



HAL
open science

Méthodes exactes et heuristiques pour l'optimisation de l'agencement d'un logement : application aux situations de handicap

Yahya Bouzoubaa

► **To cite this version:**

Yahya Bouzoubaa. Méthodes exactes et heuristiques pour l'optimisation de l'agencement d'un logement : application aux situations de handicap. Modélisation et simulation. Université de Lorraine, 2017. Français. NNT : 2017LORR0369 . tel-01810764

HAL Id: tel-01810764

<https://theses.hal.science/tel-01810764>

Submitted on 8 Jun 2018

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

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



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>



UNIVERSITÉ
DE LORRAINE



THÈSE

Présentée par

Yahya BOUZOUBAA

Pour l'obtention du grade de
Docteur de l'Université de Lorraine
Spécialité : Informatique

Intitulée

**Méthodes exactes et heuristiques pour l'optimisation de
l'agencement d'un logement – application aux situations de
handicap –**

CHU Feng	Rapporteur - Professeur, Université d'Evry
HIFI Mhand	Rapporteur - Professeur, Université de Picardie Jules Vernes (Amiens)
ELLOUMI Sourour	Examineur - Professeur, ENSTA ParisTech
SAUER Nathalie	Examineur - Professeur, Université de Lorraine
SLIMANE Mohamed	Président du jury - Professeur, Université de Tours
NAGIH Anass	Directeur de thèse - Professeur, Université de Lorraine
PRUSKI Alain	Co-directeur de thèse - Professeur, Université de Lorraine
MINICH Christian	Co-encadrant – Maître de conférences, Université de Lorraine

Remerciements

Je tiens à exprimer à travers ces mots ma profonde gratitude à mes encadrants de thèse Anass NAGIH, Professeur à l'université de Lorraine, Alain PRUSKI, Professeur à l'université de Lorraine, et Christian MINICH, Maître de conférences à l'université de Lorraine. Cette thèse aurait eu un autre goût sans leur confiance et leur implication tout au long de ces années de doctorat.

Mes remerciements vont également aux membres de mon jury de thèse : Feng CHU, Professeur à l'Université d'Evry, et Mhand HIFI, Professeur à l'Université de Picardie Jules Vernes, pour avoir accepté d'examiner ce travail en qualité de rapporteurs, Sourour ELLOUMI, Professeur à ENSTA ParisTech, et Nathalie SAUER, Professeur à l'Université de Lorraine, pour avoir accepté de faire partie du jury en qualité d'examineurs, et Mohammed SLIMANE, professeur à l'Université de Tours, pour avoir accepté de présider ce jury de thèse.

Un grand merci à mes collègues que j'ai pu côtoyer pendant ce doctorat, que ce soit dans mes activités de recherche au sein du laboratoire LCOMS ou dans mes activités d'enseignement à l'UFR MIM et à l'UFR SciFa.

Je remercie chaleureusement mes parents Majda et Jaouad, ma sœur Soukayna et mon frère Mohamed de m'avoir soutenu durant ces longues années d'études, le chemin était long mais il valait la peine. Merci également à mes amis qui étaient là jusqu'au dernier souffle.

« Le vrai pouvoir, c'est la connaissance »

**Francis Bacon (1561 - 1626).
Scientifique et philosophe anglais**

Table des matières

Remerciements	C
Introduction	1
Chapitre I --- Etat de l'art	3
1. Introduction	3
2. Handicap	3
2.1. Etymologie du mot handicap	3
2.2. Définition du handicap	4
2.3. Vers une classification internationale du handicap	4
2.3.1. Classification Internationale des Handicaps (CIH/CIM)	4
2.3.2. Classification Internationale du Fonctionnement, du Handicap et de la Santé (CIF/CIM-10) ..	5
2.4. Vers un processus d'automatisation en France	6
2.4.1. Loi du 30 juin 1975 d'orientation en faveur des personnes handicapées	6
2.4.2. Le droit à la compensation : la loi du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées	6
2.5. Les différents types de handicap	7
3. Aménagement spatial	8
3.1. Aménagement du territoire durable	9
3.1.1. Aménagement du territoire pour l'accessibilité aux personnes handicapées	10
3.1.2. Quelques initiatives pour un territoire accessible aux personnes handicapées	12
3.1.2.1. Grenoble : Projet INNOVACCES	12
3.1.2.2. Nantes : un plan d'actions handicap 2014-2020 pour répondre aux enjeux d'accessibilité	13
3.2. Aménagement d'un espace intérieur	13
3.2.1. ERP : Etablissement recevant du public	13
3.2.2. Habitat	14
4. Diverses approches de l'agencement	17
4.1. Introduction	17
4.2. Typologies et caractéristiques des problèmes de placements	17
4.3. Définition	19
4.4. Formulation du problème et nature des contraintes à respecter	20
5. Travaux existants	22
5.1. Problème de conception de circuits VLSI	23
5.2. Problème d'agencement d'un environnement intérieur	23
5.2.1. Travaux de Schwarz et al	23
5.2.2. Travaux de Michalek et al.	25
5.2.2.1. Module d'optimisation géométrique	26
5.2.2.1.1. Les contraintes supportées	26
5.2.2.1.1. Les fonctions objectifs utilisées	26
5.2.2.1.2. Les méthodes de résolutions	27
5.2.2.2. Module d'optimisation topologique	27
5.2.2.2.1. Les contraintes et les fonctions objectif supportées	29
5.2.2.2.2. Les méthodes de résolutions	29
5.2.2.3. Exemple	29
5.2.3. Projet HM2PH (Habitat Modulaire et Mobile pour Personnes Handicapées)	30
5.2.4. Les travaux de Leloup [50]	30
5.2.4.1. Les bases de données pour une analyse des déficiences et capacités de l'habitant	30
5.2.4.2. Moteur de génération Automatique de plans (MoGAP)	31
a. Méthode basée sur l'algorithme génétique	32
b. Méthode basée sur la procédure de séparation et d'évaluation	33
5.2.4.3. Validation du plan final	33
5.2.5. Quelques remarques	33

5.2.6.	Les travaux de Puret [68].....	33
5.2.6.1.	Modélisation des besoins et souhaits de l'habitant	34
5.2.7.	Module de génération de plans.....	35
5.2.7.1.	Recherche topologique	36
5.2.7.2.	Optimisation géométrique	36
5.2.8.	Quelques remarques	37
6.	Conclusion	37
Chapitre II --- Modélisation et Algorithmes géométriques		39
1	Introduction	39
2	Produit scalaire et produit vectoriel	40
3	Calcul du sens de rotation d'un polygone / La normale orientée à un polygone.....	41
4	Un sommet d'un polygone est-il convexe ou concave ?	42
5	Normale extérieure / intérieure à un segment d'un polygone	42
6	Calcul du pourtour d'une pièce.....	43
7	Une pièce est-elle orientée vers une direction donnée ?	46
8	Une pièce ne donne pas vers une direction donnée ?	48
9	Calculer un point à l'intérieur d'un polygone.....	48
10	Vérifier si un point est dans la région du plan entourée par un polygone.....	49
11	Aire d'un polygone	52
12	Angle d'un sommet d'un polygone	52
13	Calcul de l'offset à un polygone	53
14	Distance d'un point à un segment.....	55
15	Inclusion d'un polygone dans un autre	55
16	Forme raisonnable	58
17	Détection des couloirs.....	60
18	Distance d'un segment à un obstacle.....	63
Chapitre III --- Synthèse d'un aménagement.....		67
1.	Introduction.....	67
2.	Placement et dimensionnement des pièces.....	69
2.1.	Première approche de modélisation mathématique (modèle 1).....	69
2.1.1.	Les entrées.....	69
2.1.2.	Les variables.....	70
2.1.3.	Le modèle mathématique.....	71
2.1.4.	Explication de la fonction objectif	72
2.1.5.	Justification des contraintes	73
2.2.	Deuxième approche de modélisation mathématique (modèle 2)	76
2.2.1.	Les entrées.....	76
2.2.2.	Les variables.....	76
2.2.3.	Le modèle mathématique.....	77
2.2.4.	Explication des contraintes.....	78
2.2.5.	Explication de la fonction objectif	78
3.	Placement des portes.....	78
3.1.	Les entrées	80
3.2.	Les variables	81
3.3.	Le modèle mathématique	81
3.4.	Explications des contraintes	81
3.5.	Explications de la fonction objectif.....	82
4.	Suppression des couloirs inutiles.....	82
4.1.	Sélection des murs autorisés à se déplacer.....	83
4.2.	Tri des murs à déplacer par priorité décroissante.....	84
4.3.	Amplitude de l'éloignement d'un mur	84
4.4.	Algorithme de suppression des couloirs inutiles.....	86

