ling, Hu, Li, Hu [54]

Structure du magasin de D-L-N

Les éléments de D-L-N incluent les nœuds, arcs et les relations ET/OU entre les arcs. La liste orthogonale est un outil idéal pour décrire ces éléments.

Pour tout nœud $v_i \in V$, la structure des données est exprimée dans le tableau 7.2.

« *num_série* » est le nombre de série. « *en_degrée* » est le nombre d'arêtes qui finissent à v_i . « *premier_entrée* », « *premier_sortie* » sont deux liens qui pointent le premier « *bord-entrée* » et le dernier « *bord-sortie* » respectivement. Pour $v_i \in V_B$, « *premier_entrée* » doit être nul, pendant que pour $v_i \in V_E$, le « *premier_sortie* » doit être nul. « *méta_coût* » est le coût du « *méta_problème* » mentionné précédemment. « *coût_min* » est le coût compréhensif minimum de désassemblage dans le D-L-N. « *lien_{coût_min}* » est le premier arc de v_i dans le chemin de désassemblage avec le coût minimum de désassemblage (« *coût_min* »). « *départ* » est le point de départ de l'arête e_{ij} , « *fin* » est le point final .

Pour tout arc $e_{ij} \in E$, la structure de données est exprimée dans le tableau 7.2.

« *départ* » est le point de départ de d'arc e_{ij} . « *même_départ* » et « *même_fin* » sont deux points de liaison qui pointent le premier arc et qui a le même point de départ que la arc e_{ij} , respectivement ; « *extrémité* » exprime le relation logique entre e_{ij} et l'arc avec les points « *même_fin* » : 1 exprimant la relation ET, 0 exprimant un relation OU ; « *coût* » est le coût de désassemblage de l'arc e_{ij} .

Tableau 7.2 : Structure de données pour un nœud et un arc [54]

Structure de données pour un nœud (composant)

<i>num_série</i>	<i>en_degrée</i>	<i>premier_entrée</i>	<i>premier_sortie</i>	<i>méta_coût</i>	<i>coût_min</i>	<i>lien_{coût_min}</i>
------------------	------------------	-----------------------	-----------------------	------------------	-----------------	--------------------------------

Structure de données pour un arc (activité)

<i>départ</i>	<i>fin</i>	<i>même_départ</i>	<i>même_fin</i>	<i>extrémité</i>	<i>coût</i>
---------------	------------	--------------------	-----------------	------------------	-------------

Algorithme pour le coût minimum de désassemblage

L'algorithme du coût minimum de désassemblage pour un produit comprend six étapes :

1. Initialisation de D-L-N, à savoir construction de la structure de la liste orthogonale de stockage de D-L-N ;
2. Définition du composant à désassembler dans D-L-N ;

3. Génération du réseau incident positif dirigé, à savoir enlever les arcs redondants dans D-L-N ;
4. Le calcul du coût du meta-problème pour chaque nœud (voir figure 7.7) ;
5. Utilisation de l'algorithme directeur positif de coupage pour calculer de coût compréhensif minimum de désassemblage de chaque nœud, et enregistrer le chemin de désassemblage correspondant (voir figure 7.8) ;
6. Du composant à désassembler, un algorithme de recherche inversé qui suit la politique de première profondeur est utilisé pour acquérir le chemin de désassemblage avec le coût minimum de désassemblage pour le composant, et sa sortie (voir figure 7.9).

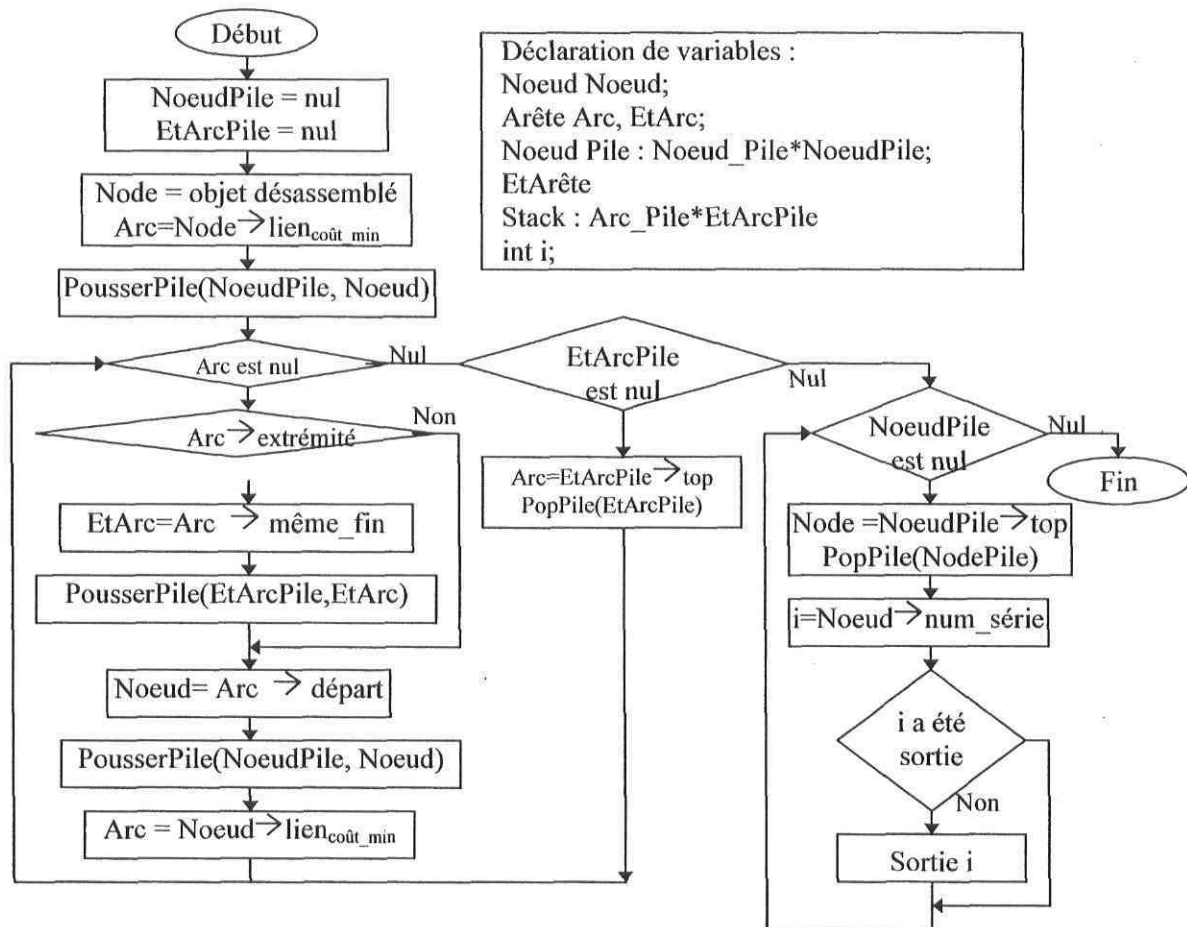


Figure 7.9 : Algorithme de recherche inversée

Source: Xu, Huang, Ling, Hu, Li, Hu [54]

Exemple de résultats

Nous prenons un D-L-N quelconque illustré à la figure 7.10(a).

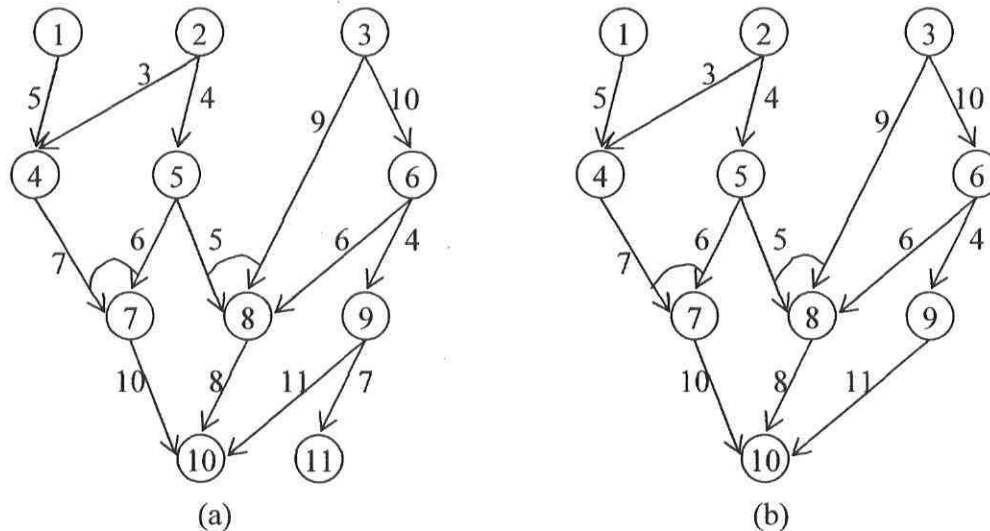


Figure 7.10 : Exemple pour le réseau logique de désassemblage

Source : Xu, Huang, Ling, Hu, Li, Hu [54]

Tableau 7.3 : Exemple de structure de magasin des nœuds [54]

num série	en degré	premier Entrée	premier sortie	méta coût	coût min	lien _{coût min}
1	0	nul	$e_{1,4}$	5	5	nul
2	0	nul	$e_{2,4}$	7	7	nul
3	0	nul	$e_{3,8}$	19	19	nul
4	2	$e_{1,4}$	$e_{4,7}$	7	10	$e_{2,4}$
5	1	$e_{2,5}$	$e_{5,7}$	11	15	$e_{2,5}$
6	1	$e_{3,6}$	$e_{6,8}$	10	20	$e_{3,6}$
7	2	$e_{4,7}$	$e_{7,10}$	10	30	$e_{4,7}$
8	3	$e_{5,8}$	$e_{8,10}$	8	24	$e_{6,8}$
9	1	$e_{6,9}$	$e_{9,10}$	18	32	$e_{6,9}$
10	3	$e_{10,7}$	nul	0	24	$e_{8,10}$

Le nombre sur les arcs définit le temps utilisé pour l'activité de désassemblage. Le point de désassemblage 10 est le composant à désassembler. En utilisant l'algorithme pour la génération du réseau d'incident directeur positif, nous obtenons la figure 7.10(b). Ensuite l'algorithme pour le coût minimum de désassemblage nous donne les résultats du tableau 7.3, où le chemin avec le coût minimum de désassemblage pour du composant à désassembler 10 est 3→6→8→10 avec un coût de désassemblage de 24 (en unité de temps).

7.3 Algorithme génétique avec aspect économique et environnemental

Il est possible d'utiliser les algorithmes génétiques pour faire l'évaluation de la séquence optimale de désassemblage en prenant en considération l'analyse économique et environnementale du produit. Voici donc l'utilisation de l'algorithme génétique (GA) pour générer la séquence optimale de désassemblage, vue par Seo, Park, et Jang [40]. Tout d'abord une analyse économique et environnementale est effectuée. Ensuite l'algorithme génétique est présent. Il comporte les étapes suivantes : l'illustration, l'objectif, la représentation, la population initiale, l'évaluation de la fonction de forme, la reproduction, le croisement, et finalement la mutation.

Définition de l'algorithme génétique

L'algorithme génétique est une méthode de recherche intelligente demandant une compréhension d'un domaine spécifique pour résoudre le problème. Les algorithmes génétiques recherchent une population de points, différents de la technique d'énumération où la fonction d'objectif est calculé à chaque point dans l'espace de recherche. L'algorithme génétique imite le processus de l'évolution naturelle en combinant l'ajustement entre les structures des solutions avec un mélange, pas encore structuré, d'échange d'information, et de cette façon créer une progéniture. La progéniture déplace la structure de solution possible durant chaque génération.

Analyse économique et environnementale

Pour obtenir la valeur de recyclage, une analyse du cycle de vie (LCA, section 2.2.1) du système de recyclage du matériel est nécessaire, et les facteurs environnementaux sont identifiés. Le coût environnemental est obtenu par le calcul du coût de prévention de pollution pour les facteurs environnementaux utilisant l'analyse du coût total (TCA, section 2.2.2).

Algorithmes génétiques pour une séquence optimale de désassemblage

La figure 7.12 illustre les différentes étapes d'utilisation des algorithmes génétiques pour obtenir une séquence optimale de désassemblage en prenant en considération l'analyse économique et environnementale.

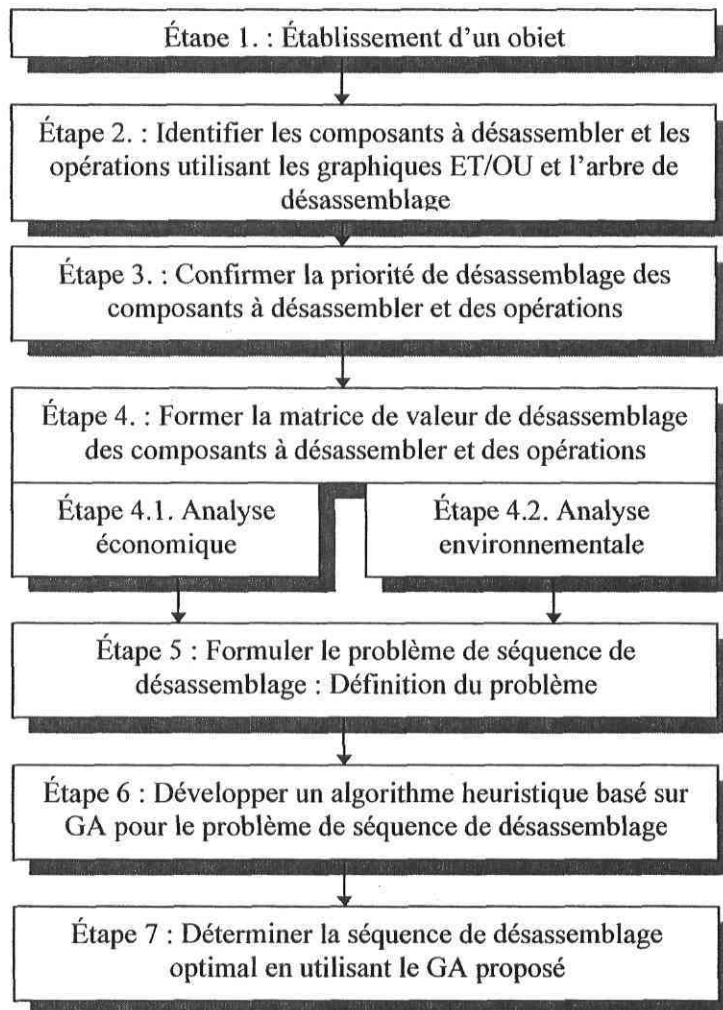


Figure 7.11 : Algorithme d'une séquence de désassemblage optimale

Source : Seo, Park, et Jang [40]

Dans GA, une solution candidate est représentée par une séquence de nombres considérés comme un chromosome ou séquence. Un chromosome potentiel a une solution qui est déterminée par ça fonction de forme, avec le problème d'optimisation toujours prit en

considération. Une série judicieusement sélectionnée de chromosomes est nommée une population et la population comme un temps donné est une génération. La taille de la population a un impact significatif sur la performance de l'algorithme. Les algorithmes génétiques opèrent avec des générations consistant en trois opérations principales :

1. Reproduction : sélection de copies de chromosomes proportionnels à la valeur de forme.
2. Croisement : un échange de sélections de chromosomes.
3. Mutation : une modification aléatoire des chromosomes.

Les chromosomes résultant de ces trois opérations, souvent nommés progéniture ou enfant, forme la prochaine génération de population. La procédure est répétée pour un nombre désiré de générations, habituellement jusqu'à un point où le système converge vers la séquence avec la meilleure performance significative.

Illustration

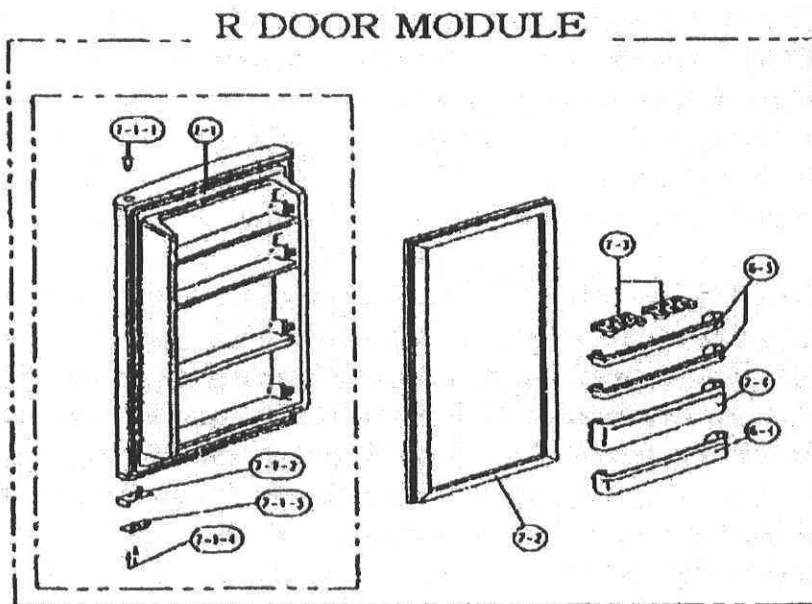


Figure 7.12 : Exemple pour l'utilisation de l'algorithme génétique

Source : Seo, Park, et Jang [40]

Voici donc, l'explication et la validation des séquences d'opération de désassemblage générées par l'approche de l'algorithme génétique, dans le cas du désassemblage du module R DOOR d'un réfrigérateur. Les opérations de désassemblage sont définies par un arbre de désassemblage (DT) ayant comme étiquettes $O_1, O_2, O_3, \dots, O_n$ (O = opération).

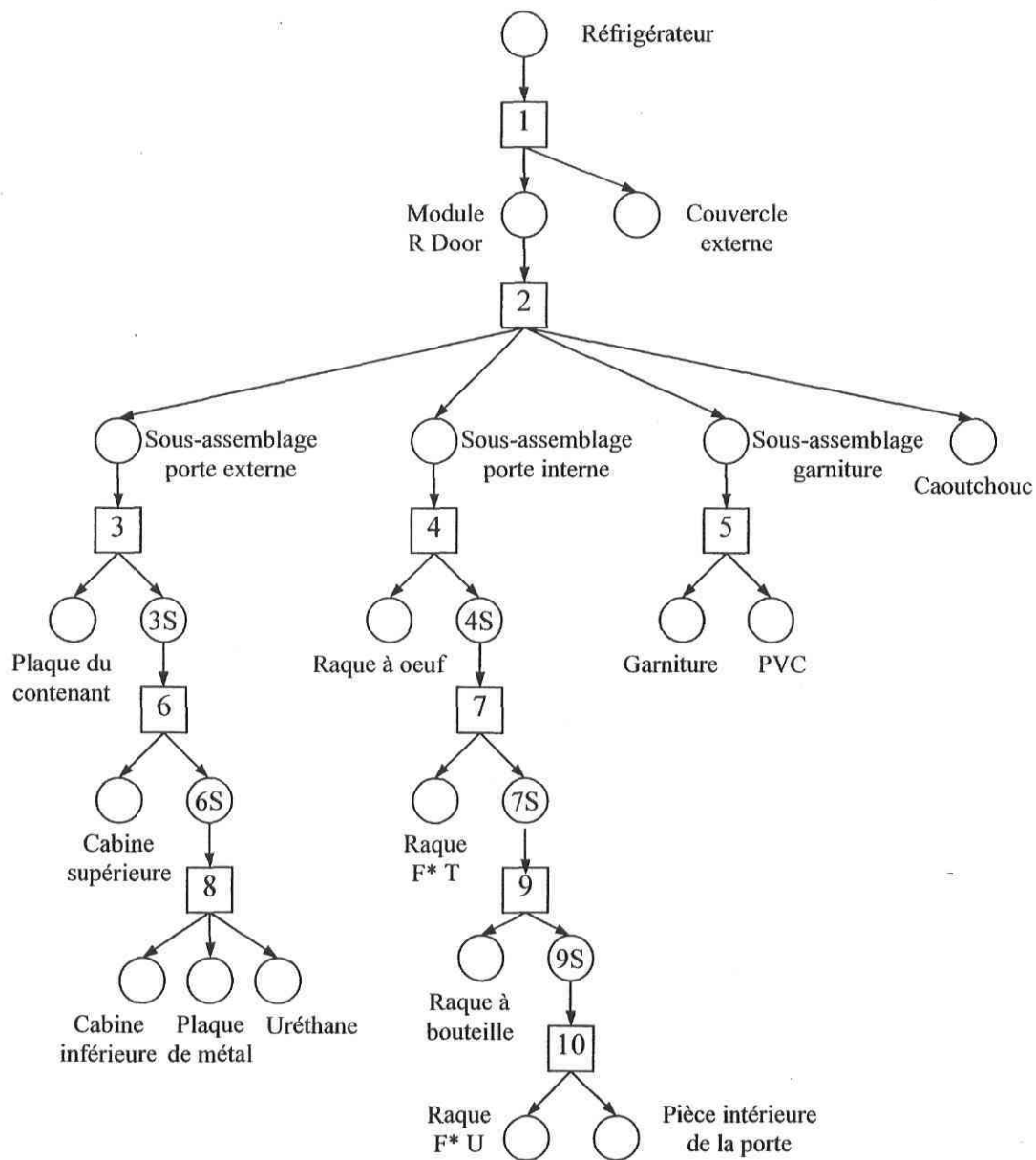


Figure 7.13 : Arbre de désassemblage pour le module R DOOR

Source : Seo, Park, et Jang [40]

Pour les besoins de l'illustration, les opérations de désassemblage sont codées comme 1,2,3,...,n, correspondant à $O_1, O_2, O_3, \dots, O_n$. Les opérations de désassemblage et la priorité de désassemblage sont illustrés dans l'arbre de désassemblage (figure 7.13), avec les contraintes considérées. Du graphique de priorité, la priorité de désassemblage et la matrice de valeurs sont générées pour chaque paire d'opérations de désassemblage basées sur les valeurs correspondant au coût de désassemblage, la valeur de recyclage et le coût de rejet.

Objectif

L'objectif primaire est de maximiser le profit associé avec le désassemblage d'un produit. Le problème de génération de séquences optimales de désassemblage est formulé en terme de maximisation des valeurs de séquence dépendante de désassemblage entre les opérations de désassemblage :

$$\text{Max } Z = \sum_i \sum_j Dv_{ij} x_{ij}$$

où, Dv_{ij} représente la valeur de désassemblage de passage entre les opérations de désassemblage i et j , x_{ij} représente la variable entre les opérations de désassemblage i et j .

Représentation

Pour le problème présent, la séquence est représentée par une collection de n éléments correspondant à n opérations données dans la priorité de désassemblage et la matrice de valeurs des composants enlevés.

Population initiale

La population initiale ne consiste pas en de simple génération de séquence aléatoire, étant donné que la priorité des opérations pour chaque composant enlevé n'est pas garantie. Pour créer une séquence valide, un élément de séquence est généré aléatoirement. Les premières opérations des groupes et du procédé partiel de chaque forme est répétés en

choisissant des éléments à partir des groupes restants d'opérations jusqu'à ce que toutes les opérations soient représentées dans la séquence. Le module R DOOR du réfrigérateur contient 10 opérations et 14 composants. Pour la priorité de désassemblage, les opérations sont $O_1 \rightarrow O_2 \rightarrow (O_3, O_4, O_5), O_3 \rightarrow O_6$ et $O_6 \rightarrow O_8$.

Chaque séquence dans une population doit contenir n éléments correspondant à n opérations. Les premiers éléments de la séquence sont générés aléatoirement du premier élément du h^e groupe, c.-à-d. de $O_1, O_2, O_3, \dots, O_n$. Si O_2 est sélectionné aléatoirement du k^e groupe, alors les seconds éléments de la séquence doivent être générés aléatoirement de O_1, \dots, O_n . Cette procédure est répétée jusqu'à ce que tous les éléments de la séquence soient remplis des premiers éléments du groupe d'opérations retenu. Ici, O_1, O_2, O_3, \dots , est une séquence initiale valide de la population.

Évaluation de la fonction de forme

L'objectif du problème de génération des séquences désassemblage est d'obtenir une séquence d'opérations de désassemblage optimale résultant de la valeur économique maximale de la priorité donnée de désassemblage et de la matrice de valeurs. La fonction objectif, qui est la somme des valeurs économiques de paires d'opérations de désassemblage, est calculée pour chaque séquence dans la population. Les valeurs économiques vont correspondre au nombre de composants qui doivent être convertis dans chaque mesure des attributs comme le temps de désassemblage, le coût de manipulation, la procédure de recyclage, de rejet, et le type de contrainte de chaque composant. Les valeurs de forme de chaque séquence pour la génération suivante sont obtenues.