5. Gestion du handicap	88
5.1. Prise en compte du handicap	88
5.2. Illustration	90
5.3. Tableau récapitulatif des résultats des trois scénarios	103
6. Résultats expérimentaux pour les deux modèles de placement et dimensionnement des pièces	104
6.1. Présentation générale	104
6.2. Résultats du premier modèle	106
6.3. Résultats du modèle 2	108
6.4. Modèle 1 versus Modèle 2	110
7. Bilan	112
8. Discussion	112
Chapitre IV --- Amélioration d'un aménagement	114
1. Introduction	114
2. Modélisation d'un aménagement	117
2.1. Préambule	118
2.2. Description de la structure de données	118
2.2.1. Ouverture	118
2.2.2. Mur	119
2.2.3. Pièce	119
2.2.4. Plateau	119
2.2.5. Logement	120
2.3. Graphe de mobilité	120
2.3.1. Dépendance du graphe à l'habitant	122
2.3.2. Points caractéristiques	123
2.3.2.1. Points caractéristiques d'une porte	124
2.3.2.2. Points caractéristiques d'une pièce	126
2.3.2.2.1. Sommets concaves	126
2.3.2.2.2. Points de départ de la pièce	126
2.3.3. Graphe de mobilité complet de l'aménagement	130
2.4. Graphe d'adjacence	131
3. Evaluation d'un aménagement	132
3.1. Les opérateurs de bas niveau pour la vérification des critères de validité	133
3.1.1. Longueur d'un segment	133
3.1.2. Intersection de deux segments	134
3.1.3. Connexité d'un graphe non orienté	134
3.1.4. Points d'articulation d'un graphe	134
3.1.5. Le plus court chemin entre deux pièces	135
3.2. Les critères d'évaluation	135
3.2.1. Critères réglementaires et géométriques	135
3.2.1.1. Non chevauchement des pièces	135
3.2.1.2. Aire minimale de la pièce principale	135
3.2.1.3. Existence d'une pièce	135
3.2.1.4. Pièce non borgne	136
3.2.1.5. Largeur minimale des portes	136
3.2.1.6. Aménagement accessible à une personne de corpulence dite normale	136
3.2.1.7. Zone d'intérêt	137
3.2.2. Les critères topologiques	137
3.2.2.1. Existence d'une pièce	137
3.2.2.2. Aire d'une pièce	137
3.2.2.3. Orientation d'une pièce dans une direction donnée	139
3.2.2.4. Pièce non borgne	139
3.2.2.5. Porte commune à deux pièces	140
3.2.2.6. Compacité / Dispersion des pièces	140

3.2.2.7.	La maîtrise du coût de construction	140
3.2.2.8.	Forme raisonnable d'une pièce	141
3.2.2.9.	La maîtrise des angles des pièces	142
3.2.3.	Les critères sur les itinéraires	142
3.2.3.1.	Mobilité	142
3.2.3.2.	Logement accessible à l'occupant	144
3.2.3.3.	Enfilades	144
3.2.4.	Modélisation informatique des critères	144
3.3.	Evaluation de l'adaptation d'un aménagement à un occupant	144
4.	Modification d'un aménagement	145
4.1.	Les opérateurs de modifications de bas niveau	145
4.1.1.	Les opérateurs sur les pièces	145
4.1.1.1.	Créer un sommet.....	146
4.1.1.2.	Fusionner deux sommets / Scinder un sommet.....	146
4.1.1.3.	Déplacer un sommet	150
4.1.1.4.	Supprimer un sommet.....	150
4.1.1.5.	Permuter deux pièces.....	151
4.1.2.	Les opérateurs sur les portes.....	152
4.2.	Cheminement dans l'espace des solutions	152
5.	Méthodes de résolution	158
5.1.	Définitions	158
5.2.	Voisinage	160
5.3.	Pondération des opérateurs de modification	160
5.4.	Les méthodes implémentées	161
5.4.1.	Recherche locale aléatoire	161
5.4.2.	Les méthodes de descente	162
5.4.2.1.	Premier voisin améliorant : First Improvement Hill Climbing (FIHC).....	163
5.4.2.2.	Meilleur voisin améliorant : Best Improvement Hill Climbing (BIHC).....	164
5.4.3.	Le recuit simulé.....	164
5.4.3.1.	Origine	164
5.4.3.2.	Principe du recuit simulé.....	165
5.4.3.3.	Paramétrages et algorithme.....	166
6.	Le logiciel réalisé et les expérimentations	170
6.1.1.	Introduction	170
6.1.2.	Présentation générale	170
6.1.3.	Illustration du bon fonctionnement des opérateurs	171
6.1.4.	Influence des critères sur les agencements proposés	176
6.1.5.	Synthèse via amélioration	184
7.	Bilan	184
8.	Discussion	185
	Conclusion et perspectives	188
	Annexes	192
1	Segment parallèle à un autre	192
2	Intersection des parallèles à deux droites sécantes	192
3	Calcul d'un angle capable	193
	Bibliographie	196
	Liste des illustrations.....	205
	Liste des algorithmes.....	206
	Liste des tableaux.....	207
	Liste des figures.....	208

Introduction

Actuellement, quand les architectes, les thérapeutes, les ergonomes, les ergothérapeutes conçoivent l'aménagement d'un logement, ils collaborent ou interviennent à tour de rôle avec l'espoir de satisfaire les nombreuses contraintes induites par ce type de projet. Citons par exemple : les contraintes architecturales (non recouvrement des pièces, largeur suffisante des couloirs, accessibilité à tout point du lieu à partir de tout autre point, nécessité de placer certaines pièces sur des arrivées ou évacuations ...), les contraintes imposées par le handicap (largeur accrue des couloirs en cas de déplacement en fauteuil, nécessité d'assurer un effort quotidien minimum pour lutter contre le vieillissement, limitation des escaliers si le futur occupant souffre d'un asthme sévère, éloignement d'une pièce des murs mitoyens en cas de surdit   ...) et les souhaits exprimés par le futur occupant (minimiser certains trajets, maximiser l'éloignement entre deux pièces, imposer l'orientation d'une pièce ...). La complexité est telle qu'il est rare qu'ils arrivent à toutes les prendre en compte de manière satisfaisante.

Les travaux de cette thèse visent à concevoir un système d'aide à la décision pour accompagner intelligemment les experts dans l'aménagement d'un espace intérieur, en tenant compte des spécificités de ses futurs occupants, en particulier quand ils présentent un handicap. Cet aménagement consiste à adapter le placement, le dimensionnement et la forme des pièces, des couloirs et des portes aux possibilités physiques ou/et mentales du futur occupant ainsi qu'à ses souhaits ; il ne concerne pas la disposition ou l'adaptation du mobilier.

Pour aborder cette problématique, nous nous sommes appuyé sur la programmation mathématique en nombres entiers, sur des méthodes exactes et heuristiques classiques en optimisation combinatoire et sur des algorithmes géométriques.

Les principaux apports sont 1) une modélisation mathématique et une méthode de résolution d'une variante du problème du bin-packing, 2) une méthodologie opérationnelle et extensible pour la prise en compte informatique des contraintes de natures très variées nécessaires à la spécification d'un aménagement, 3) un ensemble d'opérateurs de modifications d'un logement, support indispensable de certaines de nos méthodes d'agencement, 4) des méthodes capables d'améliorer un agencement proposé par un architecte et 5) une méthode de synthèse d'un agencement à partir d'un ensemble de spécifications et sans indications préalables d'aucune sorte par l'architecte.

Ce mémoire comporte quatre chapitres, une introduction, une conclusion, quelques perspectives et des annexes.

Le premier chapitre parle du handicap, de l'accessibilité et des normes instaurées en ce domaine par la loi du 11 février 2005 pour l'égalité des droits et des chances. Nous retraçons le chemin qui a conduit à une redéfinition et à une reconnaissance du handicap comme étant un désavantage dont est victime un individu, l'empêchant de vivre dans la dignité ; le handicap est donc un problème posé par un environnement inadapté et non un problème posé par la personne. Nous évoquons ensuite, les problèmes d'aménagements d'un espace (aménagement du territoire et aménagement de l'intérieur). Nous nous focalisons sur l'aménagement des espaces intérieurs, en particulier, l'aménagement des lieux de vie pour les personnes handicapées.

Nous établissons ensuite un lien formel entre notre problématique et le problème de bin-packing à deux dimensions, ce qui permet de poser un premier cadre théorique à notre problématique.

Nous finissons en présentant plusieurs travaux en lien avec la problématique de l'agencement d'un logement à usage d'une personne en situation de handicap, en particulier le projet HM2PH (Habitat Modulaire et Mobile pour Personnes Handicapées) développé au sein du laboratoire d'informatique de l'Université de Tours.

La nature pluridisciplinaire du projet nous a amené à utiliser un nombre considérable d'algorithmes géométriques, par exemple, le test de convexité d'un polygone ou le test de la disjonction de deux polygones. Le Chapitre II, détaille l'ensemble des méthodes géométriques spécifiques à la gestion d'un logement. Cette partie, même si elle n'apporte pas de contributions majeures au domaine de l'algorithmique géométrique, reste nécessaire à la compréhension des modèles et algorithmes développés dans le Chapitre IV, qui est dédié à l'optimisation d'un agencement.

La **synthèse** d'un plan est évoquée dans le Chapitre III. Elle a été abordée comme un problème de type « bin-packing », réputé NP-difficile, avec des contraintes additionnelles (conteneur de taille fixe et objets à placer - les pièces - de tailles variables et soumis à des contraintes fonctionnelles). La méthode de résolution s'appuie sur un premier modèle mathématique, qui prend la forme d'un programme quadratique (linéarisé par la suite) en variables mixtes. Elle a été appliquée avec succès pour placer les pièces d'un logement, pour les dimensionner, pour déterminer les couloirs assurant une complète **accessibilité** au logement et pour prendre en compte certaines **contraintes imposées par le handicap** du futur occupant. Un deuxième modèle mathématique a été élaboré pour le placement des portes et une heuristique a enfin été développée pour affecter l'espace occupé par les couloirs non indispensables aux pièces avoisinantes. La totalité de cette démarche a été programmée dans un prototype logiciel pleinement opérationnel.

Dans le quatrième et dernier chapitre, nous parlons de la deuxième contribution de cette thèse, à savoir l'**optimisation d'un agencement existant**. Cette optimisation a été conçue comme un processus itératif enchaînant évaluation et modification (amélioration) d'un agencement. Il est décliné de quatre manières : une métaheuristique de type « recuit simulé » et trois méthodes de type « recherche locale », qui explorent l'espace des solutions en utilisant des voisinages spécialement définis. Cette approche a d'une part permis d'appréhender le caractère multicritère de cette problématique et a d'autre part exigé la mise en œuvre de nombreux algorithmes géométriques. Ces travaux sont implantés dans un deuxième prototype logiciel.

Chapitre I --- Etat de l'art

1. Introduction

L'objectif de ce chapitre est de donner une vue d'ensemble de la problématique traitée dans cette thèse, à savoir l'aménagement d'un environnement intérieur pour le rendre adapté aux futurs occupants, en particulier quand ces derniers sont handicapés ou simplement quand ils deviennent âgés.

Nous commençons (section 2) par retracer le chemin qui a conduit à une redéfinition et une reconnaissance du handicap, comme étant un désavantage dont il est victime un individu pour vivre dans la dignité, c'est à dire que c'est un problème d'environnement inadapté à la personne et non un problème qui vient de la personne.

Un environnement peut être un habitat, un établissement recevant du public (ERP) ou un territoire (au sens administratif : un quartier, une ville, une région) ; c'est ce que nous appelons l'espace (de déplacement) dans la suite. Nous évoquerons dans la section 3 les problèmes d'aménagement de ces espaces. Dans tous les cas, on souhaite réorganiser l'espace en supprimant tous les obstacles aux déplacements d'une personne. Nous énumérons également, en fonction de la nature de l'espace et dans le respect de la loi du 11 février 2005 pour l'égalité des droits et des chances, les besoins d'une personne handicapée, en termes d'aménagement de l'espace.

La section 4 établit un lien formel entre notre problématique et le problème de bin packing, en particulier celui de l'agencement. Ce qui permet d'avoir un premier cadre théorique à notre problématique. Un problème d'agencement est un problème qui prend en compte à la fois toutes les contraintes classiques qu'on retrouve dans un problème de bin-packing (non chevauchement et appartenance) et les contraintes fonctionnelles, des contraintes qui relient les objets entre eux (par exemple, un objet a doit être à une distance donnée d'au plus 10m de l'objet b). La fonction objectif d'un problème d'agencement est une fonction qui est souvent une fonction multi-objectif, tout le contraire des autres problèmes de placement où l'objectif ultime est la maximisation de la compacité, une fonction qui est mono-objectif.

La dernière section évoquera plusieurs travaux existant. Nous mettrons en particulier l'accent sur le projet HM2PH, le seul projet qui a traité le problème d'aménagement pour les personnes handicapées.

2. Handicap

2.1. Etymologie du mot handicap

D'après le CNRTL¹, le mot *handicap* est une contraction probable de l'expression anglaise « *hand in cap* » qui signifie la « *main dans le chapeau* ». Cette expression découle d'un troc de biens entre deux personnes qui se pratiquait en Grande Bretagne au 16^{ème} siècle. Le « *handicapper* », qui est l'arbitre,

¹ CNRTL : Centre National de Ressources Textuelles et Lexicales

déterminait la différence de valeur des deux objets, puis il demandait au propriétaire dont l'objet a moins de valeur de déposer la somme compensant la différence dans un chapeau.

L'expression s'est progressivement transformée en un seul mot et a été employée aux courses de chevaux d'où la définition donnée par les dictionnaires classiques « *une course ouverte à des chevaux dont les chances de vaincre, naturellement inégales, sont, en principe, égalisées par l'obligation faite aux meilleurs de porter un poids plus important (courses au galop) ou de parcourir une distance plus longue (courses au trot)* ». En hippisme, un handicap a pour principe de donner autant de chances à tous les concurrents en ajoutant de la difficulté aux meilleurs. C'est donc un système où l'on recherche l'équité.

2.2. Définition du handicap

L'article 2 de la loi du 11 février 2005 précise que « *constitue un handicap(...) toute limitation d'activité ou restriction de participation à la vie en société subie dans son environnement par une personne en raison d'une altération substantielle, durable ou définitive d'une ou plusieurs fonctions physiques, sensorielles, mentales, cognitives ou psychiques, d'un polyhandicap ou d'un trouble de santé invalidant* ». Selon santé publique France [39] « *le handicap se présente sous de multiples formes et n'a pas de définition univoque. Il est à distinguer de la maladie ou de l'accident, qui peuvent être à l'origine du handicap, définitif ou passager. Par exemple la dégénérescence de la macula (partie de la rétine qui permet la vision fine) entraîne des difficultés à voir donc à lire, à faire certaines activités et finalement à sortir seul, d'où des répercussions sociales qui limitent l'individu dans ses relations avec l'environnement. On met désormais en évidence cette notion d'interaction entre les caractéristiques individuelles d'une personne (la déficience) et son environnement* ».

2.3. Vers une classification internationale du handicap

La classification internationale du handicap dans sa version actuelle a connu un long chemin qui a commencé à la fin du 19^{ème} siècle avec la première référence terminologique internationale du handicap apparue dans la Classification Internationale des Maladies (CIM), en passant par une première vraie classification qui est la Classification Internationale des Handicaps (CIH) dans les années 80, jusqu'à arriver à la Classification Internationale du Fonctionnement, du Handicap et de la Santé (CIF) adopté en 2001 par l'OMS.

Ce chemin a permis de redéfinir le handicap au fil du temps jusqu'à prendre en compte le contexte social dans lequel l'individu évolue en plus de l'état de santé et des capacités de ce dernier.

2.3.1. Classification Internationale des Handicaps (CIH/CIM)

En 1980, l'OMS² publie un projet expérimental de classification des handicaps intitulé CIH élaborée par un groupe de travail initialement animé par le Dr. Philip Wood. Cette dernière donne une vision radicale du handicap en le définissant comme un désavantage dont est victime un individu pour accomplir son rôle social normalement du fait de sa déficience (temporaire ou définitive) ou de son incapacité (partielle ou totale). Cette définition vient compléter le CIM, en étudiant l'effets dans la vie de l'individu des désavantages sociaux. Contrairement au CIM, ce classement introduit, pour la

² OMS : Organisation mondiale de la santé

première fois, les notions d'environnement social pour l'identification des processus d'invalidités comme l'indique la figure ci-dessus :

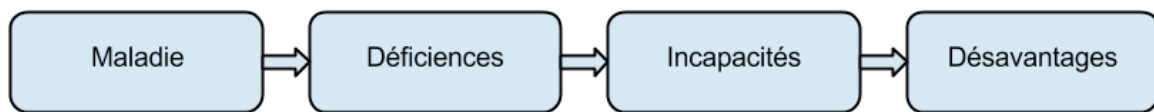


Figure 1. Classification internationale du handicap adoptée par l'OMS en 1980

La déficience correspond à toute perte ou altération d'une fonction ou d'une structure psychologique, physiologique, ou anatomique.