Reproduction

Cette opérateur génétique est utilisé pour générer une nouvelle population qui a des meilleurs séquences que la vieille population. La sélection de la meilleur séquence est basée sur le compte actuel en arrivant à l'étape précédente. La population reproduite est nommée parent 1 et est utilisée pour l'opération génétique suivante.

Croisement

Dans cette étude, un nouvel opérateur de croisement est désigné par la modification du croisement général partiellement assorti pour assurer la priorité des opérations de désassemblage et produire les progénitures faisables. Pour cela, deux parents sont sélectionnés aléatoirement de la population. En ce basant sur la longueur de la séquence, deux sites de croisement sont aléatoirement générés pour sélectionner un segment d'un parent entre ces sites de croisement. La progéniture, enfant 1, est générée par l'arrangement des éléments du segment sélectionné dans ce parent, avec l'ordre des éléments restants étant les mêmes que ceux du premier parent. Le rôle de ces parents est de changer d'ordre pour produire une progéniture, enfant 2. L'opérateur de croisement peut être illustré comme suit :

Parent 1 : 1 2 3 4 5 6 7 8 9 10 Parent 2 : 1 5 4 3 2 6 7 9 8 10

Nous considérons deux sites de croisement aléatoire comme $X = 3$ et $Y = 7$, et le segment de parent 1 entre un site de croisement est 4, 5, 6, 7. L'arrangement des éléments sélectionnés dans l'ordre du parent 2, résulte en 5, 4, 6, 7, alors la progéniture, enfant 1, du parent 1 est générée comme étant : 1, 2, 3, 4, 5, 6, 7, 9, 8, 10. Similairement, la progéniture, enfant 2, du parent 2, est générée comme étant : 1, 5, 4, 2, 3, 6, 7, 8, 9, 10. Les deux nouvelles progénitures générées sont les suivantes :

Progéniture 1 : 1 2 3 4 5 6 7 9 8 10 Progéniture 2 : 1 5 4 3 2 6 7 9 8 10

Mutation

L'opérateur de mutation fait un changement aléatoire l'un élément ou plus de la séquence. La mutation est exécutée dans une petite probabilité nommée la mutation de probabilité. Cela protège la perte de la séquence potentiellement utilisable et évite de l'éloigner de l'optimum local. L'opérateur de mutation proposé ici modifie deux éléments aléatoirement pour obtenir la population résultante. Cependant, cela est une possibilité de

séquence à éviter (infaisable) par violation de la priorité des opérations de désassemblage. Ici, un nouvel opérateur de mutation est introduit pour surveiller la faisabilité des éléments de séquences obtenues. Si la séquence est infaisable, la valeur totale est définie comme une valeur très faible qui ne devrait pas apparaître dans une génération future.

8 Stratégies pour l'entretien

Comme nous l'avons vu précédemment, certaines stratégies peuvent être utilisées pour l'entretien et la réparation de produit. Cependant, il existe des méthodes de désassemblage utilisées spécifiquement dans le but de faire de l'entretien. Dans les pages qui vont suivre nous allons analyser et détailler deux stratégies, toutes aussi différentes l'une de l'autre, qui sont développées spécialement pour la maintenance et la réparation de produit.

Analyse des deux stratégies d'entretien

Li, Khoo et Tor [32] présentent la génération de séquence de désassemblage dans le cas de la planification de la maintenance. Une représentation schématique particulière est utilisée pour illustrer dynamiquement la géométrie et les contraintes de priorité dans un produit. Avec ce schéma la procédure pour la détermination des séquences de désassemblage possibles peut être simplifiée, cela permettant la réduction de la taille de l'espace de recherche et le groupement de certains composants en sous-assemblage. Ils obtiennent la séquence de désassemblage optimale, liée avec un composant cible, en utilisant l'algorithme génétique. De cette manière, il est possible de travailler avec des produit ayant des relations complexes et un grand nombre de composant.

Srivansan, Gadh [43] ont développé l'algorithme de désassemblage sélectif global de composants sélectionnés par l'analyse des contraintes géométriques de l'assemblage. L'algorithme proposé est applicable pour des géométries 2D et 3D. Il n'utilise pas les matrices de désassemblage. La séquence déterminée par l'algorithme de désassemblage sélectif global est optimale. L'algorithme de désassemblage sélectif global est facilement applicable pour tous les objectifs de désassemblage répondants aux hypothèses posées. Son application très visuel et sa simplicité font de cet algorithme un choix intéressant. Il ne prend malheureusement pas en considération l'aspect coût et les hypothèses très restrictives limitent son utilisation dans beaucoup de situations.

8.1 Algorithme génétique pour la maintenance

L'utilisation de l'algorithme génétique pour générer la séquence optimale de désassemblage pour la maintenance tel que vu par Li, Khoo et Tor [32]. La méthode de l'algorithme diffère de celle de la précédente (section 7.3) du fait qu'elle utilise l'algorithme génétique pour aller chercher un composant spécifique dans le produit. Cette analyse est donc réalisée dans le cadre de la maintenance des produits.

Les étapes d'utilisation de l'algorithme génétique sont les suivantes (figure 8.1) :

- Étape 1 :** Trouver le composant visé v dans le graphique hybride et regrouper ses nœuds adjacents de contact et de non-contact dans deux listes, A et B, respectivement. Identifier les nœuds à enlever du graphique hybride courant. Placer les nœuds à enlever au début de la liste, auxquels ils appartiennent.
- Étape 2 :** Vérifier que le composant visé v est maintenant un nœud libre, si non, le mettre dans l'opération correspondante de l'espace de recherche, et aller à l'étape 5 ; autrement ; sélectionner le nœud vérifié actuel v' au nœud non-vérifié suivant dans les listes A et B ($v' \in B$ ou A, liste B avant la liste A), localiser tout sous-assemblage ou nœud libre. Si les listes A et B sont vides, aller à l'étape 5.
- Étape 3 :** Vérifier que le composant v' est un composant libre. Si oui, aller à l'étape 4 ; autrement, identifier les nœuds adjacents de contact et non-contact correspondants au nœud v' et les ranger les dans la liste A et B, respectivement. Chaque exécution, aller à l'étape 2 jusqu'à ce qu'il n'y est pas plus de nœud libre ou sous-assemblage extirpé pouvant être trouvés. Placer toutes les opérations rapportées des nœuds restants avec le composant cibler dans l'espace de recherche des séquences de désassemblage et aller à l'étape 5.

Étape 4 : Si le nœud v' est un composant libre, placer son opération relative dans l'espace de recherche des séquences de désassemblage optimiser et enlever le nœud v' du graphique hybride. Si le nœud est un lien du sous-assemblage, placer l'opération reliée au point de sous-assemblage dans l'espace de recherche des séquences de désassemblage et enlever tous les nœuds dans le sous-assemblage du graphique hybride, mettre à jour les listes A et B et aller à l'étape 2.

Étape 5 : Utiliser l'approche d'optimisation de la séquence de base. Ici, l'algorithme génétique présenté dans les pages suivantes, permet trouver la séquence de base optimale avec le coût le plus bas.

Optimisation de la séquence (étape 5)

Représentation du chromosome

La méthode suivante représente les séquences de désassemblage ou les chromosomes. Le schéma de représentation du chemin convient bien pour la planification de séquences de désassemblage. Par exemple, si un plan de désassemblage est donné par la séquence suivante :

Opération 12 - Opération 13 - Opération 11 - Opération 10 - Opération 0

Utiliser ce schéma peut facilement être exprimé par :

12 13 11 10 0

Une telle représentation élimine le besoin de coder ou décoder les chromosomes dans une séquence de désassemblage d'un produit.

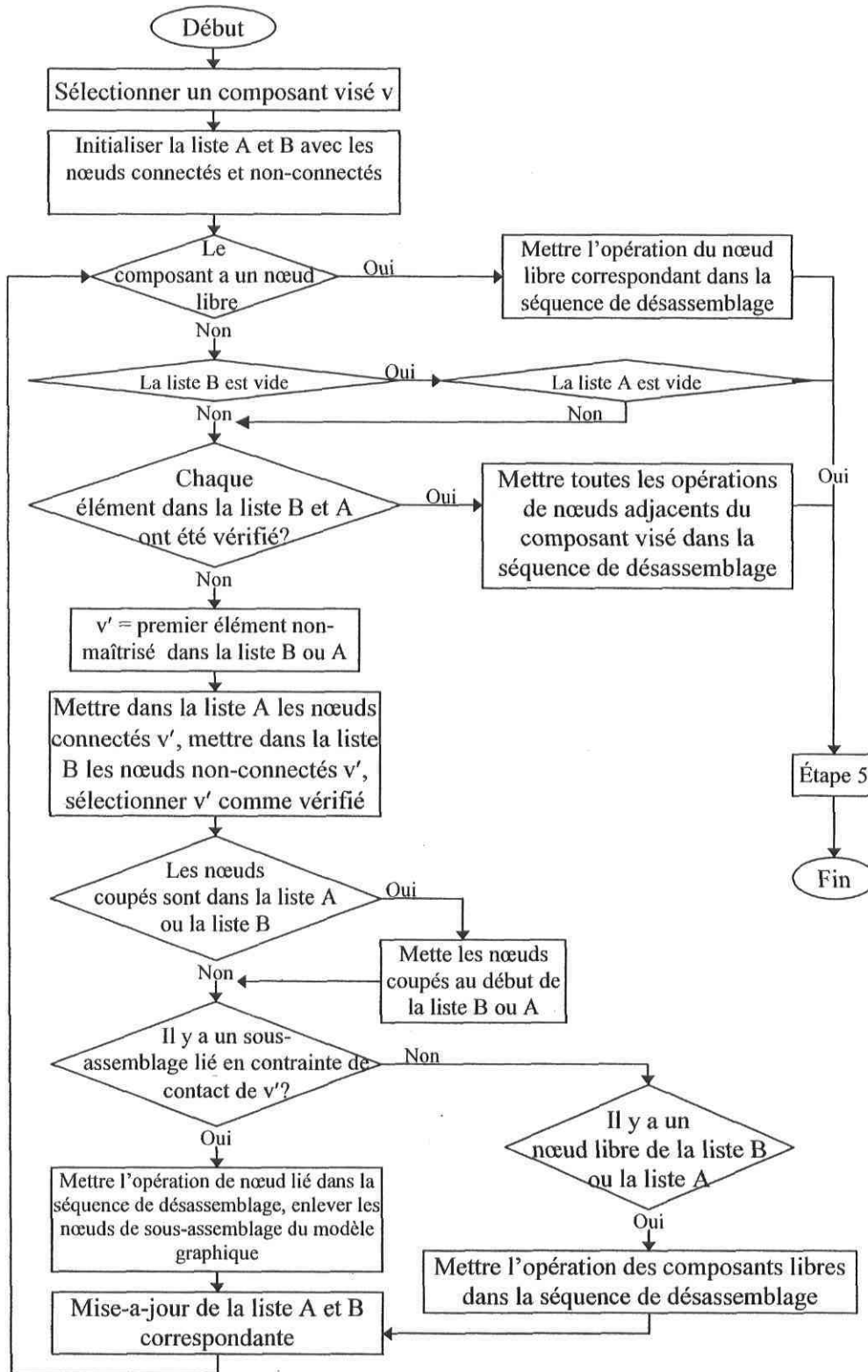


Figure 8.1 : Génération des séquences de désassemblage pour la maintenance

Source : Li, Khoo et Tor [32]

Sélection de la population

Une population est sélectionnée encourageant la survie des individus les plus en forme. L'approche roulette est utilisée pour identifier les chromosomes s'emmêlant avec la population. Cette approche donne à chaque chromosome une chance de propagation. La sélection de la prochaine génération est basée sur la distribution de probabilité des valeurs d'aptitudes. Chaque individu dans la population correspondante a une partie de la roulette.

Opérateur de croisement

Les probabilités de croisement et de mutation sont fixés à 0.9 et 0.01. respectivement. Pour une opération croisement, le site de croisement est sélectionné aléatoirement. Utiliser la méthode OX : une section de chromosomes parents est préservée durant l'opération de croisement. Essentiellement, l'opérateur de croisement exploite une propriété de représentation de chemin, qui pose une emphase sur l'ordre des composants (gènes) dans une séquence de désassemblage (chromosomes).

Nous considérons deux chromosomes parents, p_1 et p_2 , avec deux points de croisement sélectionnés aléatoirement indiqués par « | » :

$$p_1 = (0\ 1\ 2\ 10\ | 3\ 4\ 5\ 6\ | 7\ 8\ 9)$$

$$p_2 = (4\ 5\ 0\ 10\ | 9\ 8\ 6\ 3\ | 7\ 1\ 2)$$

Après l'opération de croisement, les chromosomes enfants, c_1 et c_2 , sont produits de la façon suivante. Initialement, le segment entre les points coupés est copié et représenté comme suit :

$$c_1 = (x_4 x_5 x_6 x_7 | 3 4 5 6 | x_1 x_2 x_3)$$

$$c_2 = (y_4 y_5 y_6 y_7 | 9 8 6 3 | y_1 y_2 y_3)$$

Les traits génétiques des chromosomes parents sont préservés. Subséquemment, partant du point du second croisement d'un chromosome parent comme $x_1 x_2 x_3$ ou $y_1 y_2 y_3$, les composants d'un parent sont copiés dans le même ordre, omis dans le cas de figures qui sont déjà présents. Cela implique que la séquence de composants dans le second parent peut être réarrangé comme suit :

$$7-1-2-4-5-0-10-9-8-6-3$$

Après l'enlèvement des composants 3, 4, 5 et 6, qui sont toujours présents dans le premier chromosome enfant, la séquence suivante peut être obtenue :

$$7-1-2-0-10-9-8$$

Cette séquence est alors utilisée pour générer le premier enfant chromosome (début du second point de mutation) comme suit :

$$c_1 = (0 10 9 8 | 3 4 5 6 | 7 1 2)$$

Similairement, c_2 est donné par la séquence suivante :

$$c_2 = (2 10 4 5 | 7 8 6 3 | 9 0 1)$$

Opérateur de mutation

Ici nous adoptons la mutation d'échange. Pour l'illustrer, nous choisissons aléatoirement 9 et 7 de c_2 sont pour faire une mutation d'échange. Ce résultat donne une nouvelle séquence qui est générée comme suit :

$$c_2 = (2 \ 10 \ 4 \ 5 \ | \ 7 \ 8 \ 6 \ 3 \ | \ 9 \ 0 \ 1)$$

Mécanisme de réparation

Après la génération d'une nouvelle population, la faisabilité de chaque chromosome est alors estimé en utilisant :

1. La contrainte géométrique décrite par l'arc non-dirigée entre deux nœuds.
2. L'information de priorité décrivant à peu près le désassemblage par des arcs dirigés, dans la représentation d'un graphique hybride.

Les séquences de désassemblages ou chromosomes qui ne sont pas faisables sont réparées en utilisant un mécanisme de réparation, qui scrute les gènes d'un chromosome, de gauche à droite, un par un. Une fois qu'un gène invalide est retrouvé, un gène généré aléatoirement le remplace. Après cela, la validité du remplacement du gène est regardée à nouveau en utilisant les contraintes de contacts et de non-contacts. Après chaque exécution de l'opération génétique, une nouvelle génération de chromosome est créée.

Fonction de forme

Dans cette section, deux facteurs importants, nommés l'orientation de chaque opération de désassemblage et la nécessité de changement d'outil sont considérés dans la dérivation de la fonction de forme pour optimisation comme l'un ou l'autre des résultats dans l'utilisation du temps de désassemblage. Cela se traduit par :

$$f(x, y) = \sum_{i=1}^n (ax_i + by_i)$$

où :

$f(x, y)$ est la fonction de forme ;

n est le nombre d'opérations de désassemblage dans l'ordre de chaque composant visé ;

x est le niveau de forme atteint par chaque opération de désassemblage en relation avec le besoin de changement d'outil ;

y est le niveau de forme atteint par chaque opération de désassemblage en relation avec le changement d'orientation ;

a et b sont les coefficients de désassemblage de x et y , respectivement.

Le coefficient dénote l'importance de x et y dans le calcul du niveau de forme utilisé.

En utilisant toutes les étapes de l'algorithme génétique : la représentation du chromosome, la sélection de la population, l'opérateur de croisement et de mutation, le mécanisme de réparation et la fonction de forme, nous trouvons le chemin optimal pour se rendre au composant visé.

Exemple d'application de la méthode

La lampe de poche présentée à la figure 8.2(a), est l'exemple utilisé pour cette méthode. Il est à noter que ce n'est pas un produit complexe, comme il comporte seulement sept composants. La figure 8.2(b) illustre le graphique hybride de liaison. Nous supposons que l'ampoule, le nœud 3, est brisée et nécessite un remplacement. C'est le composant cible de notre exemple. Voici donc la procédure pour effectuer le remplacement de ce composant.

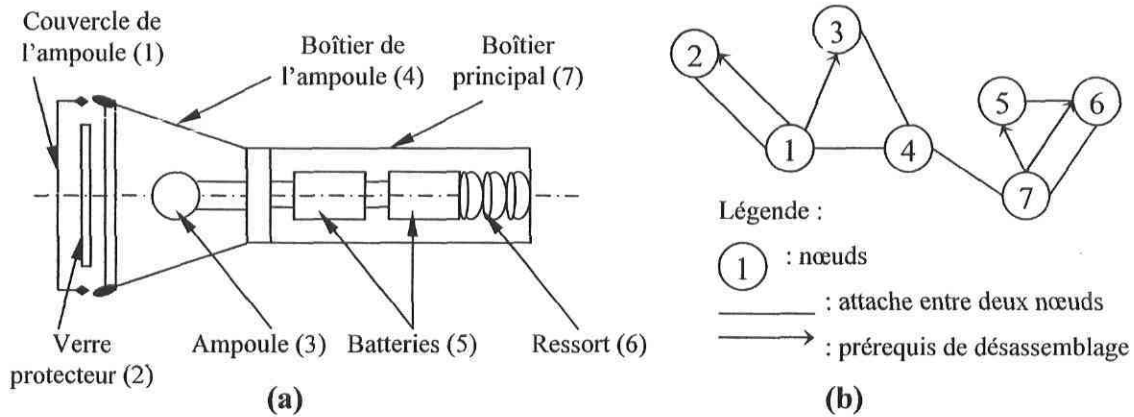


Figure 8.2 : *Lampe de poche et son graphique hybride*

Source : Li, Khoo et Tor [32]

Étape 1 : Sélectionner le composant cible comme étant l'ampoule, le nœud 3, et placer les nœuds adjacents de contact et non-contact dans les listes A et B, respectivement. Dans la liste actuelle, listes A et B, les nœuds 4 et 1 sont tous les deux des nœuds de séparation, ainsi les deux listes contiennent seulement un nœud chacune. Ainsi, le nœud de séparation est placé au début de la liste correspondante (tableau 8.1).

Tableau 8.1 : *État de départ étalant les séquences possibles [32]*

Composant cible	Composant actuellement analysé	Liste A	Liste B
3		4	1

Étape 2 : Analyser le composant cible, nœud 3, pour vérifier si c'est un nœud libre. Ce n'est pas un nœud libre dans ce cas, tous les nœuds de la liste B sont analysés un par un, suivi de ceux de la liste A. Après compilation, le composant actuel analysé est assigné au nœud 1 de la liste B (tableau 8.2).

Tableau 8.2 : *Identification du sous-assemblage [32]*

Composant cible	Composant actuellement analysé	Liste A	Liste B
3		4	1
	1	4, 2	-
			← Sous-ensemble pont Sub(1,4) est localisé

Étape 3 : Basée sur la définition du sous-ensemble pont, Sous-ensemble Pont $Brg(1,4)$ avec le nœud 1 comme nœud de séparation aillant été identifié, deux sous-assemblages, nommés $Sub(1,\{1,2\})$ et $Sub(4,\{3,4,5,6,7\})$, sont formés pour l'enlèvement du sous-ensemble pont $Brg(1,4)$.

Étape 4 : Ainsi, l'opération de désassemblage, $Opt(1,4)$, pour éliminer les contraintes de désassemblage de ce sous-ensemble pont est placée dans l'espace de recherche pour la planification de la séquence de désassemblage. Le graphique hybride est simplifié comme illustré à la figure 8.3. Après avoir mis à jour les listes A et B, l'étape 2 est révisée et le composant cible, nœud 3, est examiné de nouveau. Nous effectuons une nouvelle vérification à savoir si le nœud 3 est un nœud libre. L'opération correspondante $Opt(3,4)$, pour éliminer les contraintes entre les nœuds 3 et 4 est placée dans l'espace de recherche pour la planification de la séquence de désassemblage.

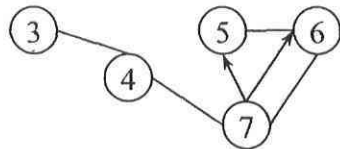


Figure 8.3 : *Le graphique hybride après l'enlèvement de $Sub(1,\{1,2\})$*

Source : Li, Khoo et Tor [32]

Étape 5 : L'information de priorité entre les deux opérations $Opt(1,4)$ et $Opt(3,4)$, est évidente. Dans ce cas, $Opt(1,4)$ va être réalisée en premier et suivie par $Opt(3,4)$. Comme ce cas d'étude est relativement simple, l'optimisation des séquences n'est pas nécessaire.

Ainsi, la séquence de désassemblage pour la maintenance du composant 3, qui est le nœud 3, est l'élimination de la contrainte de contact entre les composants 1 et 4. Conséquemment, la contrainte de non-contact entre les composants 1 et 3 est l'éliminée par l'enlèvement du sous-assemblage $Sub(1,\{1,2\})$ du graphique. Conséquemment, la contrainte de non-contact entre les composants 3 et 4 est l'éliminée. La séquence de désassemblage trouvée est faisable et correspond avec le manuel de désassemblage de la lampe de poche.

8.2 Méthode du désassemblage sélectif global

L'analyse du désassemblage sélectif implique la détermination d'une séquence et des directions pour le désassemblage d'une série de composants sélectionnés. L'application inclut l'assemblage, la maintenance et le recyclage sélectif. La méthode suivante peut travailler avec deux types de contraintes géométriques, les contraintes spatiales et les contraintes définies par l'utilisateur (celui qui veut désassembler le produit). Les contraintes spatiales sont des contraintes imposées par l'assemblage et le désassemblage d'un composant dues à la position spatiale et la géométrie de tous les autres composants. Les contraintes définies par l'utilisateur sont des contraintes imposées par l'utilisateur par rapport à la géométrie du composant qui restreint quelques opérations d'assemblage/désassemblage.

Nomenclature

U_i : série de direction de désassemblage de C_i
 d_i : direction de désassemblage
 U : univers de direction de désassemblage pour l'assemblage A
 S : séquence de désassemblage
 RI : influence d'enlèvement

Hypothèses

1. Le mouvement relatif des composants est déterminé sans prendre en considération les outils, les fixations ou les robots requis pour exécuter ce mouvement .
2. L'assemblage est considéré comme étant rigide (non-déformable), sans friction et ayant une géométrie nominale (avec un minimum de tolérance).
3. Les composants doivent être désassembler linéairement (avec des translations linéaires seulement).
4. Les séquences de désassemblage sont mono-toniques (les composants sont totalement enlevés lors du désassemblage).

Étapes de résolution du désassemblage sélectif global

1. Détermination des attributs géométriques.
2. Comptabilisation de l'algorithme de désassemblage sélectif global pour les contraintes spatiales.
3. Modifications de l'algorithme pour incorporer les contraintes définies par l'utilisateur.

Étape 1 : Attributs géométriques

Cette méthode suppose qu'il y a un ensemble fixe de directions de désassemblage. Nous avons U dénotant l'ensemble des directions de désassemblage pour les composants dans A et U_i dénotant une série de directions de désassemblage de C_i après comptabilisation pour les contraintes directionnelles imposées dans C_i . Par exemple, dans la figure 8.4, $U = \{d_1, d_2\}$, sans contrainte directionnelle appliquée sur n'importe quel autre composant $U_i = U$, c'est-à-dire $U_1 = U_2 = U_3 = U_4 = U_5 = \{d_1, d_2\}$. Autrement si, par exemple, C_1 est contraint en mouvement et C_2 est contraint le long de d_1 , alors $U_1 = \{\}$, $U_2 = \{d_2\}$, $U_3 = U_4 = U_5 = \{d_1, d_2\}$.

Définition 1 (désassemblage spatial) : Le désassemblage spatial de C_i , dénoté comme Δ_i , est VRAI s'il existe une direction de désassemblage $d_j \in U_i$, où C_i ne se heurte pas avec tout autre $C_k \in A (k \neq i)$. Un composant n'ayant pas de contrainte spatiale est dénoté comme C_b . Dans notre exemple : $\Delta_2 = \text{VRAI}$ et $\Delta_5 = \text{FAUX}$ pour C_2 et C_5 , respectivement.