L'incapacité est la conséquence de la déficience d'un organe ou d'un système sur le fonctionnement de l'individu en termes de limitation de fonctions ou de restrictions d'activités.

Le handicap (ou le désavantage) est le préjudice qui résulte de la déficience ou de l'incapacité et qui limite ou interdit qu'il accomplisse son rôle normal compte tenu de son âge, son sexe....

La causalité linéaire de la déficience au désavantage, l'insuffisante description des désavantages et l'absence de la dimension environnementale sont les principales raisons de la contestation de cette classification par de nombreux acteurs et associations à l'échelle internationale, estimant que les situations handicapantes sont dues aux obstacles environnementaux, économiques et sociaux qu'un individu ne peut surmonter de la même manière que les autres citoyens.

2.3.2. Classification Internationale du Fonctionnement, du Handicap et de la Santé (CIF/CIM-10)

Une nouvelle classification a été adoptée en 2001 par l'OMS. Dans cette classification intitulée Classification Internationale du Fonctionnement, du handicap et de la santé (CIF), les problèmes de santé (maladie, trouble, lésions et traumatismes) sont classés essentiellement en fonction de la CIM-10³, qui fournit un cadre étiologique (Figure 2) [61].

Le fonctionnement et le handicap associés aux problèmes de santé sont classés dans la CIF. La CIM-10 et la CIF sont par conséquent complémentaires. La CIM-10 permet de poser un diagnostic des maladies, troubles et autres problèmes de santé, cette information est enrichie par les informations supplémentaires apportées par la CIF sur le fonctionnement. Ainsi, ces données permettent de fournir une base scientifique pour étudier les états de santé, les conséquences et leurs déterminants. Cette classification est un repère de la comparaison entre les états et un langage commun pour décrire et de manière claire les états de la santé et les états connexes de la santé afin d'améliorer les échanges entre les différents acteurs (travailleurs de santé, chercheurs, décideurs et le public en général, y compris les personnes handicapées).

³ CIM-10 : Classification Internationale de maladies (10^{ème} révision)

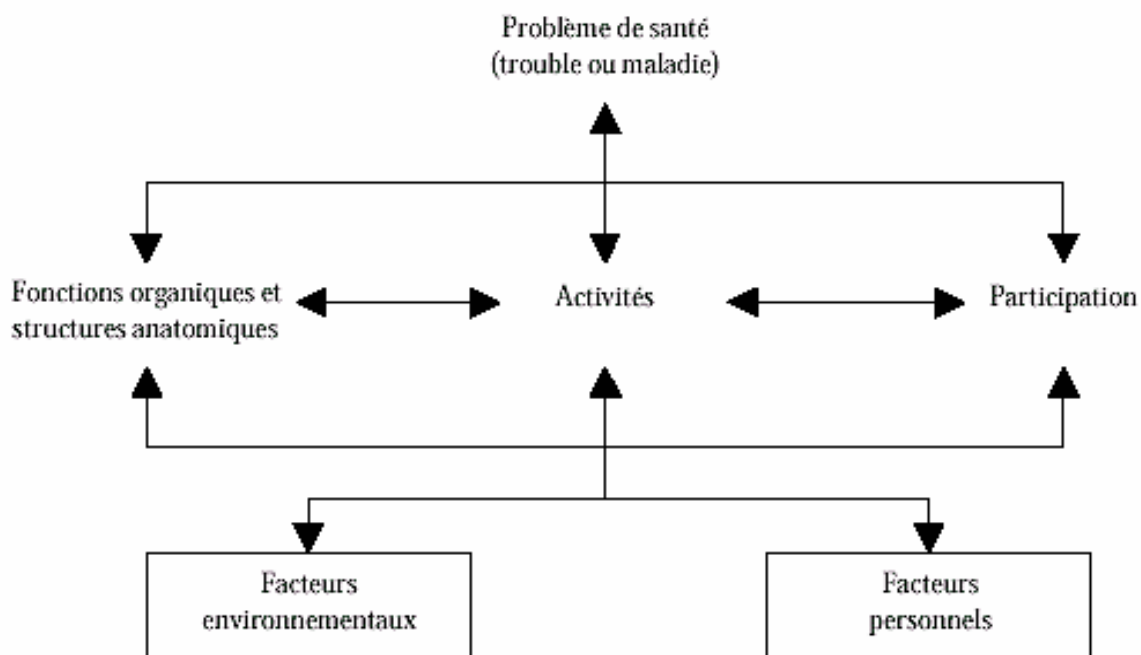


Figure 2. Classification Internationale du Fonctionnement, du Handicap et de la Santé adopté par l'OMS en 2001

2.4. Vers un processus d'automatisation en France

2.4.1. Loi du 30 juin 1975 d'orientation en faveur des personnes handicapées

Avec la loi du 30 juin 1975 concernant les personnes handicapées, la France a mis en place des normes détaillées d'accessibilité. Ces normes visent principalement les personnes ayant un handicap physique. Par exemple, le chapitre V, article 49 ordonne que « *Les dispositions architecturales et aménagements des locaux d'habitation et des installations ouvertes au public, notamment les locaux scolaires, universitaires et de formation doivent être tels que ces locaux et installations soient accessibles aux personnes handicapées. Les modalités de mise en œuvre progressive de ce principe sont définies par voie réglementaire dans un délai de six mois à dater de la promulgation de la présente loi.* ». L'article 52 du même chapitre prévoit des normes d'accès aux infrastructures de transports « *Afin de faciliter les déplacements des handicapés, des dispositions sont prises par voie réglementaire pour adapter les services de transport collectif ou pour aménager progressivement les normes de construction des véhicules de transport collectif, ainsi que les conditions d'accès à ces véhicules ou encore pour faciliter la création et le fonctionnement de services de transport spécialisés pour les handicapés ou, à défaut, l'utilisation des véhicules individuels* ».

La loi est votée à l'unanimité, elle est considérée comme innovante en son temps. Elle a permis d'affirmer des droits pour les personnes handicapées. Aujourd'hui cette loi est plus critiquée. D'abord, parce qu'elle prévoit l'accessibilité aux infrastructures communes mais ne se prononce pas par rapport à l'accessibilité aux soins, l'éducation, la formation, l'emploi, etc. Un autre point très critiqué est la non reconnaissance des personnes ayant des déficiences intellectuelles.

2.4.2. Le droit à la compensation : la loi du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées

La loi du 11 février 2005 vient réactualiser la loi du 30 juin 1975. Elle s'inscrit dans le cadre de la non-discrimination et l'égalité des chances pour tous. La volonté d'une inclusion sociale à la place d'une

intégration sociale, la reconnaissance de la pleine citoyenneté des personnes handicapées et l'intégration de la notion de situations handicapantes sont les principales nouveautés de cette loi. On évoque ci-dessus les dispositions concrètes de la loi qui nous semblent en lien direct avec notre thème de recherche [35] :

1. Cadre bâti, transports et nouvelles technologies :
 - accessibilité des établissements recevant du public ;
 - prise en compte de tous les handicaps dans les constructions nouvelles et les réhabilitations ;
 - les services de transports collectifs devront être rendus accessibles dans les 6 ans aux personnes handicapées ;
 - contrôles (délai de mise en accessibilité : 10 ans) et sanctions pénales prévus ;
 - accessibilité à l'information et aux savoirs : site internet, reconnaissance de la langue des signes, etc.
2. Accueil et information des personnes handicapées : évaluation de leurs besoins, reconnaissance de leurs droits par :
 - la maison départementale des personnes handicapées : donne un accès unique aux droits et prestations et oriente vers les établissements et services adéquats ;
 - la commission des droits et de l'autonomie des personnes handicapées.
3. Emploi :
 - des aménagements de postes de travail à prévoir.

2.5. Les différents types de handicap

La loi du 11 février 2005 a dressé la liste des champs sur lesquels peut porter le handicap ou le polyhandicap : fonctions physiques, sensorielles, mentales, cognitives ou psychiques [15,13].

Selon le Comité national Coordination Action Handicap :

Le handicap physique est la conséquence de troubles physiques⁴ qui peuvent entraîner une atteinte partielle ou totale de la motricité, notamment des membres supérieurs et / ou inférieurs (difficultés pour se déplacer, conserver ou changer une position, prendre et manipuler, effectuer certains gestes). Les causes peuvent être très variées : maladie acquise ou génétique, malformation congénitale, traumatisme dû à un accident, vieillissement

Le handicap sensoriel est la conséquence d'une atteinte d'un ou plusieurs sens. Les plus connus ouvrant le droit à la compensation, sont les atteintes de la vue et de l'ouïe.

Le handicap mental est la conséquence d'une déficience intellectuelle. Cette dernière peut être considérée comme une capacité plus limitée d'apprentissage et un développement intellectuel significativement inférieur à la moyenne. Il se traduit par des difficultés plus au moins importantes de réflexion, de conception, de communication, de décision etc.

Le handicap cognitif est la conséquence de dysfonctionnements des fonctions cognitives⁵ : troubles de l'attention, de la mémoire, de l'adaptation au changement, du langage, des identifications

⁴ Troubles de dextérité, paralysie...

⁵ Selon la période d'apparition de ces dysfonctionnements, on parlera de troubles « du développement » (qui surviennent entre la période prénatale et la petite enfance, tels que les

perceptives (gnosies) et des gestes (praxies). Ce type de handicap ne touche pas les déficiences intellectuelles mais plutôt des difficultés à mobiliser ses capacités.

Le handicap psychique est la conséquence de troubles psychiques⁶ invalidants. Comme le handicap cognitif, ce handicap ne touche pas les facultés intellectuelles, il se caractérise par l'alternance des états psychiques calmes ou tendus et par des difficultés à acquérir ou à exprimer des habilités psychosociales.

Le polyhandicap est la conséquence d'un handicap grave à expressions multiples, dans lequel, souvent, une déficience mentale sévère et une déficience motrice sont associées à la même cause, entraînant une restriction extrême de l'autonomie.

Dans ce qui vient de précéder, nous avons retracé le chemin qui a conduit à une redéfinition du handicap, comme étant un désavantage, de l'environnement, dont il est victime un individu pour vivre dans la dignité. Ensuite, nous avons donné une définition de tous les types de handicap au sens de la loi du 11 février 2005.

Nous pouvons maintenant aborder les problèmes d'aménagement spatial pour répondre aux besoins des personnes en situation handicapante.

3. Aménagement spatial

L'aménagement est l'action de modifier un objet ou un lieu dans le but de permettre à un plus grand nombre de personnes de l'utiliser à ce pourquoi il est prévu. Un aménagement spatial peut être l'aménagement d'un territoire (commune, ville, région...) ou l'aménagement d'un intérieur (maison, immeuble, hôpital...).

L'aménagement d'un territoire est une démarche volontariste de l'État, qui a pour objectif la répartition harmonieuse des ressources du territoire. Cette politique, qui s'inscrit dans une démarche globale et durable dans le temps, répond moins que la politique usuelle aux lois économiques du simple jeu de marché. Elle se veut plutôt correctrice des déséquilibres géographiques entre les régions, les villes et campagnes, les zones dynamiques et les zones déshéritées etc. Un aménagement d'un territoire peut être un projet qui consiste, par exemple, à repenser le réseau de transport pour un territoire plus écologique, en rendant les transports en commun plus adaptés que la voiture ; à construire un ou plusieurs hôpitaux, pour améliorer la prise en charge des patients ; à construire un musée, pour rendre le territoire plus attractif aux touristes etc.

L'aménagement de l'intérieur est la transformation, la rénovation ou la décoration d'un cadre bâti pour le rendre plus adapté aux personnes qui l'occupent. Dans le cadre de l'aménagement d'un logement, la démarche est souvent non imposée par des circonstances extérieures : l'habitant souhaite simplement rendre l'espace adapté à ses goûts. Mais parfois cette démarche devient une démarche nécessaire car elle permet à l'habitant de continuer à occuper son logement, suite à un

dyslexies, dysorthographies, dyscalculies, dysgraphies, etc.) ou de troubles acquis (séquelles d'un traumatisme crânien, d'un accident vasculo-cérébral, telles que les alexies agnosies, etc.)

⁶ Également appelés maladies mentales, parmi lesquelles on trouve notamment les schizophrénies, les troubles dépressifs graves et les troubles bipolaires, les troubles anxieux, les troubles de la personnalité, etc.

accident de la vie qui a provoqué un lourd handicap et qui a rendu son logement inhabitable par exemple. L'aménagement d'un commerce, d'une administration... est souvent une démarche nécessaire pour une mise en conformité (normes de sécurité, accessibilité pour les personnes handicapées...).

Dans les paragraphes qui suivent, nous abordons d'abord le problème de l'aménagement *durable* du territoire, étant donné que la notion de durabilité offre une vision plus globale d'un aménagement du territoire. Nous détaillons en particulier le problème d'accessibilité pour les personnes handicapées du fait que c'est un problème en lien direct avec le sujet de ce mémoire. Ensuite (3.2), nous traitons le problème de l'aménagement d'un lieu de vie pour les personnes handicapées. Nous abordons point par point les différents aspects à considérer d'un point de vue légal (loi du 11 février 2005 pour l'égalité et des chances). Et pour finir, nous faisons le lien entre les problèmes d'aménagement d'un espace intérieur et les problèmes d'agencement et de bin packing d'un point de vue formel.

3.1. Aménagement du territoire durable

L'aménagement durable du territoire consiste en un ensemble d'actions menées par l'état et les collectivités afin de favoriser le développement des régions. Le but est de mieux utiliser les espaces occupés (le territoire) pour mieux servir la population qui l'occupe, tout en préservant les ressources pour les générations futures (figure 3).

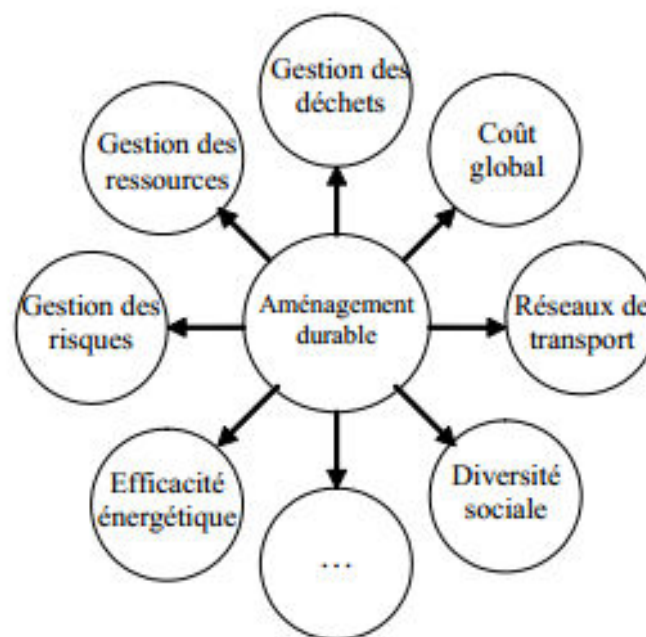


Figure 3. Aménagement durable et aspects concernés [17]

Le développement territorial durable impose de faire des choix dans un contexte caractérisé par la complexité.

Complexe dans le sens où aucune décision ne peut être réduite qu'à une seule dimension (par exemple, l'environnement, le social ou l'économique) (**Erreur ! Nous n'avons pas trouvé la source du envoi.**) [16]. Afin d'être *durable*, le développement territorial se doit de prendre en compte les différentes répercussions d'une décision : implanter une usine dans un village créera de l'emploi et rapportera un surcroît de ressources pour la municipalité mais aura aussi sûrement des impacts

environnementaux ; ne pas considérer les personnes handicapées coûtera peut être moins cher aux territoires mais favorisera l'exclusion sociale et donc le sentiment d'injustice chez cette catégorie de personnes, ce qui fera de ce territoire, un territoire inéquitable [16].

Un projet d'aménagement d'un territoire peut être considéré comme un problème de gestion des ressources dans lequel des sous problèmes d'optimisations peuvent être considérés. Par exemple, pour une santé accessible à tous les habitants d'un territoire, l'objectif serait de placer un ou plusieurs hôpitaux, de telle sorte que le temps nécessaire pour se rendre à l'hôpital le plus proche, pour chaque habitant, soit le plus court possible. Ce qui ramène l'aménagement d'un territoire dans ce contexte à un problème d'optimisation de type *localisation*. Un autre exemple d'un projet d'aménagement d'un territoire est celui de l'amélioration du réseau de transport, qui peut être modélisé par extension d'un problème classique comme celui du voyageur de commerce, du flot dans un réseau de transport ou encore d'un arbre couvrant dans un graphe.



Figure 4. Approche globale à la confluence des trois préoccupations, dites « les trois piliers du développement durable »

Comme évoqué auparavant, le développement durable d'un territoire nécessite la prise en compte de plusieurs aspects. Dans ce qui suit, nous nous concentrons sur l'accessibilité.