Définition 2 (série de limite spatiale) : La série de limite spatiale, dénotée comme $\beta_r (r > 0)$, est définie comme suit : pour $r = 1$, $\beta_r =$ série de C_b dans A . Pour $r > 1$, $\beta_r =$ série de C_b dans l'assemblage avec les composants $A - (\beta_{r-1} + \beta_{r-2} \dots + \beta_1)$. Dans notre exemple, les séries de limite spatiale sont $\beta_1 = \{C_2, C_4\}$, $\beta_2 = \{C_5\}$, $\beta_3 = \{C_3\}$ et $\beta_4 = \{C_1\}$.

Définition 3 (série d'adjacence spatiale) : La série d'adjacence spatiale de $C_i \in \beta_r$ (dénotée comme x_i) est définie comme $\alpha_i = (\beta_{r-1} \cup \beta_{r-2} \dots \cup \beta_1)$. Dans l'exemple de la figure 8.3, $C_3 \in \beta_3$ et $\alpha_3 = (C_2, C_4, C_5)$. Similairement pour $C_5 \in \beta_2$ et $\alpha_5 = (C_2, C_4)$.

Définition 4 (influence spatiale d'enlèvement) : Nous avons C' dénotant une sous-série des composants dans A . Si en l'absence de $C' \subseteq \alpha_i$ dans A , $\Delta_i = \text{VRAI}$, alors l'influence spatiale d'enlèvement de C' dans C_i dénotée comme $RI_i^{C'} = \text{VRAI}$. Dans notre exemple : l'influence spatiale d'enlèvement de (C_4, C_5) dans C_3 est $RI_3^{4,5} = \text{VRAI}$.

Étape 2 : Algorithme du désassemblage sélectif global

Un graphique de liaison est construit pour déterminer l'ordre topologique de désassemblage des composants. Dans le graphique, les nœuds correspondent aux composants dans A et les arcs correspondent à l'influence d'enlèvement entre les composants. Un arc $C_i \rightarrow C'$ dans le graphique, avec comme attribut d_j (un arc dirigé) indique qu'il est possible de désassembler C_i le long de d_j après l'enlèvement de C' .

Dans notre exemple, nous considérons A avec $C = \{C_3\}$ et $U = \{d_1, d_2\}$. Le graphique de liaison est présenté dans la figure 8.4. C_3 est désassemblé sans contrainte spatiale le long de d_1 après l'enlèvement de C_4 et C_5 ($RI_3^{4,5} = \text{VRAI}$). C_5 est désassemblé sans contrainte spatiale le long de d_2 après l'enlèvement de C_2 ($RI_5^2 = \text{VRAI}$). D'ailleurs, $\Delta_4 = \text{VRAI}$ et $\Delta_2 = \text{VRAI}$ pour C_4 et C_2 , respectivement. Une séquence $S = \{C_b \rightarrow^p C_x\} (C_x \in C)$ est dérivable du graphique de liaison (où $C_b \rightarrow^p C_x$ dénote l'ordre de désassemblage de C_b à C_x) par l'intermédiaire de l'assortiment topologique du graphique de liaison : récursivement enlever des nœuds sans arc dirigé dans le graphe jusqu'à C_x permet de trouver S . Dans notre exemple, $S = \{C_2, C_4, C_5, C_3\}$ pour C_3 . L'algorithme du désassemblage sélectif global est le suivant :

Algorithme : GSD (Entrée : A, C ; Sortie : S)

- { Étape 1 : Déterminer les contraintes spatiales pour chaque $C_i \in A$ le long de chaque $d_j \in U_i$.
- Étape 2 : Déterminer les séries limites spatiales pour A.
- Étape 3 : Déterminer la série adjacence spatiale pour chaque $C_i \in A$.
- Étape 4 : Construction du graphique de liaison pour C.
- Étape 5 : Classement topologique du graphique de liaison et calculer S. }

Les figures 8.5 illustre l'utilisation de l'algorithme. Les différentes étapes de cette figure sont directement liées avec les cinq étapes de l'algorithme de désassemblage sélectif global.

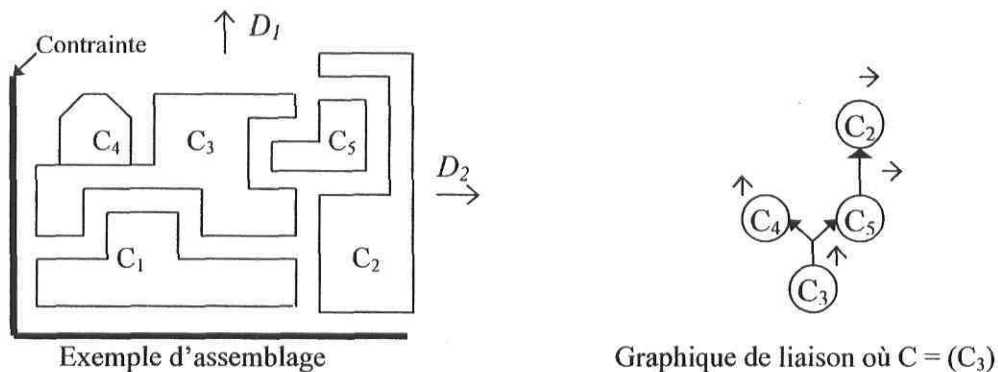


Figure 8.4 : Exemple d'assemblage avec graphique de liaison

Source : Srivansan, Gadh [43]

Étape 3 : Incorporation des contraintes définies par l'utilisateur

Un regroupement de composants peut être incorporé dans l'algorithme en le traitant comme un seul composant. En reprenant notre exemple, si C_2 et C_5 forme un sous-ensemble, alors le groupe (C_2, C_5) est contraint le long de d_1 par C_3 , car C_5 est contraint par C_3 le long de d_1 . Autrement (C_2, C_5) n'est pas contraint le long le d_2 , car C_2 n'est pas contraint le long de d_2 et que C_5 est contraint seulement par C_2 le long de d_2 . La figure 8.6 illustre le graphe de liaison pour $C = \{C_3\}$ avec $S1 = \{C_3, C_5\}$ et $S = \{C_4, S1, C_3\}$.

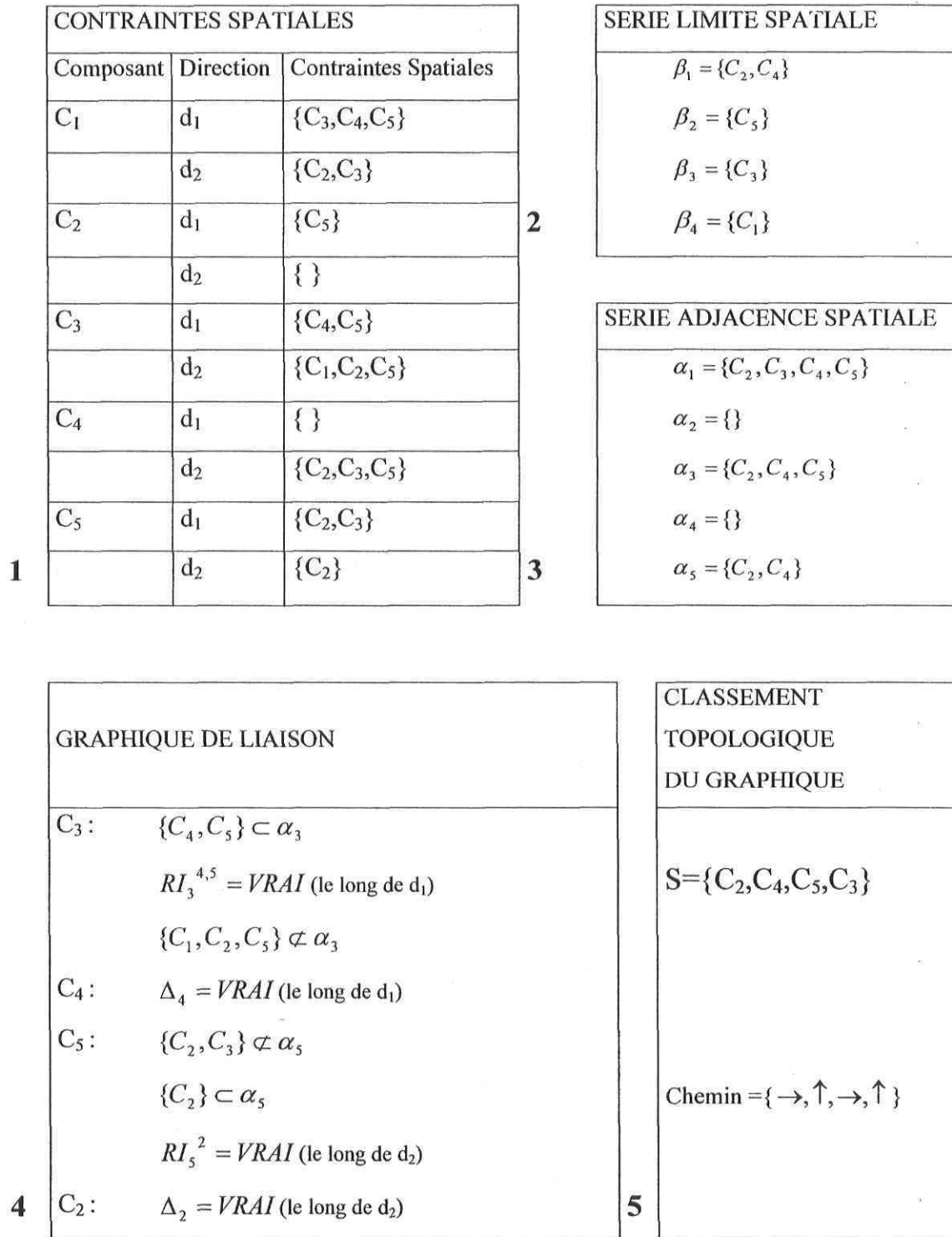


Figure 8.5: Illustration de l'algorithme de désassemblage sélectif global

Source : Srivansan, Gadh [43]

Les contraintes géométriques telle qu'une qui limite un composant pour se déplacer ou une qui contraint le long de l'une ou plusieurs directions de désassemblage peuvent être ajoutées à l'algorithme. Cela peut être effectué dans la définition de U_i pour C_i .

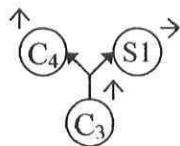


Figure 8.6: *Graphique de liaison pour $C = (C3)$ avec présence d'un sous-ensemble*

Source : Srivansan, Gadh [43]

9 Stratégie dans le cas d'une ligne de désassemblage

La stratégie de désassemblage suivante inclut deux nouveaux éléments importants. La présence de plusieurs postes de travail et dans le cas traité, la présence d'échec des tâches (séquence ou activité) de désassemblage. Ça ajoute à la complexité du problème mais permet d'éliminer des contraintes. Dans les autres stratégies, il est question de seulement un poste de travail. Il est toujours possible d'en avoir plusieurs mais la gestion et les problèmes possibles ne sont pas traités. Pour cela, l'étude des lignes de désassemblage avec possibilité d'échec de tâches présentée par Gungor et Gupta [14] est innovatrice et complète bien la question des stratégies de désassemblage de produit.

L'approche suivante permet de générer un plan de séquence de désassemblage automatiquement, pour le recyclage des produits et la nouvelle fabrication. Un algorithme est premièrement défini pour générer une géométrie basée sur les matrices de priorité de désassemblage (section 4.2.1). Ses dernières sont utilisées pour générer un arbre hiérarchique de désassemblage (section 4.2.2) qui représente la faisabilité du plan de la séquence de désassemblage. Les éléments pertinents sont la génération du plan de séquence de désassemblage, l'algorithme de création de la matrice de priorité de désassemblage, les propriétés des résultats de la matrice de désassemblage, et l'algorithme de réduction de l'arbre hiérarchique de désassemblage. Nous présentons des algorithmes pour la création de la matrice de désassemblage et de réduction de l'arbre hiérarchique de désassemblage. Güngör et Gupta [14, 15] parlent de l'importance d'une ligne de désassemblage dans la récupération des produits. L'objectif de la ligne de désassemblage est l'utilisation efficace des ressources disponibles. Un nouveau concept est analysé, celui de l'utilisation de la ligne de désassemblage. Les objectifs, les contraintes, et le balancement des lignes sont présentés. Les éléments pertinents sont la considération du produit, la considération des composants, l'objectif du problème de balancement de la ligne de désassemblage, les contraintes du problème de balancement de la ligne de désassemblage (la présence d'échec de tâches), et l'analyse des problèmes de balancement de la ligne de désassemblage. Güngör et Gupta [14] présentent le

désassemblage d'une façon différente. Ils analysent le problème en travaillant sur une ligne de production. Ils élaborent un algorithme qui va rendre l'utilisation des ressources de la ligne de désassemblage efficiente le plus possible en rencontrant la demande. Ils s'attaquent directement aux problèmes du désassemblage. Tous les éléments présents dans les étapes de désassemblage sont évalués.

9.1 Approche d'optimisation avec présence d'échec de tâches

Considérations opérationnelles pour les lignes de désassemblage

Variation dans le temps des tâches de désassemblage : Similairement au ligne d'assemblage, les temps de tâches de désassemblage peuvent varier dépendant de quelques facteurs qui sont liés au condition du produit et l'état du poste. Les temps de tâches sont considérés comme déterministe, stochastique et dynamique. Les temps de tâches dynamiques sont possibles dus aux effets d'apprentissage, qui entraîne une réduction systématique dans les temps de désassemblage.

Départ prématuré des pièces de travail (EWP) : Si une ou plusieurs (pas toute) tâches de la pièce de travail, qui avaient été assignée au poste actuel, ne peuvent pas être complétées dues à quelques défauts, la pièce de travail peut quitter le poste prématurément.

Auto-évitement des pièces de travail (SSWP) : Si toutes les tâches de la pièce de travail, qui sont assignées au poste actuel, sont disponibles dues à sa propre défauts et/ou aux relations de priorités, la pièce de travail quitte le poste avant que le travail soit effectué dessus.

Évitement des pièces de travail (SWP) : Un poste de travail m , si un ou plusieurs échecs de tâches de la pièce de travail directement ou indirectement précède toutes les tâches du poste de travail $m+1$, la pièce de travail évite le poste $m + 1$ et va au poste $m + 2$.

Élimination de la pièce de travail (DWP) : Si un échec de tâche rend possible l'achèvement de toutes les tâches restantes au poste de travail, la pièce de travail doit simplement être enlevée de la ligne de désassemblage avant qu'elle arrive à tout autre poste de travail.

Retour de pièces de travail (RWP) : Pièce de travail actuellement au poste de travail w , pouvant revisiter le poste de travail précédant $(w-\alpha)$ où $(w-\alpha) \geq 1$ et $\alpha \geq 1$ et entier, exécution de la tâche f si l'achèvement de la tâche actuelle i permet un travail sur la tâche f qui avait originellement une assignation au poste de travail $(w-\alpha)$, et avait, encore, la possibilité d'échec d'une autre tâche précédente.

Éclatement de pièces de travail (EWP) : Une pièce de travail séparée en deux ou plusieurs pièces de travail (sous-assemblage) alors qu'elles bougent dans la ligne de désassemblage parce que le désassemblage de certains composants retient encore la pièce de travail. Chacun de ces sous-assemblages agissent comme un composant individuel dans le ligne de désassemblage.

Nomenclature : Voir **Annexe D**

Objectifs

Trouver toutes les assignations de tâche, qui minimisent le nombre de poste pour un temps de cycle donné. Comparer les assignations de tâche en utilisant une approche de probabilité pour trouver les assignations de tâche qui minimisent les effets de complication dans la ligne.

- (1) Génération d'états représentant une série ordonnée de tâches qui peuvent être effectuées sans avoir complété d'autre tâche ;
- (2) Développement d'un réseau utilisant ces étapes ;
- (3) Application de l'algorithme du plus court chemin pour trouver le nombre minimum de postes de travail requis.

Bases pour la résolution

Le temps de cycle d'une ligne de désassemblage sous le réglage du problème de balancement de ligne de désassemblage en présence d'échec de tâche (DLBP-F) s'écrit comme suit :

$$c = \frac{L}{dm_{\max}} \quad \text{où :} \quad dm_{\max} = \max_{i=1, \dots, N} \left\{ \frac{dm_i}{q_i} \right\}$$

c doit satisfaire la condition suivante : $t_i \leq c, \quad i = 1, \dots, N$

Si cette condition est satisfaite, l'assignation de tâche t_{max} , où $t_{max} > c$, n'est pas possible car les tâches sont supposées être indivisibles. Une simple modification met la valeur de c égale à t_{max} .

Le nombre minimum de poste de travail peut être trouvé comme suit :

$$M_{\min} = \left\lceil \frac{T}{c} \right\rceil = \left\lceil \frac{\sum_{i=1}^N t_i}{c} \right\rceil$$

Le nombre maximum de postes est : $M_{\max} = N$

Algorithme de la solution de l'approche générale

- (1) Génération de la matrice de priorité pour le produit (voir section 4.2.1).
- (2) Génération d'un réseau d'état incomplet (ISN) représentant tous les états faisables et leurs relations partielles. Alors, toutes les relations possibles sont développées entre les états de ISN par l'application de la contrainte du temps de cycle. Le réseau résultant étant nommé réseau d'état (SN).
- (3) Génération du poids des arcs de SN en se basant sur les temps d'inactivité des tâches à réaliser. SN avec les arcs comportant des poids est nommé réseau d'état avec poids (WSN).

(4) Application de l'algorithme du plus court chemin de Dijkstra (voir **Annexe A**) pour générer tous les chemins directs les plus courts (SDP) entre le nœud source et le nœud final de WSN. Un SDP représente une tâche d'assignation résultant du temps mort minimum de la ligne. Une tâche d'assignation est représentée par chaque SDP causant différentes complications dans la ligne de désassemblage.

L'organigramme de la figure 9.1 présente ces étapes.

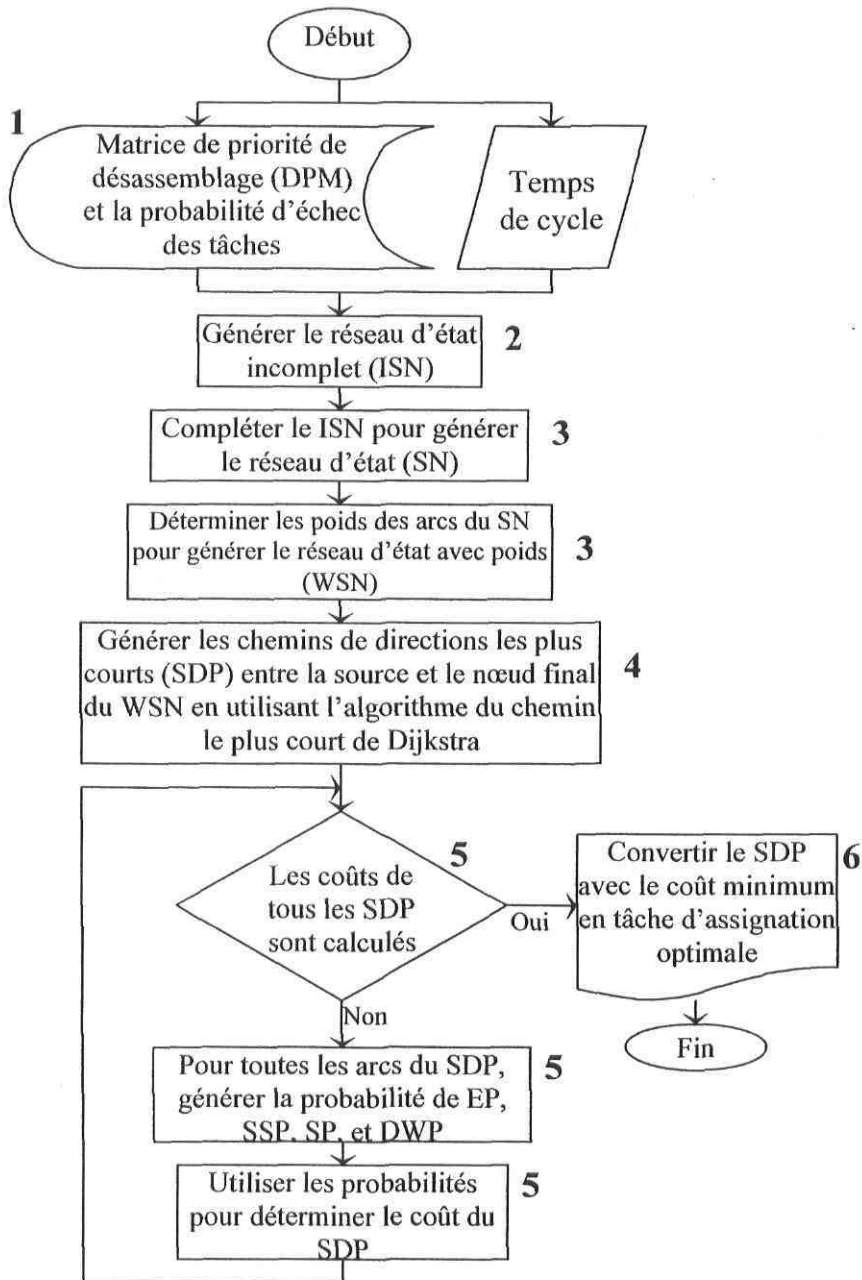


Figure 9.1 : Organigramme de l'algorithme la solution de l'approche générale

Source : Gungor et Gupta [14]

- (5) Détermination des probabilités des applications précédemment discutées (EP, SSP, SP, DWP) pour chaque arc dans SDP. Cette probabilité est utilisée dans la génération des coûts de complication des arcs de SDP considérés via la fonction de complication de coût. Nous répétons cette procédure pour tous les SDP trouvés dans WSN.
- (6) Conversion des éléments représentés par des arcs de SDP avec le coût minimum en tâche d'assignation optimum des postes de travail.

1. Réseau d'état incomplet

Un état, s_i , représente une série de tâches qui doivent être effectuées en priorité des autres tâches du DPM. Un état individuel représente la tâche d'assignation à un poste de travail.

Étapes de l'algorithme de génération de ISN :

Étape 1 : Entrée DPM. Initialiser les structures de données. Créer deux listes vides, liste de nœuds, et liste de nouveaux nœuds, qui sont dénotées par LN et NLN , respectivement. Mettre le niveau actuel (cl) à 0.

Étape 2 : Créer un nœud, (nœud source), sn , et mettre à jour cette information. Ajouter sn dans LN , Augmenter le niveau actuel par 1, $cl = cl + 1$.

Étape 3 : Si $cl \geq N$ aller à l'étape 12. Autrement, aller à l'étape 4.

Étape 4 : Si $LN = \emptyset$, copier NLN dans LN , si vide augmenter le niveau actuel de 1 et aller à l'étape 3 Autrement, aller à l'étape 5.

Étape 5 : Utiliser le premier élément de LN , fn , trouver les tâches disponibles (ET_{fn}) qui satisfont les conditions : (1) $i \notin E(fn)$, et (2) $r_{ij} = 0$, $j = 1, \dots, N$; ou

$$\exists r_{ji} = 1, j \in E(fn); \text{ et / ou } \exists r_{ji} = d, \exists j \in OG_i, d \subset E(fn).$$

Étape 6 : Si $ET_{fn} = \emptyset$ alors enlever fn de LN et aller à l'étape 4. Autrement, aller à l'étape 7.

Étape 7 : Utiliser le premier élément de ET_{fn} et, générer un état temporaire, TSE , comme suit : $TSE = E(s_{fn}) \cup et$.

Étape 8 : Si TSE égal à un des états générés précédemment au niveau actuel, aller à l'étape 10. Autrement aller à l'étape 9.

L'organigramme suivant présente l'algorithme pour générer le ISN.