3.1.1. Aménagement du territoire pour l'accessibilité aux personnes handicapées

En zone urbaine un handicapé devient inévitablement un piéton, il s'agit de répondre à la diversité de ses besoins : se déplacer, marcher, traverser les rues... [21]. On parle ici de la chaîne de déplacement et c'est la voirie qui doit être aménagée.

Selon [21], cet aménagement a trois facettes :

- *Eclairage et contraste visuel* : le maintien d'un bon niveau de visibilité ou plus généralement en condition de visibilité dégradée est nécessaire pour une personne malvoyante, par exemple, afin de garantir la pérennité du déplacement. L'objectif étant d'avoir un espace cohérent avec un guidage et une identification des lieux pour permettre d'éviter les obstacles. Un bon éclairage nécessite un contraste adapté (Illustration 1), car sans contraste il n'y a pas

de relief et donc tout se fondrait dans tout. Le contraste visuel dépend à la fois des matériaux employés et des performances de l'installation de l'éclairage.

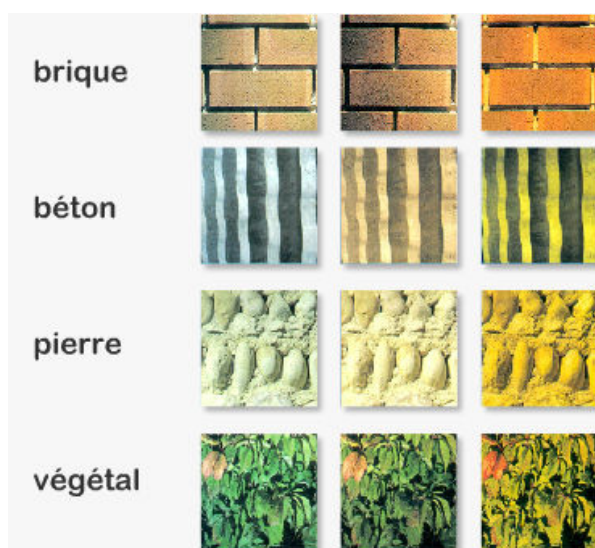


Illustration 1. Les niveaux des contrastes visuels en fonction des matériaux utilisés [21]

- *Déplacement* : pour se déplacer d'un point à l'autre, il faut prendre en compte les cheminements et les trottoirs, les escaliers, les équipements et mobiliers urbains et les stationnements réservés. D'après l'arrêté du 15 janvier 2007, il faut choisir un revêtement qui ne glisse pas avec une homogénéité du sol pour faciliter le roulage des fauteuils car il est assez fréquent que le sol soit mouillé. En cheminement courant, le dévers⁷ doit être inférieur ou égal à 2%. Un choix approprié de mobilier urbain permet d'offrir des sièges ou points d'appui assis-debout (Illustration 2) appréciés par les usagers qui s'essouffent ou se fatiguent à marcher sur de longues distances. Les emplacements réservés aux personnes handicapées sont répartis de manière homogène sur la totalité de la voirie de la commune selon un plan de zonage élaboré après avis de la commission communale.



Illustration 2. Exemple d'un point d'appui assis-debout [21]

⁷ Dévers : est la valeur de la pente transversale d'un des deux versants d'une chaussée ou d'un trottoir.

- *Poursuivre le déplacement* : les personnes handicapées ne disposent pas toutes d'une automobile ou de quelqu'un pour les véhiculer. Les transports collectifs sont donc essentiels pour certains. Pour cela, les arrêts doivent être accessibles et réalisés de façon à faciliter l'embarquement dans les véhicules. Les voiries doivent aussi être équipées de feux tricolores sonores pour permettre à une personne malvoyante de se déplacer sans danger.

3.1.2. Quelques initiatives pour un territoire accessible aux personnes handicapées

3.1.2.1. Grenoble : Projet INNOVACCES

La loi de 2005 ne désigne pas celui qui a la charge de fédérer les acteurs pour garantir le lien entre les différents maillons de la chaîne du déplacement. C'est la raison pour laquelle le projet INOVACCESS a été mis en place. Lancée le 22 janvier 2010, INOVACCESS est le fruit d'une coopération entre la ville de Grenoble, l'AGFIPH⁸ et le FIPHFP⁹. Ce projet, qui visait à créer les conditions d'une accessibilité continue Ville/Entreprise en agissant sur tous les maillons de la chaîne de déplacements, avait comme rôle :

- d'amener les différents acteurs à mobiliser leurs ressources pour travailler de manière coordonnée et complémentaire (vision globale), tout en restant, pour chacun, dans son champ de compétences ;
- de favoriser l'emploi des personnes à mobilité réduite à travers deux plans d'action :
 - ✓ aider les entreprises à réaménager leurs locaux en subventionnant une partie ou la totalité des travaux ;
 - ✓ faire assurer les travaux de la voirie et des espaces publics par la ville de Grenoble pour garantir la continuité de la chaîne de déplacement.

Le projet s'est inscrit dans une méthodologie qui s'adaptait au fil de l'eau car elle dépendait d'une démarche volontaire et libre des entreprises implémentées dans la zone visée. Cette méthodologie est divisée en trois phases comme suit :

- pré-audit + (le diagnostic + la programmation) de l'accessibilité des locaux, pris en charge à 100% ;
- avant-projet et études de la prise en charge à 100% ;
- une participation aux travaux mis en œuvre : la ville de Grenoble, pour les accès en ville, l'AGFIPH et la FIPHFP pour les locaux.

Le projet a duré 3 ans pour un budget de 3 millions d'euros. Il a permis à la ville de Grenoble de gagner le deuxième prix de l'ACCESS CITY AWARD [27], un prix qui reconnaît et honore les villes européennes qui s'efforcent d'assurer un environnement accessible pour tous, et notamment pour les personnes handicapées.

⁸ AGFIPH : Association de Gestion du Fonds pour l'Insertion professionnelle des Personnes Handicapées

⁹ FIPHFP : Fonds pour l'insertion professionnelle des personnes handicapées dans la fonction publique

3.1.2.2. Nantes : un plan d'actions handicap 2014-2020 pour répondre aux enjeux d'accessibilité

La ville de Nantes est considérée comme la ville la plus accessible en France justement parce qu'elle a adopté une politique d'intégration des citoyens en situation de handicap et ce depuis les années 90. Cette politique a été récompensée tant au niveau national qu'europpéen par « le baromètre de l'accessibilité » et « l'Access City Award ».

Aujourd'hui, la ville continue sur cette même lancée avec le plan d'action handicap 2014-2020 [47]. Celui-ci fixe les priorités, pour la période indiquée ci-dessus, comme suit :

- *favoriser l'accessibilité universelle* : la ville s'attache à rendre accessibles les espaces publics, les transports, ses établissements recevant du public (ERP). Elle incite également les commerçants à respecter de leurs obligations. Objectif : garantir la continuité de la chaîne des déplacements et le confort d'usage pour tous ;
- *construire une société inclusive* : pour permettre le développement et le plein exercice de la citoyenneté, la ville veille à assurer la continuité du parcours éducatif de l'enfant et du jeune en situation de handicap dès la petite enfance.

3.2. Aménagement d'un espace intérieur

Nous revenons à présent sur un point plus directement en rapport avec ce mémoire. L'aménagement de l'intérieur a pour objectif de disposer ses composantes afin d'optimiser l'espace, de décorer, ou de rendre accessibles certaines composantes. La loi du 11 février 2005 recense deux catégories d'espaces intérieurs, à savoir :

- *un habitat* : un lieu qui sert de logement, de lieu de vie (immeubles, maisons, appartements...);
- *un ERP* : un Etablissement Recevant du public (administrations, hôpitaux, commerces...).

Dans ce qui va suivre, nous allons mettre l'accent sur l'aménagement de l'intérieur pour une mise en conformité au sens de la loi de 2005 pour illustrer aux mieux les différents obstacles et règles à considérer lors d'un aménagement de l'intérieur, pour toute personne et non pas uniquement pour une personne handicapée car cette loi raisonne en termes d'obstacles à supprimer.

3.2.1. ERP : Etablissement recevant du public

Il s'agit de tous les bâtiments, locaux et enceintes dans lesquels des personnes (en plus du personnel) sont admises, soit librement, soit moyennant une rétribution ou une participation quelconque, ou dans lesquels sont tenues des réunions ouvertes à tous ou sur invitation payante ou non.

Pour ne pas rendre les bâtiments sélectifs, une prise en compte d'une population à mobilité réduite est nécessaire. Une personne handicapée doit pouvoir bénéficier des mêmes services que tout autre usager. L'objectif d'égalité des prestations vise à supprimer toute forme de discrimination et favoriser surtout l'indépendance et l'autonomie [21]. Il existe deux types aménagements pour un ERP :

- *environnement extérieur* : il prend en compte les trottoirs, les stationnements et les cheminements. Pour ces trois dimensions, les règles appliquées pour la voirie restent les mêmes grâce à une réflexion commune entre les diverses maîtrises d'ouvrage (commune, chef d'établissement, transporteur...) pour traiter l'accessibilité d'une manière globale et cohérente ;

- *à l'intérieur de l'établissement* : il doit prendre en compte les normes imposées pour les habitats (voir le point suivant) avec, en plus, les conditions d'accès aux services. C'est-à-dire que l'homogénéité du sol, l'absence de pentes dépassant 5%... doivent être respectés. Pour permettre le passage des pieds et des genoux d'une personne en fauteuil roulant, la tablette d'une banque d'accueil (Illustration 3) doit être à une hauteur variant de 0.70 m à un maximum de 0.80 m avec un vide en partie inférieure d'au moins 0.30 m de profondeur, 0.60 m de largeur et 0.70 m de hauteur.

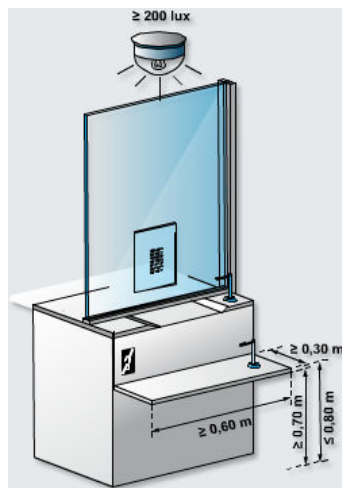


Illustration 3. Les dimensions à respecter d'une banque d'accueil [21]

D'autres considérations pour favoriser l'égalité sont à prendre en compte pour certaines catégories d'établissements comme les restaurants et les commerces. L'arrêté du 1^{er} Août 2006 impose un nombre d'emplacements pour les personnes à mobilité réduite : au moins 2 emplacements accessibles pour 50 places et un emplacement supplémentaire par tranche ou fraction de 50 places supplémentaires [60]. Au-delà de 1000 places, le nombre d'emplacements accessibles ne saurait être inférieur à 20. De plus, les établissements ayant des locaux d'hébergement doivent comporter des chambres adaptées. Le nombre de chambres est défini par tranches (1 chambre pour un établissement de moins de 20 chambres, 2 chambres pour un établissement de moins de 50 chambres et une chambre supplémentaire par tranche ou fraction de 50 chambres supplémentaires).

3.2.2. Habitat

Nous évoquons dans ce qui suit la mise en conformité d'un habitat à un occupant handicapé. L'objectif est de concevoir un environnement de qualité, adapté aux usagers et qui prend en compte les spécificités de l'habitant, de sorte qu'il soit autonome quand il est à l'intérieur [21].

D'après l'arrêté du 1^{er} Août 2006 [60], un Habitat est composé de :

- *les abords des bâtiments* : le stationnement, le cheminement vers ou depuis l'extérieur et l'entrée dans le bâtiment. La réglementation prévoit un nombre de places de stationnement minimum de 5% du nombre total de places prévu pour les occupants. Elle impose une signalisation et un type de sol qui ne crée aucune situation de pénibilité pour le cheminement vers et depuis l'extérieur et un système d'ouvertures des bâtiments adapté à tous (Illustration 4) ;

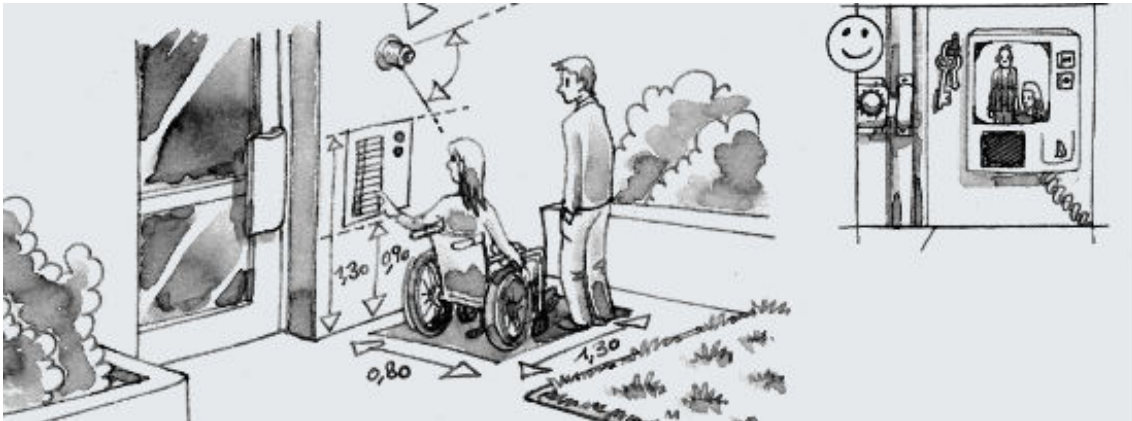


Illustration 4. Les angles à respecter pour un système adapté d'ouverture des bâtiments [21]

- *les parties communes intérieures* : les locaux collectifs à l'intérieur du bâtiment, le cheminement horizontal, l'ascenseur et les escaliers. Les personnes à mobilité réduite doivent pouvoir accéder à l'ensemble des locaux collectifs ainsi qu'aux caves et celliers... par un cheminement accessible. La largeur de cheminement intérieur doit permettre à deux personnes de se croiser dans les couloirs sans qu'il y ait une situation de gêne pour aucune d'elles. Lorsqu'une pente est inévitable, même faible, elle reste très pénible pour une personne s'aidant d'une canne ou en fauteuil ; dans ce cas la pente doit être compensée par l'installation d'un appareil élévateur. Pour l'ascenseur, les commandes doivent être adaptées aux aptitudes des personnes ayant une déficience sensorielle ou motrice. Quand l'ascenseur n'est pas obligatoire ou en panne, l'escalier reste le seul moyen pour atteindre les différents niveaux du bâtiment donc l'arrêté du 1^{er} Août 2006 impose des mains courantes espacées d'au moins 1 m (Illustration 5) ;

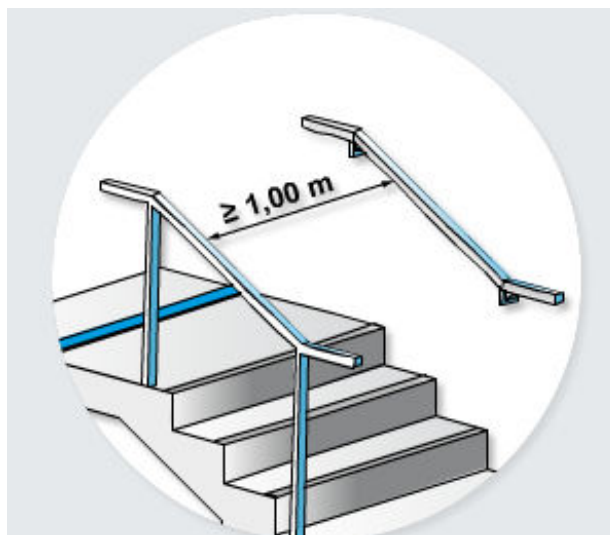


Illustration 5. La largeur minimale d'une main courante [21]

- *l'accès aux pièces* : la circulation, les portes, les équipements à l'intérieur du logement. Tous les logements doivent avoir une porte principale d'une largeur minimale de 0,9 m, les autres portes doivent avoir une largeur minimale de 0,8 m et le ressaut ne doit pas dépasser la hauteur

maximale de 2 cm pour toutes les portes. En cuisine, une personne en fauteuil roulant doit pouvoir se mouvoir sans difficulté entre les parois et les appareils indispensables. Les chambres et les salles d'eau doivent permettre une rotation du fauteuil roulant (Illustration 6). Et pour finir, les toilettes doivent permettre d'accéder à la cuvette du WC et de faire un transfert fauteuil-WC sans être gêné par la roue du fauteuil.