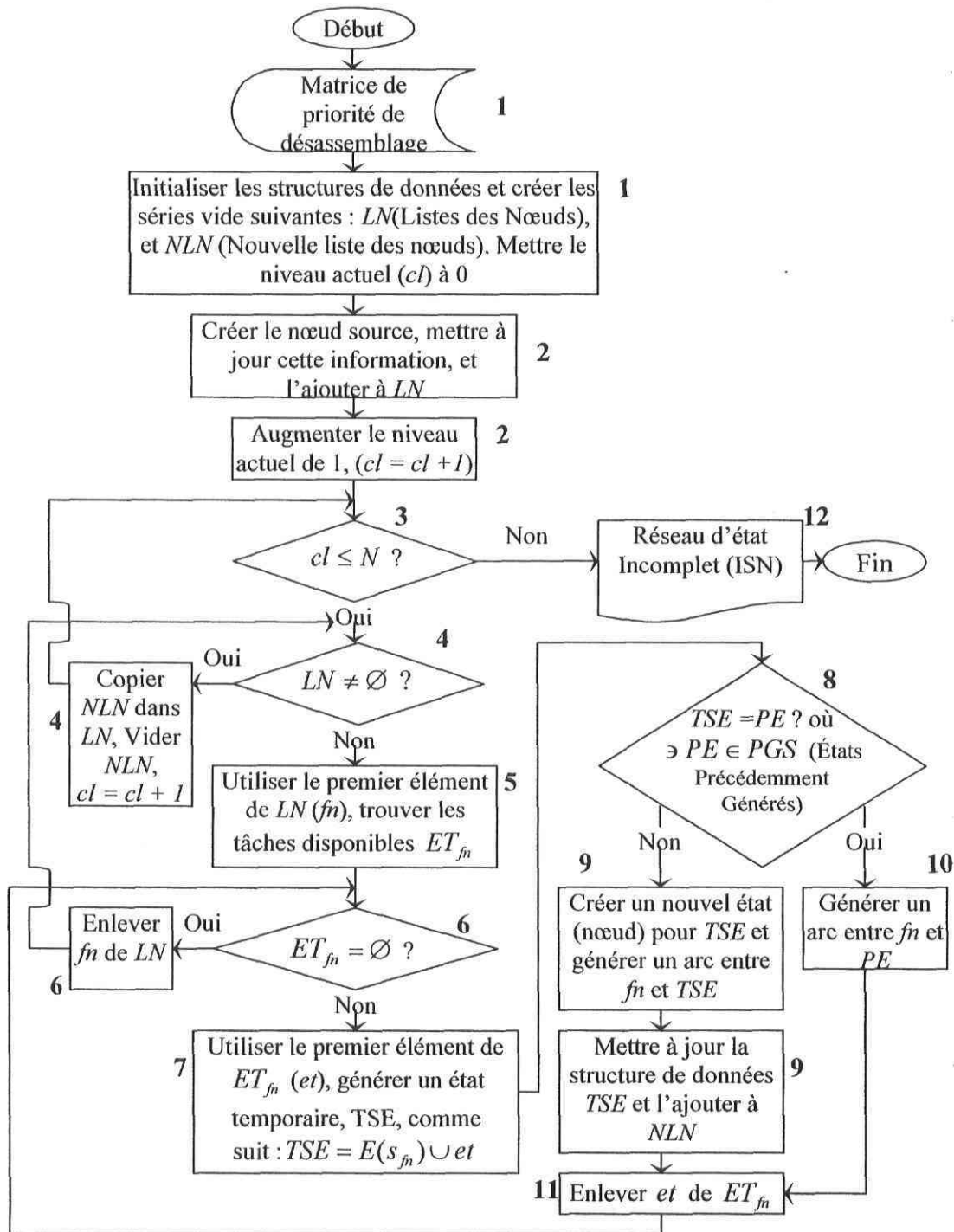


Figure 9.2 : Organigramme de l'algorithme de génération de ISN

Source : Gungor et Gupta [14]

Étape 9 : Créer un nouvel état, et générer un arc entre le fn et le nouveau nœud généré. Mettre à jour les structures de données du nouveau nœud généré et l'ajouter dans NLN . Aller à l'étape 11.

Étape 10 : Générer un arc entre fn et du nœud ayant les mêmes éléments que TSE .

Étape 11 : Enlever et de ET_{fn} , et aller à l'étape 6.

Étape 12 : Imprimer ISN. Fin.

2. Le réseau d'état (SN)

Les relations entre les états de SN sont représentées par des arcs. Il y a un arc entre les nœuds (états) a et b si et seulement si tous les arcs de ISN satisfont les conditions suivantes :

$$\left. \begin{array}{l} (1) \ E(s_a) \subset E(s_b) \\ \text{et} \\ (2) \ t(E(s_b)) - t(E(s_a)) \leq c \end{array} \right\} \quad (a)$$

Ayant ISN, la génération de plus d'arcs est réalisés en utilisant l'algorithme suivant :

Étape 1 : Entrée ISN dans le temps de cycle, c . Générer deux listes : (1) une liste contenant tous les nœuds de ISN, dénotée par LGN, et (2) une liste de magasin de nœuds de ISN qui satisfont au critère (a), dénotée par LFGN. Initialiser, les deux listes comme étant vides.

Étape 2 : Mettre tous les nœuds de ISN dans LGN.

Étape 3 : Si LGN est vide, aller à l'étape 7. Autrement, aller à l'étape 4.

Étape 4 : $testn$ est un élément dans LGN. Si $testn$ n'est pas précédemment évalué, trouver les nœuds de ISN qui satisfont aux conditions (a). Mettre les nœuds satisfaisants dans LFGN.

Étape 5 : Si LFGN est vide, enlever $testn$ de LGN et aller à l'étape 3. Autrement, aller à l'étape 6.

Étape 6 : $tempn$ est un élément de LFGN. S'il existe un arc entre $testn$ et $tempn$ dans ISN, enlever $tempn$ de LFGN et aller à l'étape 5. Autrement ajouter un arc

entre $testn$ et $tempn$ dans ISN et enlever $tempn$ de $LFGN$ et aller à l'étape 5.

Étape 7 : Imprimer SN . Fin.

L'organigramme de la figure 9.3 présente l'algorithme d'exécution de ISN pour obtenir SN .

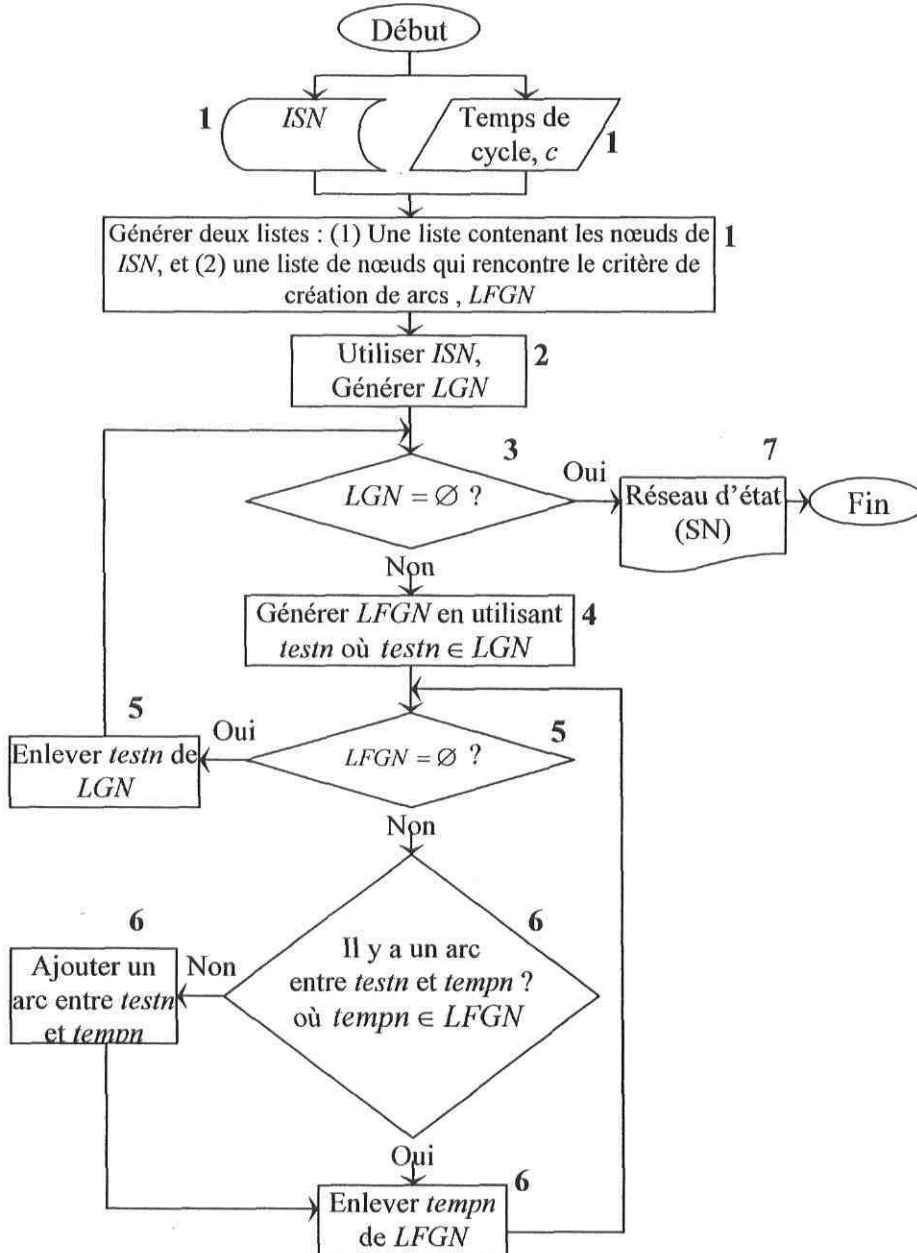


Figure 9.3 : Organigramme de l'algorithme d'exécution de ISN

Source : Gungor et Gupta [14]

3. Tâche d'assignation optimale

Pour générer la tâche d'assignation optimale en utilisant SN, nous procédons comme suit. Chaque nœud dans le SN a une structure de données associée avec cela. La structure de données consiste en une série d'éléments représentée par l'état et le temps d'état. Cette information et le temps de cycle, c , sont utilisés pour calculer le poids des arcs de SN. Le SN ayant les arcs avec poids est nommé le réseau d'état avec poids (WSN). Nous trouvons tous les chemins directs les plus courts (SDP) entre le nœud source (S) et le nœud final (F) de WSN. Ensuite, nous déterminons les probabilités d'occurrence des différentes complications des arcs de SDP. En utilisant les probabilités de complication, nous déterminons le coût de chaque SDP. Le SDP avec le coût minimum est la solution optimale de DLBP-F.

Conditions pour le calcul des probabilités

Lors du calcul du coût de chaque SDP nous devons satisfaire les conditions suivantes :

Condition 1 : Quand $M = 1$ et $|E(e_{ij})| = 1$, l'équation suivante doit être « vrai » :

$$P_{EP}(e_{ij}) = P_{SSP}(e_{ij}) = P_{SP}(e_{ij}) = 0.0$$

Condition 2 : Quand $M = 1$ et $|E(e_{ij})| \geq 2$, l'équation suivante doit être « vrai » :

$$P_{SSP}(e_{ij}) = P_{SP}(e_{ij}) = 0.0$$

Condition 3 : Quand $M = 2$ et $|E(e_{ij})| = 1$, l'équation suivante doit être « vrai » :

$$P_{EP}(e_{ij}) = P_{SP}(e_{ij}) = 0.0$$

Condition 4 : Quand $M = 2$ et $|E(e_{ij})| \geq 2$, l'équation suivante doit être « vrai » :

$$P_{SP}(e_{ij}) = 0.0$$

Condition 5 : Quand $M > 2$ et $|E(e_{ij})| = 1$, l'équation suivante doit être « vrai » :

$$P_{EP}(e_{ij}) = 0.0$$

Condition 6 : Quand $|E(s_j)| = N$, l'équation suivante doit être « vrai » :

$$P_{SSP}(e_{ij}) = P_{SP}(e_{ij}) = 0.0$$

Condition 7 : Quand $|E(s_j)| + |E(e_{jk})| = N$, l'équation suivante doit être « vrai » :

$$P_{SP}(e_{ij}) = 0.0$$

Règles pour le calcul des probabilités

Les règles suivantes sont données pour calculer les probabilités :

Règle 1 : Si $ev_i^D = \emptyset$ alors

$$P_S(ev_i) = P_{arrival}(e_{ij}) \times \left(\prod_{b \in ev_i^{ND}} (1 - p(b)) \right) \quad \text{autrement } P_S(ev_i) = 0.0$$

Règle 2 : Si $\{E(e_{ij}) - ev_i^D\} \not\subset FT(ev_i)$ et $ev_i^D = \emptyset$ alors

$$P_{EP}(ev_i) = P_{arrival}(e_{ij}) \times \left(\prod_{a \in ev_i^D} p(a) \prod_{b \in ev_i^{ND}} (1 - p(b)) \right) \quad \text{autrement } P_{EP}(ev_i) = 0.0$$

Règle 3 : Si $\{E(e_{ij}) = ev_i^D\} \subset FT(ev_i)$ alors

$$P_{SSP}(ev_i) = P_{arrival}(e_{ij}) \times \left(\prod_{a \in ev_i^D} p(a) \prod_{b \in ev_i^{ND}} (1 - p(b)) \right) \quad \text{autrement } P_{SSP}(ev_i) = 0.0$$

Règle 4 : Si $E(e_{jk}) \subset FT(ev_i)$ alors

$$P_{SP}(ev_i) = P_{arrival}(e_{ij}) \times \left(\prod_{a \in ev_i^D} p(a) \prod_{b \in ev_i^{ND}} (1 - p(b)) \right) \quad \text{autrement } P_{SP}(ev_i) = 0.0$$

Règle 5 : Si $\{AT - E(s_i) - ev_i^D\} \subset FT(ev_i)$ alors

$$P_{DWP}(ev_i) = P_{arrival}(e_{ij}) \times \left(\prod_{a \in ev_i^D} p(a) \prod_{b \in ev_i^{ND}} (1 - p(b)) \right) \quad \text{autrement } P_{DWP}(ev_i) = 0.0$$

Calcul des probabilités de e_{ij}

Ensuite, nous calculons toutes les probabilités de l'arc actuel, e_{ij} , comme suit :

$$\begin{aligned} P_S(e_{ij}) &= \sum_{ev_i \in EV(e_{ij})} P_S(ev_i), & P_{EP}(e_{ij}) &= \sum_{ev_i \in EV(e_{ij})} P_{EP}(ev_i), \\ P_{SSP}(e_{ij}) &= \sum_{ev_i \in EV(e_{ij})} P_{SSP}(ev_i), & P_{SP}(e_{ij}) &= \sum_{ev_i \in EV(e_{ij})} P_{SP}(ev_i), \text{ et} \\ P_{DWP}(e_{ij}) &= \sum_{ev_i \in EV(e_{ij})} P_{DWP}(ev_i) \end{aligned}$$

Calcul de $P_{arrival}$

Finalement, $P_{arrival}$ doit être calculé. Dans les règles le $P_{arrival}(e_{ij})$ est introduit, c'est la probabilité que la pièce de travail qui parvient au prochain poste de travail quitte le poste de travail actuel. Une pièce de travail peut arriver au prochain poste à moins qu'elle disparaisse ou saute ce poste. Donc,

$$P_{arrival}(e_{jk}) = P_S(e_{ij}) + P_{EP}(e_{ij}) + P_{SSP}(e_{ij})$$

Cependant, si les tâches sont des éléments de $E(e_{ij})$ résultant dans SSP et DWP, la probabilité $P_{SSP}(e_{ij})$ doit être exclue de $P_{arrival}(e_{ij})$.

$$P_{arrival}(e_{jk}) = P_S(e_{ij}) + P_{EP}(e_{ij})$$

Dans un autre cas, les tâches sont des éléments de $E(e_{ij})$ résultant dans SSP et SP. Bien que $P_{arrival}(e_{ij})$ soit calculé avec l'équation précédente, plus tard, $P_{SP}(e_{ij})$ va pouvoir être inclus dans le calcul de $P_{arrival}(e_{kl})$, où e_{kl} est le successeur immédiat de e_{jk} qui est le successeur immédiat de e_{ij} dans SDP.

$$P_{arrival}(e_{kl}) = P_{SP}(e_{ij}) + P_S(e_{jk}) + P_{EP}(e_{jk}) + P_{SSP}(e_{jk})$$

ou nous avons SSP et DWP dans l'arc e_{jk} , alors

$$P_{arrival}(e_{kl}) = P_{SP}(e_{ij}) + P_S(e_{jk}) + P_{EP}(e_{jk})$$

Il est à noter que quand cette approche est dans le cas de saut multiple, le calcul des probabilités d'arrivées peut être modifié. Nous avons aussi besoin d'initialiser la probabilité d'arrivée de l'arc originaire du nœud source de SDP comme suit :

$$P_{arrival}(e_{ij}) = 1.0, \text{ pour } e_{ij} \in E_{SDP} \text{ où } E(s_i) = \emptyset$$

Nous n'avons pas besoin de calculer $P_{arrival}(e_{ij})$ pour les arcs arrivants au nœud final car il y en a pas. Autre formalité, ne pas calculer $P_{arrival}(e_{ij})$ si $e_{ij} \in E_{SDP}$ et $E(s_j) = AT$.

Voici donc l'algorithme suivant qui représente la procédure pour trouver la tâche d'assignation optimale :

- Étape 1 :** Si tous SDP dans LSDP sont visités, aller à l'étape 9. Autrement, mettre le prochain SDP non visité dans LSDP comme CSDP ; mettre $P_{arrival}(e_{si})$ à 1.0, où e_{si} est un arc originaire du nœud source.
- Étape 2 :** Tous les arcs de E_{CSDP} sont visitée, aller à l'étape 8. Autrement mettre le prochain arc non visité de E_{CSDP} comme arc actuel, dénoté par e_{ij} .
- Étape 3 :** Générer $FT(a) \forall a \in E(e_{ij})$ par l'utilisation de la fonction récursive $findFT$ (figure 9.4).

```

Function findFT(R,N,a) {
    FT = ∅
    for(i = 1 to N){
        if(R[a][i] = 0){
            FT = FT ∪ i ; /* Mettre i dans FT */
            temp = findFT(R,N,i) ; /* récursivement trouver */
                /* indirectement les éléments précédents */
            FT = FT ∪ temp ; /* fusionner temp avec FT */
        }
    }
    return (FT) ;
}

```

Figure 9.4 : Fonction $findFT$

Source : Gungor et Gupta [14]

Étape 4 : Générer $EV(e_{ij})$.

Étape 5 : Déterminer $FT(ev_i) \forall ev_i \in EV(e_{ij})$ en utilisant $FT(ev_i) = \bigcup_{x \in ev_i^p} FT(x)$.

Étape 6 : Calculer les probabilités de complication $\forall ev_i \in EV(e_{ij})$. En premier, vérifier les conditions 1 à 7. Si toutes les conditions sont satisfaites, calculer les probabilités de complication en utilisant les équations appartenant aux conditions satisfaites. Si une probabilité de complication est calculée en utilisant une des conditions, ne pas exécuter les règles associées entre Règle 1 et Règle 5. Utiliser les Règles 1 à 5 pour calculer les probabilités de complications qui ne sont pas calculées en utilisant les conditions 1 à 7. Déterminer les probabilités de complication de e_{ij} en utilisant les équations. Calculer $P_{arrival}(e_{jk})$.

Étape 7 : Calculer $CCF(e_{ij})$ en utilisant :

$$CCF(e_{ij}) = (P_{EP}(e_{ij}) \times C_{EP}) + (P_{SSP}(e_{ij}) \times C_{SSP}) \\ + (P_{SP}(e_{ij}) \times C_{SP}) + (P_{DWP}(e_{ij}) \times C_{DWP})$$

Marquer e_{ij} comme étant visité et aller à l'étape 2.

Étape 8 : Calculer $CCF(CSDP)$ en utilisant : $CCF(SDP) = \sum_{e_{ij} \in E_{SDP}} CCD(e_{ij})$. Marquer

CSDP comme étant visité et aller à l'étape 1.

Étape 9 : Trouver un SDP dans LSDP où le $CCF(SDP)$ est minimum. Imprimer les éléments (tâches) représentés par les arcs de SDP comme tâche d'assignation optimale. Fin.

Solution

Donc, en appliquant l'algorithme de la figure 9.5, nous atteignons les objectifs visés par cette méthode. De façon globale nous trouvons la séquence de désassemblage optimale pour le cas de ligne de désassemblage avec présence d'échec de tâche.

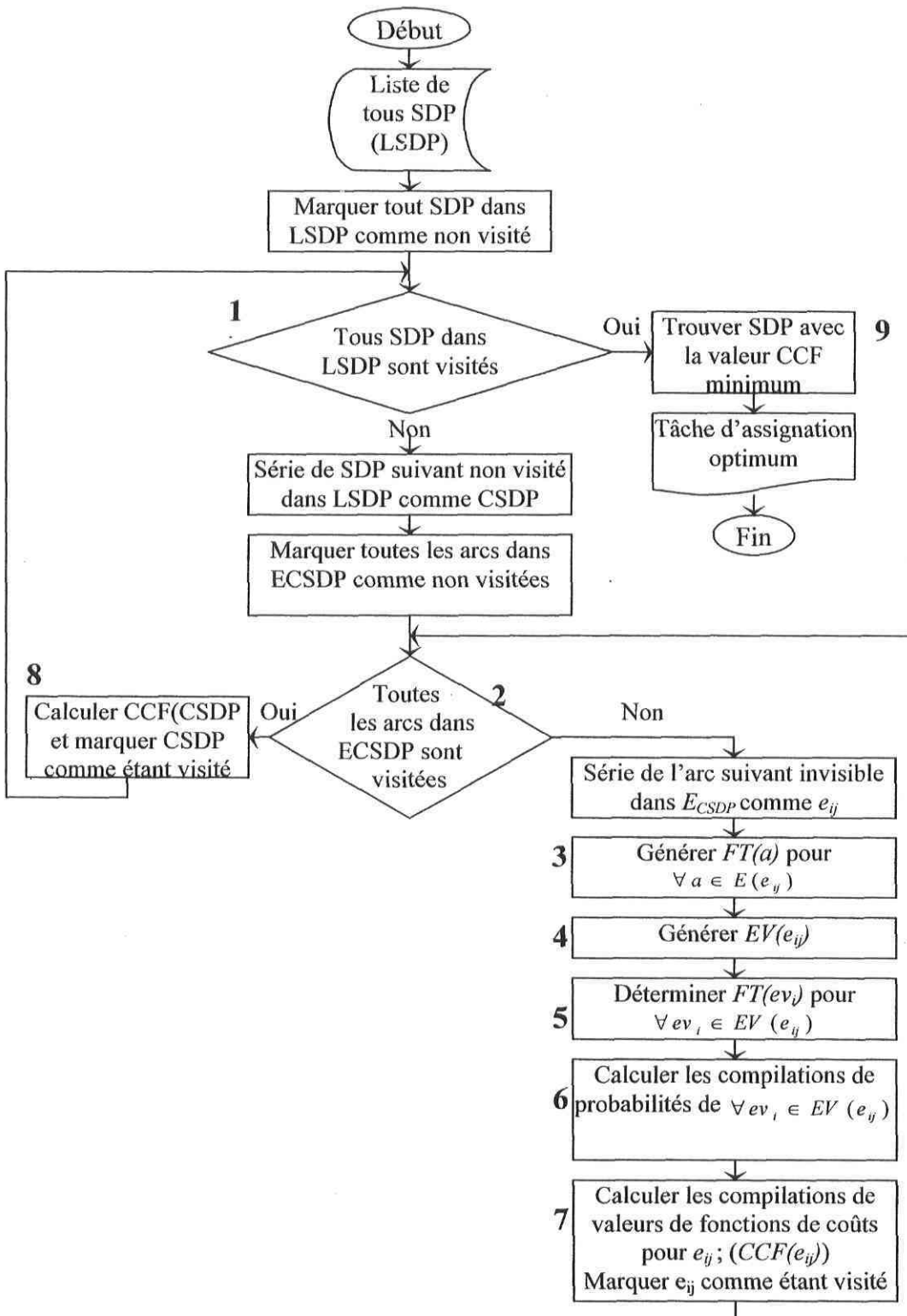


Figure 9.5 : Organigramme de l'algorithme de la tâche d'assignation optimum

Source : Gungor et Gupta [14]

9.2 Exemple d'application pour l'approche

Nous considérons un produit demandant quatre tâches pour sa séparation. La figure 9.6 illustre la matrice de priorité de désassemblage et sa représentation graphique.

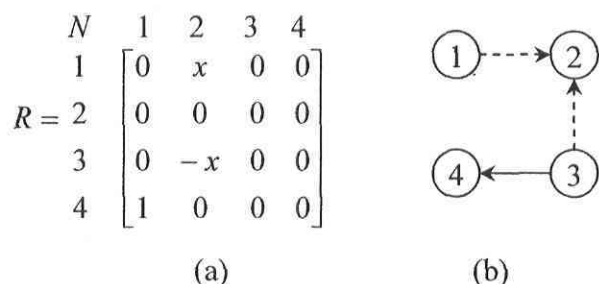


Figure 9.6 : Une matrice de priorité de désassemblage et sa représentation graphique

Source : Gungor et Gupta [15]

Nous posons le temps de cycle, $c = 15$ min. En utilisant l'algorithme ISN et SN, nous générons le ISN et SN. Les états du ISN (qui est le même pour SN) sont donnés dans le tableau 9.2. Le ISN et le SN pour l'exemple sont illustrés dans la figure 9.7(a) et la figure 9.7(b), respectivement. Après avoir généré le WSN et l'appliqué l'algorithme du plus court chemin de Dijkstra (**Annexe A**), nous obtenons sept SDP (tableau 9.3).

Afin d'illustrer le calcul des compilations de coût du SDP, nous sélectionnons aléatoirement un SDP, le SDP_6 (figure 9.8).

Tableau 9.1 : Base d'apprentissage du produit étudié [14]

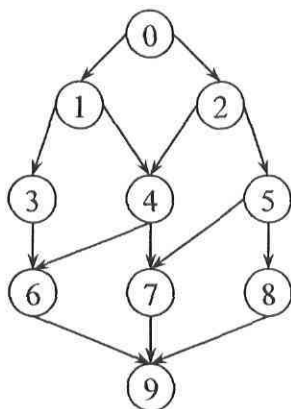
Tâche	t_i (min)	d_i (par jour)	Probabilité d'échec
1	5	8	0.7
2	10	10	0.2
3	12	12	0.1
4	8	20	0.4

Pour évaluer les Conditions 1 à 7, nous avons besoin d'identifier le nombre de machine dans la ligne, c'est-à-dire trouver la valeur de M . Nous savons que les éléments représentés par chaque arc du SDP sont les tâches assignées au poste de travail, donc le

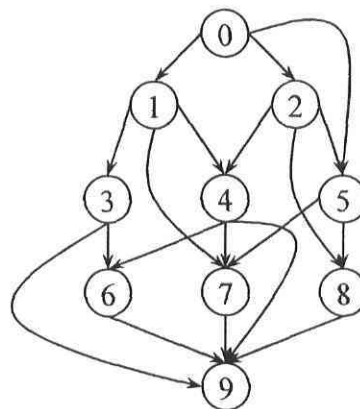
nombre d'arcs dans un SDP défini le nombre de poste de travail dans la ligne. Ainsi, dans notre cas, le nombre d'arcs du SDP₆ (figure 9.8) est 3, c'est-à-dire $M = 3$.

Tableau 9.2 : États du ISN (et SN) [14]

i	s_i	ST(s_i)
0	$\{\emptyset\}$	0
1	$\{3\}$	12
2	$\{4\}$	8
3	$\{3,2\}$	22
4	$\{3,4\}$	20
5	$\{4,1\}$	13
6	$\{3,2,4\}$	30
7	$\{3,4,1\}$	25
8	$\{4,1,2\}$	23
9	$\{3,2,4,1\}$	35



(a) ISN



(b) SN

Figure 9.7 : ISN et SN du produit étudié

Source : Gungor et Gupta [14]

Tableau 9.3 : SDP du WSN [14]

No	SDP
1	$e_{01} \rightarrow e_{13} \rightarrow e_{39}$
2	$e_{01} \rightarrow e_{14} \rightarrow e_{49}$
3	$e_{01} \rightarrow e_{17} \rightarrow e_{79}$
4	$e_{02} \rightarrow e_{24} \rightarrow e_{49}$
5	$e_{02} \rightarrow e_{24} \rightarrow e_{49}$
6	$e_{05} \rightarrow e_{57} \rightarrow e_{79}$
7	$e_{05} \rightarrow e_{58} \rightarrow e_{89}$

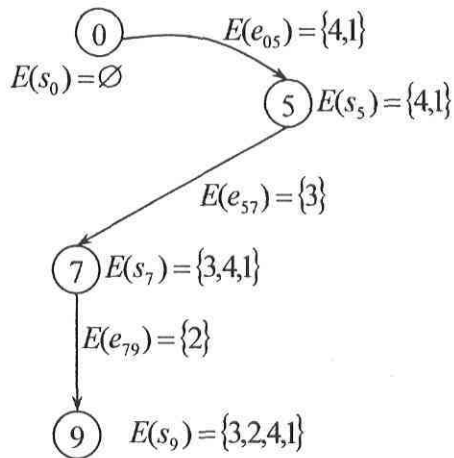


Figure 9.8 : SDP_6 du tableau 9.3

Source : Gungor et Gupta [14]

Nous pouvons maintenant trouver la tâche d'assignation optimale.

Étape 1. $CSDP = SDP_6 (e_{05} \rightarrow e_{57} \rightarrow e_{79})$

$$P_{arrival}(e_{05}) = 1.0$$

Étape 2. Pas tous les nœuds de E_{CSDP} sont visités par conséquent, le nœud actuel = e_{05} .

Étape 3. Pour $\forall a \in E(e_{05})$, générer $FT(a)$,

$a \in E(e_{05})$	$FT(a)$
4	{1,2}
1	{2}

Étape 4. $EV(e_{05}) = \{(4,1), (\bar{4},1), (4,\bar{1}), (\bar{4},\bar{1})\}$

Étape 5. En utilisant $FT(ev_i) = \bigcup_{x \in ev_i^p} FT(x)$, Nous déterminons $FT(ev_i) \forall ev_i \in EV(e_{ij})$:

$ev_i \in EV(e_{05})$	$FT(ev_i)$
(4,1)	\emptyset
($\bar{4}$,1)	{1,2}
(4, $\bar{1}$)	{2}
($\bar{4}$, $\bar{1}$)	{1,2}

Étape 6. Nous vérifions si toutes les conditions sont satisfaites. Non, parce ce que $M > 2$ et $|E(e_{05})| = 2$. Par conséquent, nous voyons les règles 1 à 5 pour déterminer les probabilités comme suit :

ev_1

$$ev_1 = (4,1); ev_1^D = \emptyset, ev_1^{ND} = \{4,1\}.$$

Règle 1 satisfaite en $ev_1^D = \emptyset$ donc en utilisant les paramètres de cette règle,

$$\begin{aligned} P_S(ev_1) &= P_{arrival}(e_{05}) \times ((1 - p(4)) \times (1 - p(1))) \\ &= 1.0 \times 0.6 \times 0.3 \\ &= 0.18 \end{aligned}$$

Règles 2 à 5 sont pas satisfaites. Par conséquent, en utilisant les paramètres de ces règles, nous trouvons que $P_{EP}(ev_1) = P_{SSP}(ev_1) = P_{DWP}(ev_1) = 0.0$.

ev_2

$$ev_2 = (\bar{4},1); ev_2^D = \{4\}, ev_2^{ND} = \{1\}.$$

Règle 1 n'est pas satisfaite puisque $ev_2^D = \emptyset$ donc en utilisant les paramètres de cette règle, $P_S(ev_2) = 0.0$.

Règle 2 n'est pas satisfaite puisque $\{E(e_{05}) - ev_2^D\} \subset FT(ev_2) = \{\{4,1\} - \{4\}\} \subset \{1,2\}$, donc $P_{EP}(ev_2) = 0.0$

Règle 3 est satisfaite puisque $\{E(e_{05}) - ev_2^D\} \subset FT(ev_2) = \{\{4,1\} - \{4\}\} \subset \{1,2\}$, donc en utilisant les paramètres de cette règle,

$$\begin{aligned} P_{SSP}(ev_2) &= P_{arrival}(e_{05}) \times (p(4) \times (1 - p(1))) \\ &= 1.0 \times 0.4 \times 0.3 \\ &= 0.12 \end{aligned}$$

Règle 4 n'est pas satisfaite puisque $E(e_{57}) \not\subset FT(ev_2) = \{3\} \not\subset \{1,2\}$, donc en utilisant les paramètres de cette règle, $P_{SP}(ev_2) = 0.0$

Règle 5 n'est pas satisfaite puisque, $\{AT - E(s_0) - ev_2^D\} \not\subset FT(ev_2) = \{1,2,3\} \not\subset \{1,2\}$,

Donc en utilisant les paramètres de cette règle, $P_{DWP}(ev_2) = 0.0$

ev_3

$$ev_3 = (4,1); ev_3^D = \{1\}, ev_3^{ND} = \{4\}.$$

Règle 1 n'est pas satisfaite puisque $ev_3^D \neq \emptyset$, donc en utilisant les paramètres de cette

règle, $P_S(ev_3) = 0.0$

Règle 2 est satisfaite puisque $\{E(e_{05}) - ev_3^D\} \subset FT(ev_3) = \{\{4,1\} - \{4\}\} \subset \{2\}$, donc en utilisant les paramètres de cette règle,

$$\begin{aligned} P_{EP}(ev_3) &= P_{arrival}(e_{05}) \times (p(1) \times (1 - p(4))) \\ &= 1.0 \times 0.7 \times 0.6 \\ &= 0.42 \end{aligned}$$

Règle 3 n'est pas satisfaite puisque $\{E(e_{05}) - ev_3^D\} \not\subset FT(ev_3) = \{\{4,1\} - \{4\}\} \not\subset \{2\}$, donc en utilisant les paramètres de cette règle, $P_{SSP}(ev_3) = 0.0$.

Règle 4 n'est pas satisfaite puisque $E(e_{57}) \not\subset FT(ev_3) = \{3\} \not\subset \{2\}$, donc en utilisant les paramètres de cette règle, $P_{SP}(ev_3) = 0.0$.

Règle 5 n'est pas satisfaite puisque $\{AT - E(s_0) - ev_3^D\} \not\subset FT(ev_3) = \{2,3,4\} \not\subset \{2\}$, donc en utilisant les paramètres de cette règle, $P_{DWP}(ev_3) = 0.0$.

ev_4

$$ev_4 = (\bar{4}, \bar{1}); ev_4^D = \{4,1\}, ev_4^{ND} = \emptyset$$

Règle 1 n'est pas satisfaite puisque $ev_4^D \neq \emptyset$ donc en utilisant les paramètres de cette règle, $P_S(ev_4) = 0.0$.

Règle 2 n'est pas satisfaite puisque $\{E(e_{05}) - ev_4^D\} \not\subset FT(ev_4) = \{\{4,1\} - \{4,1\}\} \not\subset \{1,2\}$, donc en utilisant les paramètres de cette règle, $P_{EP}(ev_2) = 0.0$

Règle 3 est satisfaite puisque $\{E(e_{05}) - ev_4^D\} \subset FT(ev_4) = \{\{4,1\} - \{4,1\}\} \subset \{1,2\}$, donc en utilisant les paramètres de cette règle,

$$\begin{aligned}
P_{SSP}(ev_4) &= P_{arrival}(e_{05}) \times (p(1) \times p(4)) \\
&= 1.0 \times 0.7 \times 0.4 \\
&= 0.28
\end{aligned}$$

Règle 4 n'est pas satisfaite puisque $E(e_{57}) \not\subset FT(ev_4) = \{3\} \not\subset \{1,2\}$, donc en utilisant les paramètres de cette règle, $P_{SP}(ev_4) = 0.0$

Règle 5 n'est pas satisfaite puisque $\{AT - E(s_0) - ev_4^D\} \not\subset FT(ev_3) = \{2,3\} \not\subset \{1,2\}$, donc en utilisant les paramètres de cette règle, $P_{DWP}(ev_4) = 0.0$

En utilisant les équations de la partie « Calcul des probabilités de e_{ij} » les probabilités sont trouvées comme suit :

$$P_S(e_{05}) = 0.18, \quad P_{EP}(e_{05}) = 0.42, \quad P_{SSP}(e_{05}) = 0.40, \quad P_{SP}(e_{05}) = 0.0, \text{ et } P_{DWP}(e_{05}) = 0.0.$$

La table suivante décrit la compilation des probabilités $\forall ev_i \in EV(e_{05})$;

$ev_i \in EV(e_{05})$	$FT(ev_i)$	Compilations des probabilités				
		$P_S(ev_i)$	$P_{EP}(ev_i)$	$P_{SSP}(ev_i)$	$P_{SP}(ev_i)$	$P_{DWP}(ev_i)$
(4, 1)	\emptyset	0.18	0.0	0.0	0.0	0.0
($\bar{4}$, 1)	{1,2}	0.0	0.0	0.12	0.0	0.0
(4, $\bar{1}$)	{2}	0.0	0.42	0.0	0.0	0.0
($\bar{4}$, $\bar{1}$)	{1,2}	0.0	0.0	0.28	0.0	0.0
	Prob.	0.18	0.42	0.40	0.0	0.0
	cumulative					

La probabilité que cette pièce de travail aille à la station suivante est trouvée comme suit :

$$P_{arrival}(e_{57}) = P_S(e_{05}) + P_{EP}(e_{05}) + P_{SSP}(e_{05}) = 0.18 + 0.42 + 0.40 = 1.0.$$

Étape 7. En utilisant l'équation $CCF(e_{ij})$ de la fonction de coût, nous déterminons la compilation de coût du nœud actuel, e_{05} , comme suit : $CCF(e_{05}) = 0.42C_{EP} + 0.4C_{SSP}$.

Nous marquons e_{05} comme étant visité et allons à l'étape 2. Note que si nous indiquons un R après l'étape cela signifie qu'elle est réparée.

Étape 2R. Les nœuds E_{CSDP} ne sont pas tous visités par conséquent, le nœud actuel = e_{57} .

Étape 3R. $\forall a \in E(e_{57})$, gènère $FT(a)$,

$a \in E(e_{57})$	$FT(a)$
3	{2}

Étape 4R. $EV(e_{57}) = \{(3), (\bar{3})\}$

Étape 5R. En utilisant l'équation de $FT(ev_i)$, nous déterminons $FT(ev_i) \forall ev_i \in \ddot{O}EV(e_{57})$;

$ev_i \in E(e_{57})$	$FT(ev_i)$
(3)	\emptyset
($\bar{3}$)	{2}

Étape 6R. Condition 5 est satisfaite puisque $|E(e_{57})| = 1$ et $M > 2$, par conséquent $P_{EP}(e_{57}) = 0.0$. Ainsi, on n'a pas à exécuter la Règle 2. La condition 7 est aussi satisfaite puisque $|E(s_7)| + |E(e_{79})| = N (= 4)$, $P_{SP}(e_{57}) = 0.0$. Ainsi, la règle 4 n'est pas à exécuter. Règles 1, 3 et 5 sont utilisées pour déterminer les probabilités comme suit.

ev_1

$$ev_1 = (3); ev_1^D = \emptyset, ev_1^{ND} = \{3\}.$$

Règle 1 est satisfaite puisque $ev_1^D = \emptyset$, ainsi en utilisant les paramètres de cette règle,

$$\begin{aligned} P_S(ev_1) &= P_{arrival}(e_{57}) \times (1 - p(3)) \\ &= 1.0 \times 0.9 \\ &= 0.9 \end{aligned}$$

Règle 3 et 5 ne sont pas satisfaites. Ainsi, en utilisant les paramètres de ces règles,

$$P_{SSP}(ev_1) = P_{DWP}(ev_1) = 0.0.$$

ev_2

$$ev_2 = (\bar{3}); ev_2^D = \{3\}, ev_2^{ND} = \emptyset.$$

Règles 1 n'est pas satisfaite puisque $ev_2^D \neq \emptyset$, ainsi en utilisant les paramètres de cette règle, $P_S(ev_2) = 0.0$

Règle 3 est satisfaite puisque $\{E(e_{57}) - ev_2^D\} \subset FT(ev_2) = \{\{3\} - \{3\}\} \subset \{2\}$, ainsi en utilisant les paramètres de cette règle,

$$\begin{aligned} P_{SSP}(ev_2) &= P_{arrival}(e_{57}) \times p(3) \\ &= 1,0 \times 0,1 \\ &= 0,1 \end{aligned}$$

Règle 5 est satisfaite puisque $\{AT - E(s_7)\} \subset FT(ev_2) = \{2\} \subset \{2\}$, ainsi en utilisant les paramètres de cette règle,

$$\begin{aligned} P_{DWP}(ev_2) &= P_{arrival}(e_{57}) \times p(3) \\ &= 1,0 \times 0,1 \\ &= 0,1 \end{aligned}$$

En utilisant les équations de la partie « Calcul des probabilités de e_{ij} » les probabilités sont trouvées comme suit :

$$P_S(e_{57}) = 0,9, \quad P_{EP}(e_{57}) = 0,0, \quad P_{SSP}(e_{57}) = 0,1, \quad P_{SP}(e_{57}) = 0,0, \text{ et } P_{DWP}(e_{57}) = 0,1.$$

La table suivante décrit la compilation des probabilités $\forall ev_i \in EV(e_{57})$;

$ev_i \in EV(e_{05})$	$FT(ev_i)$	Compilations des probabilités				
		$P_S(ev_i)$	$P_{EP}(ev_i)$	$P_{SSP}(ev_i)$	$P_{SP}(ev_i)$	$P_{DWP}(ev_i)$
(3)	\emptyset	0,9	0,0	0,0	0,0	0,0
($\bar{3}$)	$\{2\}$	0,0	0,0	0,1	0,0	0,1
	Probs. cumulative	0,9	0,0	0,1	0,0	0,1

La probabilité que cette pièce de travail aille à la station suivante est trouvée comme suit :

$$P_{arrival}(e_{79}) = P_S(e_{57}) = 0,9.$$

Étape 7R. En utilisant l'équation $CCF(e_{ij})$ de la fonction de coût, nous trouvons $CCF(e_{57}) = 0,1C_{SSP} + 0,1C_{SWP}$.

Nous marquons e_{57} comme étant visité et allons à l'étape 2.

Étape 2R. Les nœuds E_{CSDP} ne sont pas tous visités par conséquent, le nœud actuel = e_{79} .

Étape 3R. Pour $\forall a \in E(e_{79})$, génère $FT(a)$,

$a \in E(e_{79})$	$FT(a)$
2	\emptyset

Étape 4R. $EV(e_{79}) = \{(2), (\bar{2})\}$.

Étape 5R. En utilisant l'équation de $FT(ev_i)$, on détermine $FT(ev_i) \forall ev_i \in EV(e_{79})$;

$ev_i \in E(e_{79})$	$FT(ev_i)$
(2)	\emptyset
($\bar{2}$)	\emptyset

Étape 6R. Condition 5 est satisfaite puisque $|E(e_{79})| = 1$ et $M > 2$, par conséquent $P_{EP}(e_{79}) = 0.0$. Condition 6 est aussi satisfaite puisque e_{79} est le nœud final, ainsi $P_{SP}(e_{79}) = P_{DWP}(e_{79}) = 0.0$. Pour cette raison les règles 2, 4, et 5 ne seront pas à exécuter. Les règles 1 et 3 sont exécutées pour déterminer les probabilités restantes comme suit.

ev_1

$$ev_1 = (2); ev_1^D = \emptyset, ev_1^{ND} = \{2\}.$$

Règle 1 est satisfaite puisque $ev_1^D = \emptyset$, ainsi en utilisant les paramètres de cette règle,

$$\begin{aligned} P_S(ev_1) &= P_{arrival}(e_{79}) \times (1 - p(2)) \\ &= 0.9 \times 0.8 \\ &= 0.72 \end{aligned}$$

Règle 3 n'est pas satisfaite. Ainsi, en utilisant les paramètres de cette règle,

$$P_{SSP}(ev_1) = 0.0.$$

ev_2

$$ev_2 = (\bar{2}); ev_2^D = \{3\}, ev_2^{ND} = \emptyset.$$

Règle 1 n'est pas satisfaite puisque $ev_2^D \neq \emptyset$, ainsi en utilisant les paramètres de cette règle, $P_S(ev_3) = 0.0$

Règle 3 est satisfaite puisque $\{E(e_{79}) - ev_2^D\} \subset FT(ev_2) = \{\{2\} - \{2\}\} \subset \emptyset$ ainsi en utilisant les paramètres de cette règle,

$$\begin{aligned} P_{SSP}(ev_2) &= P_{arrival}(e_{79}) \times p(2) \\ &= 0.9 \times 0.2 \\ &= 0.18 \end{aligned}$$

En utilisant les équations de la partie « Calcul des probabilités de e_{ij} » les probabilités sont trouvées comme suit :

$$P_S(e_{79}) = 0.72, \quad P_{EP}(e_{79}) = 0.0, \quad P_{SSP}(e_{79}) = 0.18, \quad P_{SP}(e_{79}) = 0.0, \text{ et } P_{DWP}(e_{57}) = 0.0.$$

La table suivante décrit la compilation des probabilités $\forall ev_i \in EV(e_{79})$;

$ev_i \in EV(e_{05})$	$FT(ev_i)$	Compilations des probabilités				
		$P_S(ev_i)$	$P_{EP}(ev_i)$	$P_{SSP}(ev_i)$	$P_{SP}(ev_i)$	$P_{DWP}(ev_i)$
(2)	\emptyset	0.72	0.0	0.0	0.0	0.0
($\bar{2}$)	\emptyset	0.0	0.0	0.18	0.0	0.0
	Probs. cumulative	0.72	0.0	0.18	0.0	0.0

Puisque e_{79} est le nœud final, la probabilité d'arrivée de la station suivante n'est pas calculée.

Étape 7R. En utilisant l'équation $CCF(e_{ij})$ de la fonction de coût, c'est-à-dire $CCF(e_{79}) = 0.18C_{SSP}$.

Nous marquons e_{79} comme étant visité et allons à l'étape 2.

Tableau 9.4 : Compilation des coûts de tous les SDP [14]

i	SDP _i	CCF(SDP _i)
1	$e_{01} \rightarrow e_{13} \rightarrow e_{39}$	$0.42C_{EP} + 0.28C_{SSP} + 0.1C_{SP} + 0.4C_{DWP}$
2	$e_{01} \rightarrow e_{14} \rightarrow e_{49}$	$0.06C_{EP} + 0.5C_{SSP} + 1.1C_{DWP}$
3	$e_{01} \rightarrow e_{17} \rightarrow e_{79}$	$0.42C_{EP} + 0.5C_{SSP} + 1.0C_{DWP}$
4	$e_{02} \rightarrow e_{24} \rightarrow e_{49}$	$0.06C_{EP} + 0.5C_{SSP} + 0.7C_{DWP}$
5	$e_{02} \rightarrow e_{28} \rightarrow e_{89}$	$0.036C_{EP} + 0.92C_{SSP} + 0.4C_{SP} + 0.1C_{DWP}$
6	$e_{05} \rightarrow e_{57} \rightarrow e_{79}$	$0.42C_{EP} + 0.68C_{SSP} + 0.1C_{DWP}$
7	$e_{05} \rightarrow e_{58} \rightarrow e_{89}$	$0.42C_{EP} + 0.62C_{SSP} + 0.92C_{DWP}$

Étape 2R. Puisque tous les arcs de ECSDP sont visités, nous allons à l'étape 8.

Étape 8. En utilisant l'équation de CCF(SDP), nous trouvons la compilation de coût de CSDP comme suit :

$$CCF(SDP_6) = 0.42C_{EP} + 0.68C_{SSP} + 0.1C_{DWP}.$$

Nous marquons CSDP comme étant visité et nous allons à l'étape 1 pour trouver le coût de la SDP suivante, et ainsi de suite.

Nous répétons la procédure suivante pour tous SDP et WSN. Nous arrêtons quand nous avons analysé tous les SDP. Donc on va à l'étape 9.

Étape 9. Le tableau 9.4 présente la compilation de coûts de tous le SDP du WSN. Supposer que les coûts de compilations sont donnés comme suit : $C_{EP} = C_{SSP} = C_{SP} = C_{DWP} = 10\$$. En d'autres mots, chaque compilation influence la ligne de désassemblage de la même manière. Quand nous calculons les coûts de SDP, nous trouvons qu'il y a deux SDP (SDP_1 et SDP_6) avec le plus bas coût de 12\$ chaque. Ainsi, les deux meilleurs assignations peuvent être écrites en utilisant les arcs de SDP_1 et SDP_6 . En utilisant SDP_1 , nous trouvons $A_1 = \{3\}, A_2 = \{2\}, A_3 = \{4,1\}$; et en utilisant SDP_6 nous trouvons $A_1 = \{4,1\}, A_2 = \{3\}, A_3 = \{2\}$; FIN.

10 Lien entre la conception et le désassemblage de produit

Pour fermer la boucle sur les stratégies d'assemblage et de désassemblage dans un contexte de développement durable, il est important d'apporter des précisions sur la conception. Tout d'abord le retour sur la conception donne une nouvelle dimension au principe de désassemblage. Après avoir passé par tout le cycle de vie du produit et de son désassemblage, il est maintenant temps de revenir sur la conception. Nous voyons s'il est possible d'apporter des modifications aux éléments de conception pour faciliter tout le processus par lequel ont été passé. Donc l'objectif du retour sur la conception est de rendre les produits plus facile à désassembler. Le développement « durable » vient alors prendre toute son importance.

Ce retour sur la conception est très important mais le plus important du développement durable est de prendre en considération le désassemblage dès la conception du produit, avant de passer par tout le cycle de vie du produit (section 2.2). Il existe des outils et des guides pour réaliser ce type de conception.

10.1 Analyse de la conception pour aider le désassemblage

10.1.1 Analyse du désassemblage pour un retour sur la conception

Viswanathan et Allada [52] présentent des outils pour permettre de faire de la conception verte en analysant le désassemblage de fin de vie des produits. Leur approche donne à la conception les outils de base pour faciliter le désassemblage du produit. Contrairement à la plupart des autres travaux sur l'optimisation du désassemblage pour des produits déjà en fin de vie, ils proposent une analyse du désassemblage dès la conception. L'étude se

concentre sur l'analyse de la configuration des produits et des améliorations possibles. Les outils sont tout de même sensiblement les mêmes que ceux présentés dans d'autres travaux. Ils présentent un calcul pour les stratégies de fin de vie et une analyse de la configuration par les graphiques ET/OU et le concept de pétition.

La section suivante comprend les implications de la configuration du produit sur des procédures de désassemblage de fin de vie. La configuration du produit est relative à l'arrangement spatial et logique de différentes parties du produit. Une base de travail formelle pour supporter ces arguments est présentée par une méthode de détraction basée sur l'information de conception de cette base de travail. Cela aide la configuration du retour sur conception dans le but de supporter la conception pour le désassemblage (DFD).

Objectifs

- 1) Réduction des coûts de chacune des opérations de désassemblage ;
- 2) Augmentation de la valeur des segments de produits ;
- 3) Réarrangement de la valeur et des coûts des éléments de la séquence.

Hypothèses

- 1) La description complète de la configuration et des valeurs des différents quotients, est disponible a priori.
- 2) Seulement des désassemblages non destructifs.
- 3) Après chaque opération de désassemblage un maximum de z quotients sont formés.
- 4) En développant un modèle de valeur de configuration (CV), nous supposons que l'identification de groupes préliminaires de composants pouvant être assemblés et doivent être traités comme un composant unique.
- 5) Le désassemblage est réalisé par un seul intervenant.

Modèle : valeur de configuration (CV)

Espace de séquence

Q est l'ensemble des quotients m qui ont besoin d'être assemblés. Une séquence de désassemblage ou Q peut être exprimé comme $S = (Q, <)$, où $x < y$ implique que le quotient atomique y est enlevé en priorité du composant x, pour tout x et $y \in Q$. S est une chaîne due à l'hypothèse que seulement un composant peut être enlevé à chaque étape.

La série, Ω_Q est définie comme la série de toutes les séquences possibles de Q, c'est-à-dire $\Omega_Q = \{S_1, S_2, S_3, \dots, S_n\}$ où S_i représente la i^e séquence et n est le nombre total de séquences faisables. Donc, $\Omega_Q \subseteq \Omega$, ou Ω est la série de toutes les séquences faisables ou infaisables possibles pour la même série de quotients m où le désassemblage est total. Donc $\Omega_Q = \{S_1, S_2, S_3, \dots, S_k\}$ où $k = \text{factoriel}(m)$ ou $m!$.

La distance entre n'importe quelles deux séquences $\delta(S_i, S_j)$ peut être définie comme la cardinalité de la différence $(S_i - S_j)$. Donc, si $S_i = a < b < c < d$ et $S_j = a < c < b < d$ quand $\delta(S_i, S_j) = 1$ comme une différence d'une ordre de paire $b < c$. Ça peut aussi être vue quand $\delta(S_i, S_j) = \delta(S_j, S_i)$.

La série Ω peut être conceptualisée, comme un graphique non directionnel ou un arc existe entre n'importe quel z séquence, S_i et S_j si $\delta(S_i, S_j) = 1$ ou $S_i, S_j \in \Omega$. Chaque vecteur du graphique a donc $(m-1)$ voisins où m est le nombre de quotient dans l'ensemble Q. Le graphique ainsi construit, $G(\Omega)$, consiste à un composant de connexion simple. Cela implique qu'il existe au moins une chaîne entre n'importe quelle z séquences. S_i et S_j , ou $S_i, S_j \in \Omega$. De toutes les chaînes qui existent entre S_i et S_j , il existe au moins une chaîne mono tonique, c'est-à-dire $\delta(S_i, S_x) + \delta(S_j, S_x) = \delta(S_i, S_j) \forall S_x$ dans la chaîne.

Idéalement, Ω_Q pour un assemblage est la série de toutes les séquences de désassemblage qui répondent à l'hypothèse qu'il n'existe aucune contrainte dans les directions d'enlèvement du quotient. Les contraintes existantes affectent directement Ω_Q .

L'effet des contraintes sur Ω_Q est illustré dans la figure 10.1.

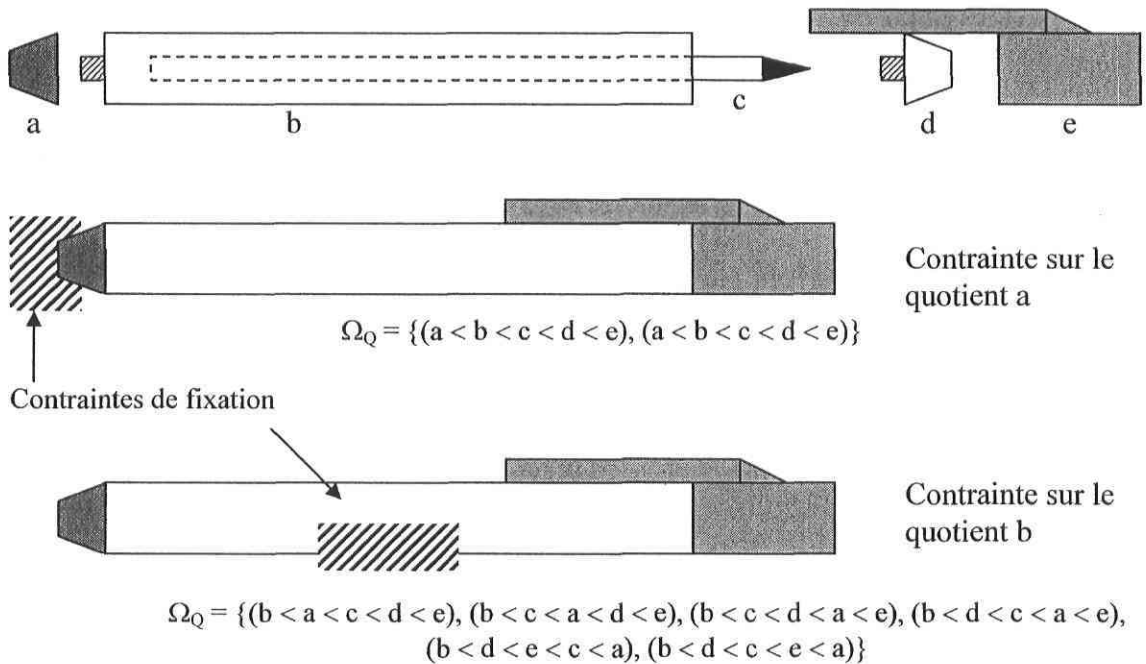


Figure 10.1 : Effet des contraintes sur Ω_Q

Source : Viswanathan et Allada [52]

Écriture du concept

Une séquence de désassemblage, où la valeur des quotients plus élevés peut être extrait en premier dans le désassemblage, est très souhaitable. D'où une séquence idéale voulant être le ou les quotients devant être enlevés dans l'ordre descendant de leur valeur. Ce VP est nommé, la valeur de priorité désirée (DVP) ou S_{dvp} . S_{dvp} appartient à Ω mais n'appartient pas à Ω_Q .

Si deux quotients n'ont pas la même valeur alors S_{dvp} est unique. Comme S_{dvp} est la séquence idéale n'importe quelle séquence S_x ou $\delta(S_{dvp}, S_x) > 0$ est donc non-idéale.

Dans l'ordre pour mieux quantifier l'étendue du changement de direction d'une séquence de S_{dvp} , le concept de pétition est introduit. Toute paire ordonnée dans la séquence S qui diffère de S_{dvp} est nommée une pétition. Les pétitions sur la séquence S peuvent être définies comme un $T_s = (Q, <_{\psi})_s$ où $b <_{\psi} a$ implique que :

- $b < a$
- Quotient $b(V_b)$ est plus grand que le quotient $a(V_a)$, où $a, b \in Q$

Chaque pétition est assignée à une valeur p_{ab} ($a <_{\psi} b$, a et $b \in Q$)

$$p_{ab} = T_1 e^{T_2}$$

Où :

$$T_1 : (V_a - V_b) / (V_{\max} - V_{\min}) ;$$

$$T_2 : (V_a - V_{\min}) / (V_{\max} - V_{\min}) ;$$

V_{\max} : valeur maximum du quotient dans S ;

V_{\min} : valeur minimum du quotient dans S ;

$$V_{\max} \neq V_{\min}.$$

Le premier terme (T_1) est la mesure de la différence de valeur qui cause la pétition. Le terme (e^{T_2}) amplifie cette différence de valeur comme une fonction des valeurs de l'évolution des quotients. D'où, si les pétitions p_{ij} et p_{ab} ont un T_1 égal, alors $p_{ij} > p_{ab}$ est vrai seulement si $V_i > V_a$. La magnitude nette, p_s est la somme des magnitudes de pétition sur T de la séquence S . Donc, pour chaque deux séquences, S_x et S_y , si $P_x > P_y$ alors S_x n'est pas égal à une plus grande extension que S_y . Une façon intuitive la représentation de la relation entre la séquence et la pétition est illustrée par un diagramme de pétition. Ce

dernier est une amélioration du diagramme de Hasse pour la séquence obtenue par un arc du quotient b à a si $b <_{\psi} a$ où a et $b \in Q$.

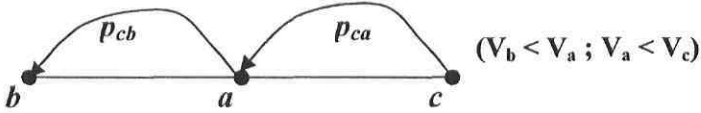


Figure 10.2 : Exemple de diagramme de pétition

Source : Viswanathan et Allada [52]

Finalement, pour résumer, l'application du concept de pétition permet le traçage de VP de toute séquence S dans un seul vrai nombre évalué P_S .

Configuration du retour sur la conception

La valeur optimale de priorité (OVP) où S_{ovp} est définie comme la séquence de Ω_Q ayant la plus haute magnitude de U_{avg} c'est-à-dire la moyenne de U sur la séquence entière. La raison pour utiliser U_{avg} est que la profondeur du désassemblage ne peut pas servir comme une base pour l'analyse étant donné qu'il doit observer une variation après chaque modification de la conception de l'assemblage.

S_{ovp} n'est pas égale à S_{dvp} . Toutefois l'objectif idéalisé pour la configuration du retour sur la conception peut être établi comme la modification de la configuration de Q s'il remplit la condition suivante $S_{ovp} = S_{dvp}$.

La séquence ayant le plus basse magnitude de pétition dans Ω_Q n'est pas nécessairement S_{ovp} . Cette séquence dans Ω_Q ayant la plus basse magnitude de pétition est nommée, la meilleur valeur de priorité (BVP) ou S_{bvp} . Quand $S_{dvp} \in \Omega_Q$, $S_{bvp} = S_{dvp}$ et quand $S_{dvp} \notin \Omega_Q$, S_{bvp} est tel qu'il n'existe pas d'autre séquence S_i dans Ω_Q tel que $\delta(S_i, S_{dvp}) < \delta(S_{bvp}, S_{dvp})$.

L'ensemble des séquences, S_{dvp} , S_{bvp} , S_{ovp} servant à caractériser toutes configurations données. S_{dvp} peut être considéré pour définir un point de référence global pendant que S_{bvp} définit un point de référence local pour S_{ovp} . De cette interprétation, l'objectif de mettre $S_{ovp} = S_{dvp}$ est réalisé par les deux étapes suivantes :

- 1) Augmentation de la similarité entre S_{bvp} et S_{dvp}
- 2) Augmentation de la similarité entre S_{ovp} et S_{bvp}

L'accomplissement de ces objectifs se présente sous différentes formes. D est une action de conception qui agit dans l'assemblage Q pour le changement en Q' c'est-à-dire $D : Q \rightarrow Q'$

- 1) Si le changement D est dans S_{ovp} pendant que Ω_Q , S_{dvp} , S_{bvp} et S_{ovp} sont changés cela implique une modification dans l'effet du coût des éléments. Si les distances $\delta(S_{ovp}, S_{dvp})$ dans Q' sont perdues seulement dans Q et ${}^Q U_{avg} < {}^{Q'} U_{avg}$ nous avons une amélioration de la configuration.
- 2) Si S_{dvp} change, mais Ω_Q reste inchangé, cela implique une modification de la valeur du quotient pendant que le reste de la configuration de l'assemblage n'est pas affecté. La variation dans S_{dvp} peut être accomplie par un changement dans S_{bvp} et S_{ovp} .
- 3) Si l'action de conception résulte de la modification Ω_Q nous pouvons nommer cela une modification active. Cela est dû à (a) un changement dans le nombre de quotient ou à (b) un changement dans les relations spatiales entre les quotients avec Ω demeurant inchangé.

L'action actuelle de conception peut être une combinaison de modifications passives et actives. Les modifications précédentes sont toutes des modifications actives.

10.1.2 Retour sur la conception pour le recyclage

Pneli et Zussman [38] présentent un algorithme pour calculer les valeurs de fin de vie (EOL) des produits pour déterminer des solutions optimales aux options de récupération. Ils utilisent un algorithme pour identifier les points faibles dans la conception, et un outil pour permettre la génération automatique d'amélioration de valeur EOL. Ils voient de façon différente l'analyse économique réalisée par Johnson et Wang [20, 21]. Ils travaillent avec les graphiques de recouvrement avec évaluation des coûts directement sur le graphe. Johnson et Wang [21] utilisent la matrice de coûts en combinaison avec les arbres de désassemblage. Ils suggèrent un recyclage orienté vers une approche de réingénierie, utilisant un concept de plan de récupération basé sur une analyse quantitative de la valeur EOL d'un produit. Cette aide à un avantage sur les autres travaux parce qu'elle offre le choix des outils pour intégrer les facteurs de fin de vie (EOL) dans la conception des produits.

Retour sur la conception pour le recyclage

Après avoir évalué les valeurs de fin de vie de notre produit (section 3.3), nous pouvons nous attaquer au problème de la conception pour le recyclage. Cette dernière nous permet de faire la conceptualisation du produit dans une perspective d'amélioration des valeurs de fin de vie. Notre approche de résolution consiste en une méthode d'analyse prenant en considération les spécifications environnementales désirées. Elle se réalise en quatre étapes [38] :

- 1) Évaluation de $d(p)$: noter quels sous-ensembles sont utilisables dans l'optimisation des procédures et décider quand sortir de la boucle (si $d(p)$ est trop élevé ou quand la marge de profit après l'itération précédente est négligeable) ;
- 2) Identification des points faibles : aspects de la conception où nous nous attendons à ce que l'utilité de faire une modification soit importante;
- 3) Génération d'indicateur : modification de la conception pour augmenter la valeur EOL ;

- 4) Vérification des indications : assurer les modifications de la conception en prenant en considération toutes les fonctions et les autres contraintes du produit. Cette étape nous permet de faire un retour sur les étapes 2 et 3.

L'étape 1 est étudiée dans les sections précédentes par l'analyse de la structure du produit et de nos besoins. L'étape 4 est, tout simplement un retour sur les étapes 2 et 3. Alors, dans les pages qui vont suivre nous allons présenter les détails de ces deux étapes : l'identification des points faibles et la génération d'indicateurs pour une modification de la conception.

Identification des points faibles de la conception

Intuitivement, les points faibles sont des aspects de la conception où nous nous attendons à ce que l'utilité de faire une modification soit importante. Donc à priori tout aspect de la conception peut être un point faible. Il existe très peu de règles formelles pour le retour sur la conception. Seul les expériences antérieures nous permettent d'en faire l'évaluation. Le sommaire suivant est une énumération des principaux critères fréquemment cités [38] :

Critère 1 : *Regarder individuellement les composants*

- a) Éviter les matériaux hasardeux et tous autres nuisibles ;
- b) Éviter les matériaux et composants incompatibles avec des procédures standards de récupération.

Critère 2 : *Regarder les sous-ensembles*

- a) Les groupes de matériaux avec un respect de la compatibilité d'utilisation.

Critère 3 : *Regarder les opérations de désassemblage*

- a) Utiliser des éléments qui ont des joints faciles à désassembler entre sous-ensembles pour des matériaux incompatibles ;
- b) Conception facilitant l'accessibilité des éléments joints ;

- c) Utiliser le moins possible d'éléments joints (fixations) ;
- d) Protéger les éléments joints de la corrosion et de la pollution ;
- e) Éviter les techniques nécessitant un désassemblage destructif si il y a des arêtes dangereuses (parties coupantes ou pointues) entrant dans le produit.

Critère 4 : *Regarder le produit dans son ensemble*

- a) Minimiser la variété des matériaux ;
- b) Minimiser le nombre et la variété des éléments de joint ;
- c) Rendre les composantes/matériaux ayant des valeurs résiduelles élevées , facilement accessible;
- d) Éviter les relations de priorités complexes entre les composants.

Critère 5 : Recyclage pertinent mais non inclus dans notre optimisation

- a) Permettre l'utilisation de matériaux/composants/sous-assemblages recyclés;
- b) Fournir de l'information provenant du recyclage : les matériaux récupérables, les procédures de désassemblage et les options possibles de recyclage.

Pour avoir plus de détails sur des aides au retour sur la conception nous pouvons nous référer aux besoins et solutions de la section 10.2.

Cette heuristique indépendante du produit, nous permet de former une définition des points faibles, basée sur le graphique de recouvrement. Les points faibles sont des nœuds v où nous avons une des situations suivantes [38] :

1. Heuristiques Groupe 1 : v est un composant avec une valeur EOL distinctive faible.
2. Heuristiques Groupe 2 et 3 : v est un composant avec une valeur EOL distinctive très faible. Telle la somme des valeurs de EOL pour tous les fils de v

dans tous les groupes ET. C'est une indication de joints complexes ou d'un groupe de matériaux incompatibles.

3. Heuristiques Groupe 4 : v est un sous-ensemble avec une signification haute ou basse valeur de recouvrement et le coût de désassemblage des composants est significativement élevé.

Des situations suivantes nous pouvons former un algorithme pour l'identification automatique des points faibles (figure 10.3).

Étape 1 :	<p>Pour tous les v libres dans le graphique de recouvrement {composant}</p> <p>Si $(c(v) < t_1)$, alors v marque un point faible ;</p>
Étape 2 :	<p>Pour tous les μ non libres dans le graphique de recouvrement</p> <p>$delta = \max\{(d(v) + d(\omega)) - d(\mu)\}$ (Où nous avons un maximum pour tous les groupes)</p> <p>ET de μ et v, ω sont des nœuds de chaque groupe ET spécifique)</p> <p>Si $(delta > t_2)$, alors μ est marqué comme un point faible ;</p>
Étape 3 :	<p>Marquage de tous nœuds v significatifs dans le graphique de recouvrement avec $c(v) > t_3$;</p> <p>Faire le graphique incluant seulement les nœuds significatifs ;</p> <p>Récurivement remonter dans le graphique</p> <ol style="list-style-type: none"> a) Tous les sorties et arcs non significatifs b) Tous les nœuds avec un seul arc sortant <p>{pour ce nœud v et $\mu \rightarrow v$, $v \rightarrow \omega$ arcs dans le graphique. Après, dessiner l'arc en deux parties $\mu \rightarrow \omega$ devient un arc avec $c(\mu, \omega) = c(\mu, v) + c(v, \omega)$ }</p> <p>À partir de ce nouveau graphique utiliser l'algorithme de l'Étape 2</p>

Figure 10.3 : Proposition d'un algorithme d'identification des points faibles

Source : Pneli et Zussman [38]

Stade d'optimisation du retour sur la conception

Trouver une série de règles et de sous-routines pour une vérification locale, de chaque nœud v du graphique de recouvrement, nous pouvons calculer $h(v)$, la meilleur indication faisable pour v , et $c_n(v)$, où la meilleur indication est donné comme suit :

$h(v) := \text{Identity transformation ;}$

$c_n(v) := c(v) ;$

Pour chaque rôle r des arcs de transformation permis :

Si (r relève de v) et

(r n'apparaît pas dans la liste d'indication défendue de v) et

($r(v)$ passe le test local de vérification, alors commencer { r est faisable pour v }

$c_r(v) :=$ la valeur c de v si r est implanté dans v

Si $c_r(v) > c_n(v)$, alors

Mettre :

$c_n(v) := c_r(v) ;$

$h(v) := r(v) ;$

Fin ;

Fin.

10.2 Conception environnementale

La conception pour le désassemblage est aussi nommée conception environnementale ou conception verte. Elle est un élément important du développement durable. nous pourrions même dire qu'elle en est à la base. Tous les facteurs environnementaux directement liés au produit, en découle. Il est donc impossible de parler de stratégies de désassemblage sans donner toute l'information possible pour effectuer la conception pour le désassemblage. Son utilisation permet de simplifier les méthodes de désassemblage et la structure du produit. Elle minimise le nombre de composants non recyclable et facilite toutes les interventions sur le produit lors du désassemblage. Il est intéressant de faire un retour sur la conception (section 10.1) mais l'idéal est d'appliquer cette conception dès le départ. À court terme elle peut s'avérer très restrictive mais peut avoir une grande importance si notre produit passe par toute la chaîne du cycle de vie (section 2.2). Ce type de conception est aussi très important pour assurer la sécurité de l'intervenant sur le produit, élément normalement très peu considéré dans les stratégies de désassemblage. La conception pour le désassemblage n'est pas seulement bonne pour l'environnement, elle permet de rendre le travail de désassemblage sécuritaire.

Voici donc un outil complet et visuel de conception environnementale qui se nomme « Conception axiomatique » (axiomatic design) [4]. Il se lit de façon hiérarchique, et inclut le besoin (B) versus la solution de conception (SC). La figure 10.3, présente la hiérarchie pour les solutions de conception. Les jonctions U représentent une relation détachée et les jonctions D représentent une relation découplée. Pour la jonction U le parent SC est réalisé pour l'implantation de tous ces enfants SC en même temps ou dans n'importe quel ordre. Pour la jonction D, le parent SC est réalisé par l'implantation de tous ses enfants dans un ordre particulier (de gauche à droite dans la figure 10.4 et de haut en bas dans le tableau 10.1). Les détails sur les besoins et la solution de conception associée sont les suivants.

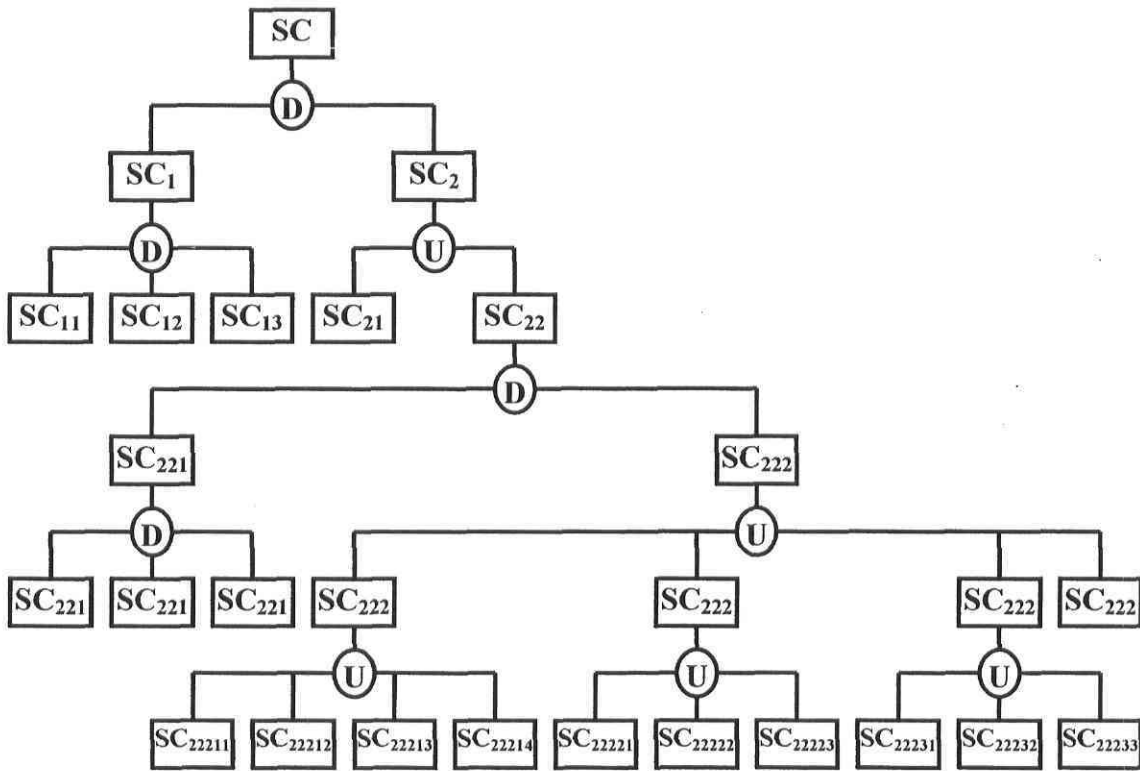


Figure 10.4 : Hiérarchie des solutions de conception

Source : Chen [4]

B₁	Peu de composants ou faible volume de matériaux dans un produit
B₂	Recyclage plus économique pour tous les composants dans le produit jeté

SC₁	Réduire le nombre de composants ou le volume de matériaux utilisé dans le produit
SC₂	Conception de produits qui peuvent être recyclés plus économiquement

$$\begin{Bmatrix} B_1 \\ B_2 \end{Bmatrix} = \begin{bmatrix} X & 0 \\ X & X \end{bmatrix} \begin{Bmatrix} SC_1 \\ SC_2 \end{Bmatrix} \quad \begin{array}{l} SC_1 \text{ doit être sélectionné pour la réalisation de } B_1; \\ SC_2 \text{ doit être sélectionné pour la réalisation de } B_2 \text{ mais n'affecte pas } B_1. \end{array}$$

B₁₁	Moins de matériaux dans un produit
B₁₂	Pas de composants non nécessaires pour chaque phase du cycle de vie dans le produit
B₁₃	Pas de volume de matériaux non nécessaire utilisé dans chaque composant nécessaire

SC₁₁	Conception de produits ayant le plus d'information possible pour le remplacement des matériaux
------------------------	--

SC₁₂	Éliminer les composants non nécessaires pour tout le cycle de vie du produit, ou fusionner le avec d'autres composants nécessaires dans le produit
SC₁₃	Éliminer le volume de matériaux non nécessaire utilisé dans chaque composant critique

$$\begin{cases} B_{11} \\ B_{12} \\ B_{13} \end{cases} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{cases} SC_{11} \\ SC_{12} \\ SC_{13} \end{cases}$$

SC₁₁ doit être sélectionné pour la réalisation de B₁₁; SC₁₂ doit être sélectionné pour la réalisation de B₁₂ mais n'affecte pas B₁₁; SC₁₃ doit être sélectionné pour la réalisation de B₁₃ mais n'affecte pas B₁₁ et B₁₂.

B₂₁	La plus haute valeur du matériel recyclé de tous les composants dans le produit
B₂₂	Le plus facile ou plus économique désassemblage des composants dans le produit

SC₂₁	Utiliser le matériel qui peut être réutilisé pour le plus d'applications possibles et ayant la plus grande demande avec la plus basse différence de prix entre le matériel vierge et le recyclé
SC₂₂	Conception du produit créant une facilité de séparation ou de désassemblage

$$\begin{cases} B_{21} \\ B_{22} \end{cases} = \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} \begin{cases} SC_{21} \\ SC_{22} \end{cases}$$

SC₂₁ doit être sélectionné pour la réalisation de B₂₁ mais n'affecte pas B₂₂; SC₂₂ doit être sélectionné pour la réalisation de B₂₂ mais n'affecte pas B₂₁.

B₂₂₁	Le plus bas besoin de composants pour désassemblage dans une unité de désassemblage
B₂₂₂	Opération de désassemblage facile pour les composants dans l'unité à désassembler

SC₂₂₁	Réduire le nombre de composants nécessaires pour le désassemblage
SC₂₂₂	Faire la conception des composants facile à désassembler

$$\begin{cases} B_{221} \\ B_{222} \end{cases} = \begin{bmatrix} X & 0 \\ X & X \end{bmatrix} \begin{cases} SC_{221} \\ SC_{222} \end{cases}$$

SC₂₂₁ doit être sélectionné pour la réalisation de B₂₂₁; SC₂₂₂ doit être sélectionné pour la réalisation de B₂₂₂ mais n'affecte pas B₂₂₁.

B₂₂₁₁	Pas de matériaux différents non nécessaires utilisés dans l'unité à désassembler
B₂₂₁₂	Le plus bas besoin de composants devant être enlevés avant les composants ou sous-assemblages visés par le désassemblage
B₂₂₁₃	Pas de composants non nécessaires séparés dans l'unité à désassembler

SC₂₂₁₁	Utiliser le même ou des matériaux compatibles avec le matériel ayant le plus d'importance dans le produit à désassembler ou dans chaque sous-assemblage; et éviter la combinaison de contamination mutuelle des différents matériaux.
SC₂₂₁₂	Moduler le produit et minimiser le nombre de composants nécessaires devant être enlevés avant les composants visés par le désassemblage dans chaque module ou sous-assemblage

SC₂₂₁₃	Éliminer les composants séparés non nécessaires pour tout le cycle de vie du produit, ou joindre avec d'autres composants nécessaires dans chaque module ou sous-assemblage
--------------------------	---

$$\begin{Bmatrix} B_{2211} \\ B_{2212} \\ B_{2213} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} SC_{2211} \\ SC_{2212} \\ SC_{2213} \end{Bmatrix}$$

SC₂₂₁₁ doit être sélectionné pour la réalisation de B₂₂₁₁; SC₂₂₁₂ doit être sélectionné pour la réalisation de B₂₂₁₂ mais n'affecte pas B₂₂₁₁; SC₂₂₁₃ doit être sélectionné pour la réalisation de B₂₂₁₃ mais n'affecte pas B₂₂₁₁ et B₂₂₁₂.

B₂₂₂₁	Enlèvement facile des fixations apparentées
B₂₂₂₂	Séparation facile des composants à désassembler
B₂₂₂₃	Manipulation facile des composants à enlever
B₂₂₂₄	Pas de repositionnement pour le désassemblage

SC₂₂₂₁	Minimiser le nombre de fixations et utiliser des méthodes de liaison facilitant le désassemblage
SC₂₂₂₂	Conception des composants à désassembler pour faciliter la séparation
SC₂₂₂₃	Conception de composants à enlever pour faciliter la manipulation
SC₂₂₂₄	Éviter le démontage des composants dans des directions opposées

$$\begin{Bmatrix} B_{2221} \\ B_{2222} \\ B_{2223} \\ B_{2224} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & X & 0 \\ 0 & 0 & 0 & X \end{bmatrix} \begin{Bmatrix} SC_{2221} \\ SC_{2222} \\ SC_{2223} \\ SC_{2224} \end{Bmatrix}$$

SC₂₂₂₁ doit être sélectionné pour la réalisation de B₂₂₂₁ mais n'affecte pas B₂₂₂₂, B₂₂₂₃, et B₂₂₂₄; SC₂₂₂₂ doit être sélectionné pour la réalisation de B₂₂₂₂ mais n'affecte pas B₂₂₂₁, B₂₂₂₃, et B₂₂₂₄; SC₂₂₂₃ doit être sélectionné pour la réalisation de B₂₂₂₃ mais n'affecte pas B₂₂₂₁, B₂₂₂₂, et B₂₂₂₄; SC₂₂₂₄ doit être sélectionné pour la réalisation de B₂₂₂₄ mais n'affecte pas B₂₂₂₁, B₂₂₂₂, et B₂₂₂₃.

B₂₂₂₁₁	Le moins possible de fixations utilisés
B₂₂₂₁₂	Accès adéquat et vision non restrictive pour le désassemblage des fixations
B₂₂₂₁₃	Enlèvement facile des fixations
R₂₂₂₁₄	Le moins possible d'outils différents ou des outils simples

SC₂₂₂₁₁	Minimiser le nombre de fixations utilisés
SC₂₂₂₁₂	Conception des composants pour assurer l'accès adéquat et une vision non restrictive pour le démontage de fixations
SC₂₂₂₁₃	Utilisation de type de fixations faciles à être enlevés
SC₂₂₂₁₄	Utilisation d'outils « innés » ou d'outils standards et le même élément de liaison pour le plus possible de composants pour minimiser le nombre de types de fixations

$$\begin{Bmatrix} B_{22211} \\ B_{22212} \\ B_{22213} \\ B_{22214} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & X & 0 \\ 0 & 0 & 0 & X \end{bmatrix} \begin{Bmatrix} SC_{22211} \\ SC_{22212} \\ SC_{22213} \\ SC_{22214} \end{Bmatrix}$$

SC₂₂₂₁₁ doit être sélectionné pour la réalisation de B₂₂₂₁₁ mais n'affecte pas B₂₂₂₁₂, B₂₂₂₁₃, et B₂₂₂₁₄; SC₂₂₂₁₂ doit être sélectionné pour la réalisation de B₂₂₂₁₂ mais n'affecte pas B₂₂₂₁₁, B₂₂₂₁₃, et B₂₂₂₁₄; SC₂₂₂₁₃ doit être sélectionné pour la réalisation de B₂₂₂₁₃ mais n'affecte pas B₂₂₂₁₁, B₂₂₂₁₂, et B₂₂₂₁₄; SC₂₂₂₁₄ doit être sélectionné pour la réalisation de B₂₂₂₁₄ mais n'affecte pas B₂₂₂₁₁, B₂₂₂₁₂, et B₂₂₂₁₃.

B₂₂₂₂₁	Identification facile des composants à désassembler
B₂₂₂₂₂	Faible force exercée pour séparer le composant à désassembler
B₂₂₂₂₃	Faible résistance de séparation des composants à désassembler

SC₂₂₂₂₁	Fournir un plan détaillé ou un marquage de couleur avec le code de matériaux pour les composants et le concepteur des composants pour assurer une vision non restrictive pour l'identification des composants à désassembler
SC₂₂₂₂₂	Conception des composants pour assurer l'accès adéquat aux composants à désassembler pour et conception des composants à désassembler pour minimiser la force exercée pour la séparation.
SC₂₂₂₂₃	Conception de composants à désassembler avec un dégagement généreux

$$\begin{Bmatrix} B_{22221} \\ B_{22222} \\ B_{22223} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} SC_{22221} \\ SC_{22222} \\ SC_{22223} \end{Bmatrix}$$

SC₂₂₂₂₁ doit être sélectionné pour la réalisation de B₂₂₂₂₁; SC₂₂₂₂₂ doit être sélectionné pour la réalisation de B₂₂₂₂₂ mais n'affecte pas B₂₂₂₂₁; SC₂₂₂₂₃ doit être sélectionné pour la réalisation de B₂₂₂₂₃ mais n'affecte pas B₂₂₂₂₁, et B₂₂₂₂₂.

B₂₂₂₃₁	Pas de contact avec des matériaux toxiques
B₂₂₂₃₂	Pas de blessure pour le désassembleur
B₂₂₂₃₃	Prise facile sur les composants

SC₂₂₂₃₁	Enfermer les matériaux toxiques dans une unité scellée
SC₂₂₂₃₂	Conception de composants qui ne sont pas coupant
SC₂₂₂₃₃	Conception de composants pas trop gros ou ayant des surfaces difficilement prenables

$$\begin{Bmatrix} B_{22231} \\ B_{22232} \\ B_{22233} \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} SC_{22231} \\ SC_{22232} \\ SC_{22233} \end{Bmatrix}$$

SC₂₂₂₃₁ doit être sélectionné pour la réalisation de B₂₂₂₃₁; SC₂₂₂₃₂ doit être sélectionné pour la réalisation de B₂₂₂₃₂ mais n'affecte pas B₂₂₂₃₁; SC₂₂₂₃₃ doit être sélectionné pour la réalisation de B₂₂₂₃₃ mais n'affecte pas B₂₂₂₃₁, et B₂₂₂₃₂.

La conception axiomatique amène directement au développement du guide de conception. Ce dernier peut être utilisé directement pour la conception pour le désassemblage par les ingénieurs. Il existe même une évaluation numérique de la conception axiomatique [4].

Tableau 10.1 : Guide de conception développé par la conception axiomatique [4]

1.	(DP ₁₁)	Produits conçus avec le plus de matériaux recyclables possibles pour remplacer les matériaux non-recyclable
2.	(DP ₁₂)	Éliminer les composants non-nécessaires pour tout le cycle de vie du produit, ou fusionner les avec d'autres composants du produit
3.	(DP ₁₃)	Éliminer le volume de matériaux non-nécessaires utilisé dans chaque composants critiques
4.	(DP ₂₁)	Utiliser des matériaux qui peuvent être réutilisés pour le plus d'applications possibles et ayant la plus grande demande avec la plus basse différence de prix entre les matériaux vierges et les recyclés
5.	(DP ₂₂₁₁)	Utiliser le même ou des matériaux compatibles avec le matériel ayant le plus d'importance dans le produit à désassembler ou dans chaque sous-assemblage; et éviter la combinaison de contamination mutuelle des différents matériaux.
6.	(DP ₂₂₁₂)	Moduler le produit et minimiser le nombre de composants nécessaires devant être enlevés avant les composants visés par le désassemblage dans chaque module ou sous-assemblage
7.	(DP ₂₂₁₃)	Éliminer les composants séparés non nécessaires pour tout le cycle de vie du produit, ou joindre avec d'autres composants nécessaires dans chaque module ou sous-assemblage
8.	(DP ₂₂₂₁₁)	Minimiser le nombre de fixations utilisés
9.	(DP ₂₂₂₁₂)	Conception des composants pour assurer l'accès adéquat en le vision non restrictive pour le démontage de fixations
10.	(DP ₂₂₂₁₃)	Utilisation de type de fixations faciles à être enlevés
11.	(DP ₂₂₂₁₄)	Utilisation d'outils « innés » ou d'outils standards et le même élément de liaison pour le plus possible de composants pour minimiser le nombre de types de fixations
12.	(DP ₂₂₂₂₁)	Fournir un plan détaillé ou un marquage de couleur avec le code de matériaux pour les composants et la concepteur des composants pour assurer une vision non restrictive pour l'identification des composants à désassembler
13.	(DP ₂₂₂₂₂)	Conception des composants pour assurer l'accès adéquat aux composants à désassembler pour et conception des composants à désassembler pour minimiser la force exercée pour la séparation.
14.	(DP ₂₂₂₂₃)	Conception de composants à désassembler avec un dégagement généreux
15.	(DP ₂₂₂₃₁)	Enfermer le toxique dans une unité scellée
16.	(DP ₂₂₂₃₂)	Conception de composants qui ne sont pas coupant
17.	(DP ₂₂₂₃₃)	Conception de composants pas trop gros ou ayant des surfaces difficilement prenables
18.	(DP ₂₂₂₄)	Éviter le désassemblage des composants dans des directions opposées

11 Conclusion

Quelque soit les applications ou les besoins face à la problématique du désassemblage, l'analyse complète du produit nous a été primordial. Dans le contexte social actuel rien ne doit être négligé dans le processus de valorisation et la manitenance des produits.

Tout d'abord, le cycle de vie des produits dans une perceptive globale et incluant les différents coûts encourus, permet de bien les connaître . C'est la première étape et la plus importante pour bien fixer nos objectifs en fonction de la vie de notre produit.

La première étape et la plus importante consiste à analyser le cycle de vie des produits dans une perspective globale, incluant l'ensemble des coûts qui peuvent intervenir tout au long de ce cycle. Ensuite, les aspects de pré-désassemblage sont analysés en commençant par la détermination des différents besoins. Suite à cela, les choix de fin de vie sont classés en différentes options (la réparation, la remise à neuf, la nouvelle fabrication, la cannibalisation, le recyclage, l'incinération, l'enfouissement, et la réutilisation directe) présentés de façon qualitative et quantitative. Connaissant le cycle de vie complet des produits et les différentes alternatives de fin de vie, l'étude porte alors sur les paramètres physiques et leur niveau de complexité (le réassemblage, la profondeur, et le cheminement). La stratégie et la méthode de résolution appropriées peuvent finalement être choisies. Pour, Plusieurs outils de collecte d'information sur la structure du produit sont présentés afin de fournir les informations requises par la méthode de résolution retenue. On retrouve notamment un outil de prise de données et la carte de pointage DEL. Pour permettre le fonctionnement des méthodes de résolution, il est nécessaire de d'adopter un mode de représentation des produits. Les deux représentations les plus utilisées dans la résolution des méthodes de désassemblage sont la matrice d'interférence et de désassemblage, ainsi que les arbres de désassemblage.

Afin de bien comprendre et de faire un choix éclairé dans le plus de cas possible, nous avons présenté et analysé un grand nombre de stratégies de désassemblage de produit. Premièrement des stratégies de base plus spécifiques ont été comparées et analysées

(planification du désassemblage avec ou sans composant défectueux, abstractions géométriques, apprentissage et raisonnement géométrique, et relation de priorité entre les composants d'assemblage). Deuxièmement les quelques méthodes de résolutions par réseaux de Petri ont été détaillées, vu leur grande utilisation dans les problèmes de désassemblage de produits, et présentées en trois parties : le réseau de Petri normal, le réseau de Petri avec relation de priorités ET/OU complexes et le réseau de Petri expert stochastique coloré et augmenté. Troisièmement, nous avons détaillé quelques stratégies considérant l'aspect environnemental et économique : l'évaluation économique des opérations de désassemblage pour le recyclage, la remise à neuf et la réutilisation, le réseau logique de désassemblage. Nous avons considéré plus particulièrement la résolution de l'aspect économique et environnemental par les algorithmes génétiques. Quatrièmement, nous avons étudié des stratégies plus spécifiques axées sur la maintenance de produit, avec des méthodes de résolution par algorithmes génétiques pour la maintenance et le désassemblage sélectif global. Pour finir, l'étude des lignes de désassemblage nous a permis d'élargir notre compréhension du désassemblage en introduisant plusieurs postes de travail et la présence d'échec des tâches.

En complément, nous avons analysé de la conception dans le but de nous aider lors du recyclage et du désassemblage. Finalement, la conception pour le désassemblage nous a donné des outils pour penser et faciliter le désassemblage des produits.

11.1 Résumé de l'analyse comparative sur les stratégies

L'analyse comparative, présentée dans les chapitres 4 à 9 avec l'étude des stratégies, peut être résumée dans les deux tableaux suivants (tableau 11.1 et 11.2). Le premier tableau (11.1) comprend les caractéristiques particulières pour chacune des stratégies.

Tableau 11.1 : Caractéristiques particulières des stratégies

Section	Stratégies	Caractéristiques particulières
5.1	Planification du désassemblage avec ou sans composant défectueux	Considération de la présence d'incertitude durant le désassemblage et possibilité de désassemblage destructif.
5.2	Abstractions géométriques	Utilisation de deux abstractions géométriques : graphe topologique d'assemblage et séries de composants limites, considération de composants avec une géométrie complexe.
5.3	Apprentissage et raisonnement géométrique	Considération du changement de direction lors du désassemblage.
5.4	Relations de priorité entre les composants d'assemblage	Algorithme simple, méthode visuelle à utiliser avec des produits simples et résolution rapide.
6.2	Réseau de Petri avec relation de priorité complexe	Évaluation des coûts de désassemblage et considération des changements d'outils.
6.3	Réseau de Petri expert stochastique coloré et augmenté	Considération de l'assemblage du produit, analyse complète de toutes les possibilités de désassemblage et d'assemblage et considération de produits plus complexes que la méthode du réseau de Petri normal.
7.1	Évaluation économique des opérations de désassemblage	Analyse de fin de vie en quatre niveaux, considération de la compatibilité des matériaux et des nœuds de disposition, utilisation des matrices de coûts.
7.2	Réseau logique de désassemblage	Méthode de résolution du chemin le plus court, résolution visuelle.
7.3	Algorithme génétique en considération de l'aspect économique et environnemental	Introduction des options de récupérations de produits dans la résolution du problème.
8.1	Algorithme génétique pour la maintenance	Recherche d'un composant spécifique du produit et utilisation dans le cadre de la maintenance et de la réparation des produits.
8.2	Désassemblage sélectif global	Application visuelle et considération de deux types de contraintes : les contraintes spatiales et celles définies par l'utilisateur.
9	Optimisation d'une ligne de désassemblage avec présence d'échec de tâche	Considération de la présence d'échec de tâches, balancement des lignes de désassemblage, et considération de la demande en composant.

Tableau 11.2 : Liens fonctionnels entre les stratégies et les différents éléments étudiés

Section	Stratégies	Besoins et types de désassemblage							Paramètres du produit								
		Optimisation des coûts	Optimisation du temps	Optimisation de la séquence	Désassemblage partiel	Désassemblage complet	Optimisation d'une ligne de désassemblage	Maintenance du produit	Optimisation assemblage et désassemblage	Aspect environnemental	Réparation	Remise à neuf	Nouvelle fabrication	Cannibalisation	Recyclage	Beaucoup de composants	Relations de priorité complexes
5.1	Planification du désassemblage avec ou sans composant défectueux																
5.2	Abstractions géométriques																
5.3	Apprentissage et raisonnement géométrique																
5.4	Relations de priorité entre les composants d'assemblage																
6.2	Réseau de Petri avec relation de priorité complexe																
6.3	Réseau de Petri expert stochastique coloré et augmenté																
7.1	Évaluation économique des opérations de désassemblage																
7.2	Réseau logique de désassemblage																
7.3	Algorithme génétique en considération de l'aspect économique et environnemental																
8.1	Algorithme génétique pour la maintenance																
8.2	Désassemblage sélectif global																
9	Optimisation d'une ligne de désassemblage avec présence d'échec de tâche																

Le deuxième tableau (11.2) fait la comparaison entre les stratégies au niveau des besoins du désassembleur (optimisation des coûts, optimisation du temps, optimisation de la séquence, optimisation d'une ligne, désassemblage partiel ou complet, considération de l'aspect environnemental et économique, et maintenance du produit), des différents types de désassemblage (réparation, remise à neuf, cannibalisation, nouvelle fabrication, et recyclage) et des paramètres particuliers du produit (beaucoup de composants et relations de priorité complexes). Suite à cette analyse, il est possible de trouver la meilleure stratégie pour notre application.

11.2 Perspectives

À court terme, les outils développés vont permettre une amélioration du temps de désassemblage ainsi que des techniques à utiliser, une meilleure gestion des pièces récupérées et une diminution des coûts de désassemblage. À moyen et long terme, ils vont rendre la conception plus adaptée au développement durable. Ces résultats vont être réalisés par l'entremise de stratégies et de méthodes d'aide au désassemblage et à la récupération.

Les méthodes de conception permettent d'ouvrir la voie à une gestion environnementale dans des domaines dont la production se fait à grand volume et où les méthodes traditionnelles de conception sont actuellement bien encrées (ex. : l'automobile). Le présent projet aura comme principales retombées les deux points suivants : 1) Améliorer les pratiques actuelles des entreprises, de manière à ce qu'elles soient plus respectueuses de l'environnement, en proposant à ces entreprises des outils adaptés aux exigences du développement durable. 2) Par une diffusion des connaissances développées, il sera possible de démontrer à l'industrie manufacturière en générale qu'il est possible de concevoir des produits dans une perspective de conception environnementale et de développement durable.

Si les stratégies présentées ne semblent pas être en mesure de travailler avec des produits très complexes (avec un très grand nombre de composants), il est possible de travailler avec des sous-assemblage de produit. Des sous-ensembles peuvent être créés pour leur désassemblage et ensuite nous travaillons avec les sous-ensembles séparément. Ce document peut être utilisé par des entreprises désirant faire le désassemblage de leurs produits dans un objectif de développement durable et pouvant s'appliquer pour une grande variété de types et de structures de produits.

Comme travaux futurs, en complément de cette analyse, il serait important d'appliquer les stratégies pour les produits dans le but de voir la réelle efficacité des différentes stratégies. La mise à jour des stratégies en intégrant de nouvelles méthodes et algorithmes de désassemblage, serait aussi essentiel pour rester branché sur ce domaine toujours en mouvement.

Le développement durable et en particulier le problème de désassemblage de produit prennent de plus en plus d'importance dans les entreprises et dans le domaine politique. Beaucoup d'argent, mais surtout l'environnement, en jeu dans tout cela, alors rien ne peut détourner notre regard sur cette analyse des stratégies de désassemblage. Peut-être qu'un jour tous les produits seront biodégradables et que les méthodes pour faire le désassemblage seront tout exactement les mêmes que celles pour réaliser l'assemblage!

12 Références

- [1] ALTING, L., "Life-Cycle Design of Products: A New Opportunity for Manufacturing Enterprises", *Institute of Manufacturing Engineering*, Auerbach Publishers, (1991).
- [2] BAUMGARTEN, H., R. KLINKNER et T. SOMMER-DITTRICH, "Reconfigurable logistics systems in production and disassembly networks", *International Journal of production Research*, vol. 42, no. 17, (2004), p. 3647-3655.
- [3] BOVEA, M. D., et B. WANG. "Identifying Environmental Options by Combining Life Cycle Assessment and Fuzzy set Theory", *International Journal of Production Research*, vol. 41, no. 3, (2003), p. 593-609.
- [4] CHEN C., "Design for the environment: A Quality-Based Model for Green Product Development", *Management Science*, vol. 47, no. 2, (2001), p. 250-263.
- [5] CHEN, Ke-Zhang, "Development of integrated design for disassembly and recycling in concurrent engineering", *Integrated Manufacturing Systems*, vol. 12, no. 1, (2001), p. 67-79.
- [6] DAS, Sanchoy K., Pradeep YEDLARAJIAH, et Raj NARENDRA. "An approach for estimating the end-of-life product disassembly effort and cost". *International Journal of Production Research*, vol. 38, no. 3, (2000), p. 657-673.
- [7] ERTAS A., J. C JONES. "The Engineering Design Process". Second Edition, John Wiley & Sons Inc, (1996), p.195-198.
- [8] GAO, Meimei, MengChu ZHOU, et Reggie J. CAUDILL, "Integration of Disassembly Leveling and Bin Assignment for Demanufacturing Automation". *IEEE Transactions on Robotics and Automation*, vol. 18, No. 6, (2002), p. 867-873.
- [9] GARCIA, Miguel Angel, Albert LARRÉ, Beatriz LOPEZ, et Albert OLLER. "Reducing the Complexity of Geometric Selective Disassembly". *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Takamatsu, Japan, vol.2, (november 2000), p. 1474-1479.
- [10] GUIDE, V. D. R., R. SRIVASTAVA, et M. E. KRAUS. "Product Structure Complexity and Scheduling of Operations in Recoverable Manufacturing". *International Journal of Production Research*, vol. 35, no. 11, (1997), p. 3179-3199.

- [11] GUNGOR, Askiner, et Surendra M. GUPTA. "An Evaluation Methodology For Disassembly Processes". *Computers & Industrial Engineering*, vol. 33, nos. 1-2, (1997), p. 329-332.
- [12] GUNGOR, Askiner, et Surendra M. GUPTA. "Disassembly Sequence Planning for Products with Defective Parts in Products Recovery". *Computers & Industrial Engineering*, vol. 35, nos. 1-2, (1998), p.161-164.
- [13] GUNGOR, Askiner, et Surendra M. GUPTA. "Disassembly sequence plan generation using a branch-and-bound algorithm". *International Journal of Production Research*, vol. 39, no. 3, (2001), p. 481-509.
- [14] GUNGOR, Askiner, et Surendra M. GUPTA. "A solution approach to the disassembly line balancing problem in the presence of task failures". *International Journal of Production Research*, vol. 39, no. 7, (2001), p. 1427-1467.
- [15] GUNGOR Askiner, et Surendra M. GUPTA. "Disassembly line in product recovery". *International Journal of Production Research*, vol. 40, no. 11, (2002), 2569-2589.
- [16] GUPTA, Surendra M., et Charles R. MCLEAN. "Disassembly of Products". *Computers & Industrial Engineering*, vol. 31, nos. 1-2, (1996), p. 225-228.
- [17] GUPTA, S. M., et K. N. TALEB. "Scheduling disassembly". *International Journal of Production Research*, vol. 32, no. 8, (1994), p. 1857-1866.
- [18] HUANG, Yuan Mao, et Chun-Ting HUANG. "Disassembly matrix for disassembly processes of products". *International Journal of Production Research*, vol. 40, no. 2, (2002), p. 255-273.
- [19] HU, D., LI C., et HU Y. "Mechanical Product Disassembly Sequence and Path Planning Based on Knowledge and Geometric Reasoning". *International Journal Advanced Manufacturing Technology*, vol. 22, (2002), p. 688-696.
- [20] JOHNSON, M. R., et M. H. WANG. "Planning product disassembly for material recovery opportunities". *International Journal of Production Research*, vol. 33, no. 11, (1995), p. 3119-3142.
- [21] JOHNSON, M. R., et M. H. WANG. "Economical evaluation of disassembly operations for recycling, remanufacturing and reuse". *International Journal of Production Research*, vol. 36, no. 12, (1998), p. 3227-3252.
- [22] KAZMIERCZAK, Karolina, Jorden WINKEL, et Rolf H. WESTGOARD. "Car disassembly and ergonomics in Sweden : current situation and future perspectives in light if new environmental legielation". *International Journal of Production Research*, vol. 42, no. 7, (2004), p. 1305-1324.

- [23] KROLL, E., et B. S. CARVER. "Disassembly analysis through time estimation and other metrics", *Robotics and Computer Integrated Manufacturing*, no.15, (1999), p. 191-200.
- [24] KUMAR, S., R. KUMAR, R. SHANKAR, et M. K. TIWARI. "Expert Enhanced Coloured Stochastic Petri Net and its Application in Assembly/Disassembly", *International Journal of Production Research*, vol. 41, no. 12, (2003), p. 2727-2762.
- [25] KUMAR, S. A., M. K. TIWARI, et S. K. MUKHOPADHYAY. "Modelling and Planning of the Disassembly Processes Using an Enhanced Expert Petri Net". *International Journal of Production Research*, vol. 41, no. 16, (2003), p. 3761-3792.
- [26] KUO, Tsai C. "Disassembly sequence cost analysis for electromechanical products", *Robotics and Integrated Manufacturing*, vol. 16, (2000), p. 43-54.
- [27] KUO, Tsai C., Hong C. ZHANG, et Samuel H. HUANG. "Disassembly analysis for electromechanical products: a graph-based heuristic approach". *International Journal of Production Research*, vol. 38, no. 5, (2000), p. 993-1007.
- [28] LAMBERT, Alfred J. D. "Optimal disassembly of complex products". *International Journal of Production Research*, vol. 35, no. 9, (1997), p. 2509-2523.
- [29] LAMBERT, Alfred J. D. "Linear Programming in Disassembly/Clustering Sequence Generation". *Computers & Industrial Engineering*, vol. 36, (1999), p. 723-738.
- [30] LAMBERT Alfred J. D. "Determining optimum disassembly sequences in electronic equipment". *Computers & Industrial Engineering*, no.43, (2002), p. 553-575.
- [31] LAMBERT Alfred J. D. "Disassembly sequencing: a survey". *International Journal of Production Research*, no.41, (2003), p. 553-575.
- [32] LI, J. R., L. P. KHOO, et S. B. TOR. "A Novel Representation Scheme for Disassembly Sequence Planning". *International Journal Advanced Manufacturing Technology*, vol. 20, (2001), p. 621-630.
- [33] MATINEZ, Michel, Viet-Hung PHAM, et Joël FAVREL, "Dynamic Generation of Disassembly Sequences", *IEEE 6th International conference on Emerging Technologies and Factory Automation*, 9-12 sept., (1997), p. 177-182.

- [34] MOORE, K. E., A. GUNGOR, et S. M. GUPTA, "Disassembly Petri Net Generation in the presence of XOR precedence relationships", *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 1, 11-14 oct., (1998), p. 289-296.
- [35] MOORE, Kendra E., Askiner GUNGOR, et Surendra M. GUPTA. "A Petri Net Approach to Disassembly Process Planning". *Computers & Industrial Engineering*, vol. 35, nos. 1-2, (1998), p. 165-168.
- [36] MOORE, Kendra E., Askiner GUNGOR, et Surendra M. GUPTA. "Petri Net Approach to Disassembly Process Planning for Products with Complex AND/OR Precedence Relationships". *European Journal of Operational Research*, vol. 135, (2001), p. 428-449.
- [37] PENEV, K. D., et A. J. de RON. "Determination of a disassembly strategy". *International Journal of Production Research*, vol. 34, no. 2, (1996), p. 495-506.
- [38] PNUELI, Y., et E. ZUSSMAN. "Evaluating the end-of-life value of a product and improving it by redesign". *International Journal of Production Research*, vol. 35, no. 4, (1997), p. 921-942.
- [39] RAI R., V. RAI, M. K. TIWARI, et V. ALLADA. "Disassembly sequence generation : a Petri net based heuristic approach". *International Journal of Production Research*, vol. 40, no. 13, (2002), 3183-3198.
- [40] SEO, K. K., J. H. PARK, et D. S. JANG. "Optimal Disassembly Sequence Using Genetic Algorithms Considering Economic and Environmental Aspects". *International Journal Advanced Manufacturing Technology*, vol. 18, 2001, p. 371-380.
- [41] SHYAMSUNDAR, N., et Rajit GADH. "Geometric Abstractions to Support Disassembly Analysis". *IIE Transactions*, vol. 31, (1999), p. 935-946.
- [42] SKAF A., S. GERNER, Z. BINDER, B. DESCOTES-GENON, et B. DAVID. "Disassembly Process Planning", *Proceeding of the IAR Annual Meeting, Strasbourg (France)*, 1999.
- [43] SRINIVASAN, Hari, et Rajit GADH, "Complexity Reduction in Geometric Selective Disassembly Using the Wave Propagation Abstraction", *IEEE International Conference on Robotics and Automation*, vol. 2, no. 1, (1998), p. 1478-1483.
- [44] SRINIVASAN, Hari, et Rajit GADH. "A Non-Interfering Selective Disassembly Sequence for Components with Geometric Constraints". *IIE Transactions*, vol. 34, (2002), p. 349-361.

- [45] SUGA, T., K. SANESHIGE, et J. FUJIMOTO, "Quantitative Disassembly Evaluation", *IEEE International symposium on Electronics and the environment*, 6-8 May, (1996), p. 19-24.
- [46] TALEB, Karim N., et Surendra M. GUPTA. "Disassembly of Multiple Product Structures". *Computers & Industrial Engineering*, vol. 32, no. 4, (1997), p. 949-961.
- [47] TANG, Ying, MengChu ZHOU, et Reggie J. CAUDILL, "An Integrated Approach to Disassembly Planning and Demanufacturing Operation", *IEEE Transactions on Robotics and Automation*, vol. 17, No. 6, (2001), p. 773-784.
- [48] TANG, O., R. W. GRUBBSTROM et S. ZANONI, "Economic evaluation of disassembly processes in remanufacturing systems", *International Journal of Production Research*, vol. 42, no. 7, (2004), p. 3603-3617.
- [49] THIERRY, Martin, Marc SALOMON, Jo Van NUNEN, et Luk Van WASSENHOVE. "Strategic Issues in Product Recovery Management". *California management review*, vol. 37, no. 2, (1995), p. 114-136.
- [50] TIWARI, M. K., Niraj SINHA, Shailendra HUMAR, Rahul RAI, et S. K. MUKHOPADHYAY. "A Petri Net Based Approach to Determine the Disassembly Strategy of a Product". *International Journal of Production Research*, vol. 40, no. 5, (2001), p. 1113-1129.
- [51] TORRES, F., S. T. PUENTE, et R. ARACIL. "Disassembly Planning Based on Precedence Relations Among Assemblies". *International Journal Advanced Manufacturing Technology*, vol. 21, (2003), p. 301-327.
- [52] VISWANATHAN, Shivakumar, et Venkat ALLADA. "Configuration analysis to support product redesign for end-of-life disassembly". *International Journal of Production Research*, vol. 39, no. 8, (2001), p. 1733-1753.
- [53] VUJOSEVIC, R., R. RASKAR, N. V. YETUKURI, M. C. JOTHISHANKAR, et S.-H. JUANG. "Simulation, animation, and analysis of design disassembly for maintainability analysis". *International Journal of Production Research*, vol. 33, no. 11, (1995), p. 2999-3022.
- [54] XU, Xiaoming, Song HUANG, Ling LING, Yujin HU, Chenggang LI, et Junjun HU. "Modelling and solving of product disassembly activity logic network". *International Journal Advanced Manufacturing Technology*, vol. 23, (2004), p. 220-226.
- [55] ZHANG, Hong C., et T. C. KUO, "A graph-based disassembly sequence planning for EOL product recycling", *IEEE international Electronics Manufacturing Technology Symposium*, 13-15 oct., (1997), p. 140-151.

- [56] ZUSSMAN, A., et MengChu ZHOU, "A methodology for modeling and adaptive planning of disassembly process", *IEEE Transactions on Robotics and Automation*, vol. 15, No. 1, (1999), p. 190-194.
- [57] ZUSSMAN, Eyal. "Planning of disassembly systems", *Assembly Automation*, vol. 15, No. 4, (1995), p. 20-23.

Annexe A : Algorithme de Dijkstra

Algorithme de Dijkstra :

A chaque lien (i,j) (arc) du graphe, on associe une longueur $l(i,j)$; $l(i,j)$ peut s'exprimer en unité de longueur, de temps, d'argent ou autres.

La longueur d'un chemin S entre une paire de sommets du graphe est donnée par :

$$L(S) = \sum_{(i,j) \in S} l(i,j)$$

L'algorithme consiste, partant d'une source s, à chercher progressivement le premier, le second, ..., sommets les plus proches jusqu'à ce que tous les sommets soient atteints.

Pour chaque sommet j, on utilisera des paramètres $d(j)$ et $p(j)$ où :

$D(j)$: longueur du plus court chemin de s à j

$P(j)$: le nœud précédent immédiatement j sur le plus court chemin de s à j

Chaque nœud peut être soit « ouvert » ou « fermé ».

Soit k l'indice indiquant le dernier nœud fermé (dernier nœud \in au plus court chemin).

Étape 1 : Pour initialiser le processus, on pose :

$$d(s) = 0 \quad \text{et} \quad p(s) = *$$

$$d(j) = \infty \quad \text{et} \quad p(j) = - \quad \text{pour tout } j \neq s$$

On considère que s fermé ($k \leftarrow s$) et que tous les autres sont ouverts.

Étape 2 : Pour actualiser les indices de chaque sommet, on examine tous les arcs (k,j). Si le nœud j est fermé, aller au prochain nœud i ; si le nœud j est ouvert, remplacer $d(j)$ par :

$$d(j) = \text{Min}\{d(j), d(k) + l(k, j)\}$$

Étape 3 : Pour choisir le prochain nœud à fermer, comparer le $d(j)$ de tous les nœuds ouverts et choisir le nœud ayant le plus petit $d(j)$. Soit i ce nœud, c'est-à-dire :

$$d(j) = \text{Min}\{d(j)\} \quad \forall j \text{ ouvert}$$

Étape 4 : Pour trouver le nœud précédent immédiatement i , considérer un à la fois, les arcs (j,i) partant du nœud fermé j au nœud i jusqu'à ce qu'on trouve celui qui vérifie la relation :

$$d(i) - l(j,i) = d(j)$$

Étape 5 : Considérons maintenant i comme un nœud fermé. Si tous les nœuds du graphe sont fermés, on arrête (fin de la procédure). Sinon poser $k = i$ et aller à l'étape 2. Lorsque tous les nœuds sont fermés, alors la quantité $d(j)$ indique la longueur du plus court chemin s à j et $p(j)$ le précédent immédiat de j sur ce plus court chemin.

Source : Ait-Kadi, Daoud, « Optimisation des systèmes », département de génie mécanique, Université Laval, (automne 2001)

Annexe B : Document de collecte de données

Document de collecte de données

Produit :

Marque : _____

Age : _____

Type de fin de vie pour lequel il est destiné : _____

Coût de l'intervenant : _____ \$/hr

Regroupement de pièces :

Nombre de pièces à désassembler : _____

Annexe C : Nomenclature Réseau de Pétri

Nomenclature pour le réseau de Pétri avec relations de priorité ET/OU complexes
(Moore, Gungor, Gupta [34, 35,36])

- 0 : vecteur zéro (ligne ou colonne) ou matrice zéro
- α : pénalité de changement d'outil
- β : pénalité de changement de direction
- δ : pénalité pour le délais d'enlèvement de composant
- Δt_j : temps de changement de t_j , $t_j \in E[M_n]$
- a : fonction binaire pour calculer la pénalité de changement d'outil
- A : matrice d'arc d'incidence pour PN; représentation compacte de F et W
- a_{ij} : un élément dans A; également $a(p_i, t_j)$
- $a(p_i, t_j)$: un élément dans A; également a_{ij}
- AG_j : série de priorité ET
- b : fonction binaire pour calculer la pénalité de changement de direction
- B: matrice de priorité de désassemblage (DPM) $B = [b_{ij}], i, j = 1, \dots, k$, où
- $$b_{ij} = \begin{cases} 1 & \text{composant } i \text{ précède en ET composant } j, \\ d & \text{composant } i \text{ précède en OU composant } j, \\ 0 & \text{autrement.} \end{cases}$$
- B_{i-} : ligne i dans B
- B_{-j} : colonne j dans B
- c_i : composant i dans un produit à désassembler
- C_i : coût associé avec le i^e marquage de DPN
- D : série de désassemblage de mouvement de directions rectilinéaires possibles
- d : direction de désassemblage
- E : matrice d'éléments d'obstruction, $k \times |D|$, $E = [e_{id}]$, où e_{id} est la série d'éléments qui obstrue le composant i dans la direction de $d(i, \dots, k)$

$$\hat{e}_{id} = \begin{cases} e_{id} & |e_{id}| = \min\{|e_{id}| \mid \forall d \in D\}, \\ \infty & \text{autrement.} \end{cases}$$

- e_{id} : série d'éléments d'obstruction de i dans la direction d ($i = 1-k$)
 \hat{E} : E minimisé contenant les séries minimales de e_{id} (sans conflits)
 $E[M_i]$: série de transitions disponibles à M_i
 ET_q : transitions disponibles avec la profondeur q
 F : la série d'arcs dans PN
 g_i : cardinalité minimum d'une série de lignes i de E min $\{e_{id}\}$ pour la ligne i de E , $d \in D$
 G_i : série de priorités minimales dans un ligne i de E
 G_i^A : série de ET avec G_i
 G_c : série de ET pour une paire d'éléments dans G_i
 g_c : un élément de conflit dans G_i
 H : série de composants contenant les matériaux hasardeux
 H_r : série de composants hasardeux qui ont été enlevé
 IM_q : série de marquage intermédiaire générée de n'importe quel M' ou IM_{q-1}
 \overline{IM} : $\{IM_q\}$ dans d'ordre ascendant de coût
 k : nombre de composants dans le produit
 L_c : niveaux de recherche calculés accordés à $|T|$ et w , $e = 1$ à z ; (ex : $L = \{0,3,6,9\}$ quand $|T| = 9$ et $w = 3$);
 M_0 : marquage initial de PN
 M_i : le i^e marquage de PN
 M' : série actuelle de meilleur marquage v avec la profondeur, q
 nz_i : 0 si B_i contient des entrées non-zéro ; 1 si $B_i = 0$
 O_j : directions où il existe un groupe de priorités OU à j ; $O_j \subseteq D$
 $OG_{j,d}$: Groupe de priorité OU à j dans la direction d
 P : série de places dans PN
 p_i : place dans P
 q : profondeur actuelle de l'arbre
 $R(M_0)$: série accessible de marquage dans PN
 Σ : $\{\sigma_i\}$, $i = 1$ à v , où σ_i est un séquence de transitions éliminées associées avec le i^e élément dans M'

- \bar{S} : série de sous-réseaux de désassemblage associés avec un DPM, B
- S_A : sous-réseaux de désassemblage pour un composant de priorité ET
- S_{AO} : sous-réseaux de désassemblage pour un composant de priorité ET/OU
- S_M : sous-réseaux de désassemblage émergeant d'autres sous-réseaux de désassemblage
- S_{NA} : sous-réseaux de désassemblage pour un composant sans antécédent
- S_{NP} : sous-réseaux de désassemblage pour un composant sans priorité
- SA : série contenant les composants ayant une relation de priorité ET avec le composant i
- T : série de transitions dans PN
- t_j : une transition dans T
- w : profondeur de recherche
- W : série de poids d'arc pour PN
- $w(a,b)$: poids de l'arc de a à b dans un PN
- z : nombre de niveaux de recherche dans l'arbre de liens, où
- $$z = \begin{cases} \lceil |T|/w + 1, & (|T| \text{ Mod } w) = 0, \\ \lfloor |T|/w + 2, & \text{autrement.} \end{cases}$$
- zc : compteur de niveau de recherche

Annexe D : Nomenclature Ligne de désassemblage

Nomenclature pour l'approche d'optimisation avec présence d'échec de tâches (Gungor, Gupta [14])

$\lceil x \rceil$	plus petit entier $\geq x$.
$ X $	cardinalité de série x .
A_k	série de tâches assignées au poste de travail k .
AP_k	série de tâches que ont été assigné aux postes de travail précédant le poste de travail k .
AT	série de toutes les tâches de désassemblage ; $ AT = N$.
c	temps de cycle.
cl	niveau actuel.
C_{EP}	coût encouru à chaque fois qu'un départ prématuré de pièces de travail est observé.
C_{SP}	coût encouru à chaque fois qu'un évitement de pièce de travail est observé.
C_{SSP}	coût encouru à chaque fois qu'un auto-évitement de pièces de travail est observé.
CA_k	série de tâches candidates qui peuvent être assignées au poste de travail k .
CT_k	série de tâches complètes qui incluent les tâches complétées au poste de travail précédant le poste k et les tâches complétées au poste k , ($CT_k = AP_k + A_k$).
$CCF(e_{ij})$	coût de complication de l'arc e_{ij} .
$CCF(SDP)$	coût de complication du chemin direct le plus court (SDP).
$CSDP$	SDP actuel.
dm_i	demande pour le composant i .
dm_{max}	demande maximum.
ev_i	évènement i de $EV(e_{ij})$.
ev_i^D	série de tâches défectueuses dans ev_i .
ev_i^{ND}	série de tâches non-défectueuses dans ev_i .
E_{CSDP}	série d'arcs du SDP actuel.

E_{SDP}	série d'arcs du SDP.
E_{SN}	série d'arcs de SN.
$E(e_{ij})$	série d'éléments représentés par l'arc entre les états s_i et s_j .
$E(s_i)$	série d'éléments représentés par l'état s_i .
ET_{s_i}	tâches disponibles de l'état s_i .
$EV(e_{ij})$	espace d'événement s'accordant aux éléments de l'arc e_{ij} . EV représente toutes les occurrences possibles.
$fE(e_{ij})$	premier élément dans $E(e_{ij})$.
$FT(ev_i)$	série de tâches éliminées dues aux tâches échouées contenues dans ev_i .
$FT(f)$	série de tâches éliminées dues à l'échec de la tâche f . Ces tâches sont les tâches qui sont directement ou indirectement précédées par la tâche f .
$I(e_{ij})$	temps mort du poste de travail représenté par l'arc e_{ij} .
L	durée de la période de planification.
$LFGN$	liste de nœuds dans ISN satisfaisant le critère (a).
LGN	liste de tous les nœuds dans ISN.
LN	liste de nœuds.
$LSDP$	liste de tous SDP dans WSN.
M	nombre de postes de travail.
N	nombre de composants du produit, étant égal au nombre de tâches
NLN	liste de nouveaux nœuds.
$OG_{i,d}$	groupe OU, le série de tâches qui inclues des priorités OU dans la direction d.
$p(i)$	probabilité d'échec de la tâche i.
$P_{arrival}(e_{ij})$	probabilité que la pièce de travail arrive au poste représenté par l'arc e_{ij} .
$P_{DWP}(e_{ij})$	probabilité d'élimination de la pièce de travail.
$P_{DWP}(ev_i)$	probabilité d'élimination de la pièce de travail ayant comme résultat l'événement ev_i .
$P_{EP}(e_{ij})$	probabilité de départ prématuré de la pièce de travail.
$P_{EP}(ev_i)$	probabilité de départ prématuré de la pièce de travail ayant comme résultat l'événement ev_i .

$P_S(e_{ij})$	probabilité de la pièce de travail de quitter le poste de travail avec succès (S) comme planifié.
$P_S(ev_i)$	probabilité de la pièce de travail de quitter le poste de travail avec succès (S) travail ayant comme résultat l'événement ev_i .
$P_{SP}(e_{ij})$	probabilité de la pièce de travail d'éviter le prochain poste de travail ; probabilité d'évitement.
$P_{SP}(ev_i)$	probabilité de la pièce de travail d'éviter le prochain poste de travail ayant comme résultat l'événement ev_i .
$P_{SSP}(e_{ij})$	probabilité de la pièce de travail d'auto évitement du poste de travail ; probabilité d'auto évitement.
$P_{SSP}(ev_i)$	probabilité de la pièce de travail d'auto évitement du poste de travail ayant comme résultat l'événement ev_i .
q_i	nombre de composant semblable dans le produit.
r_{ij}	élément dans la i° ligne et la j° colonne de la matrice.
R	matrice de priorité de désassemblage (DPM) ; $r = \{r_{ij} ; i, j = 1, \dots, N\}$.
$R_{i.}$	ligne i de R .
$R_{.j}$	colonne j de R .
$ST(s_i)$	temps de l'état s_i .
T	durée cumulative de toutes les tâches de désassemblage.
TSE	éléments d'état temporaire.
t_i	temps nécessaire pour effectuer la tâche i .
t_{max}	temps nécessaire pour compléter la plus longue tâche.