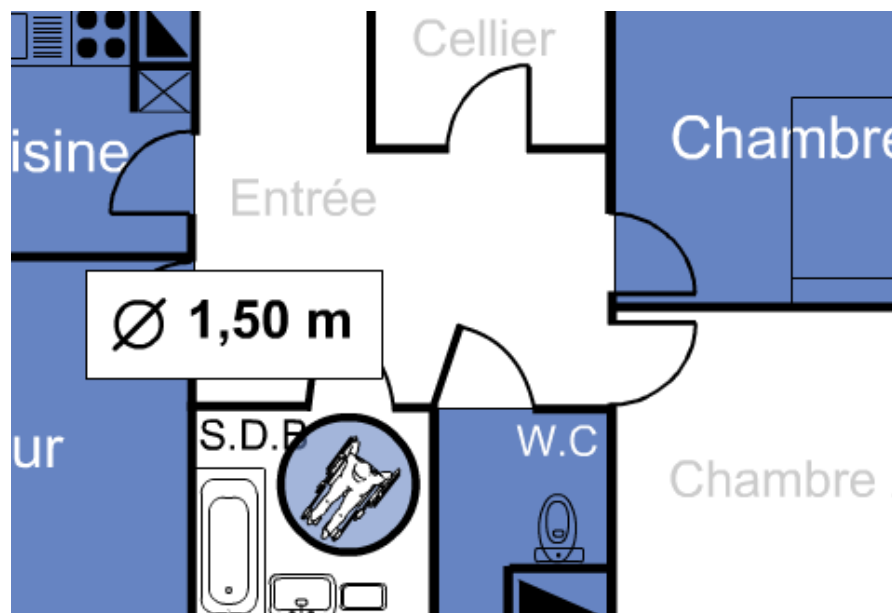


Illustration 6. Le diamètre minimal à respecter dans une salle de bain [21]

On a exposé quelques normes imposées par l'état français afin de rendre un lieu de vie accessible aux personnes handicapées. Pour le reste des normes, il existe des guides et référentiels. Ce sont des supports indispensables pour les professionnels du bâtiment ; chaque norme y est décrite de manière précise. On retrouve par exemple : « *le guide du représentant des personnes handicapées dans les commissions d'accessibilité* » [28] réalisé par la FNATH avec le soutien de la Poste et de la DGCS¹⁰ ; nous l'avons utilisé pour nos travaux. Citons aussi « *Adaptation des logements existants au handicap et au vieillissement* » [1] réalisé par l'ANAH¹¹.

Dans le Chapitre IV, nous verrons plus en détails ces normes, que nous utiliserons dans notre approche pour réaménager un lieu de vie afin de l'adapter aux personnes handicapées.

¹⁰ DGCS : la Direction générale de la cohésion sociale

¹¹ ANAH : Agence nationale de l'habitat

4. Diverses approches de l'agencement

4.1. Introduction

D'un point de vue formel, l'aménagement d'un lieu de vie présente une ressemblance évidente avec le problème de bin-packing à deux dimensions, à cette différence que les objets sont de formes et tailles variables (ce sont les pièces et leurs paramètres - longueur, aire ... - sont définis dans des intervalles au lieu d'être fixés au préalable) et que le contenant est unique, de taille connue et fixe. Dans sa version traditionnelle, le bin-packing est un problème combinatoire réputé NP-difficile. Nous conjecturons que c'est aussi le cas dans notre variante (conteneur imposé et objets de tailles variables).

Dans l'étude bibliographique qui suit, nous faisons d'abord une rapide synthèse des travaux sur la version classique du *bin-packing* et sur le problème voisin du *placement*. Le placement est très étudié dans la littérature. Il regroupe une gamme très large de thèmes, à savoir la découpe et le conditionnement (*C&P*) et l'agencement (*Layout*). Les applications industrielles et les enjeux sont considérables : on peut citer la découpe de matières premières (bois, tissus...) pour limiter les coûts de découpe et les chutes générées [53], le chargement de conteneurs pour minimiser par exemple le nombre de containers pour un nombre donné d'objets à transporter [22,81], le placement des composants électroniques [30,41,52], les problèmes d'agencement d'atelier [23], les problèmes d'agencement des pièces d'un logement [49,68] etc.

Une présentation plus détaillée des travaux directement en rapport avec notre étude est donnée dans le Chapitre I.5.

4.2. Typologies et caractéristiques des problèmes de placements

La première typologie des problèmes de découpe et de placement a été proposée par Dyckhoff en 1990 [24]. Elle est basée sur quatre paramètres, qui sont :

- la dimension du problème :
 - 1 problème unidimensionnel
 - 2 problème bidimensionnel
 - 3 problème tridimensionnel
 - N problème multidimensionnel avec $N > 3$
- le type de tâches :
 - B tous les contenants et une sélection des composants
 - V une sélection des contenants et tous les composants
- les caractéristiques du bin (container) :
 - O un seul bin
 - I plusieurs bins identiques
 - D plusieurs bins différents
- les caractéristiques des objets :
 - F quelques objets
 - M plusieurs objets relativement différents
 - R plusieurs objets relativement identiques
 - C objets identiques

La Figure 5 présente un découpage des problèmes C&P retrouvés en industrie, proposé par Dyckhoff.

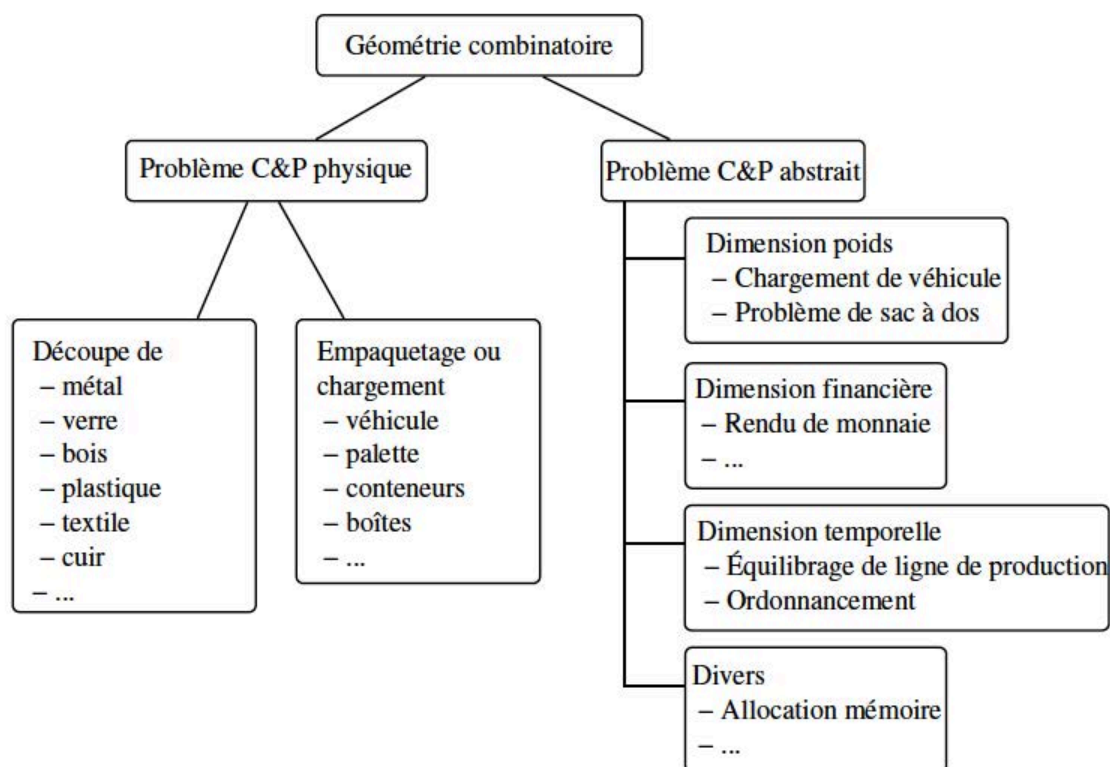


Figure 5. Typologie des problèmes de découpe et de conditionnement proposé par Dyckhoff

[24]

Quinze ans plus tard, Wäscher et al. [84] proposent une typologie plus complète que celle de Dyckhoff. Dans la nouvelle typologie, on voit apparaître de nouvelles problématiques de découpe et de conditionnement. Cette typologie est basée sur cinq critères pour composer le nom d'un problème de la famille des C&P. il s'agit de :

- CR1 : dimensions du problème (une, deux, trois dimensions) ;
- CR2 : différents objectifs à optimiser (maximisation sur les composants pour un nombre donné de contenants ou minimisation du contenant pour un nombre donné de composants) ;
- CR3 : différentes corrélations entre les composants (tous identiques, faiblement hétérogènes) ;
- CR4 : différentes corrélations entre les contenants (un ou plusieurs contenants de dimensions fixes, variables...) ;
- CR5 : différentes formes d'objets considérées (rectangle, cercle... dans le cas d'un problème à deux dimensions).

La Figure 6 définit six classes de problèmes de découpe et de conditionnement basées sur les critères de Wäscher.

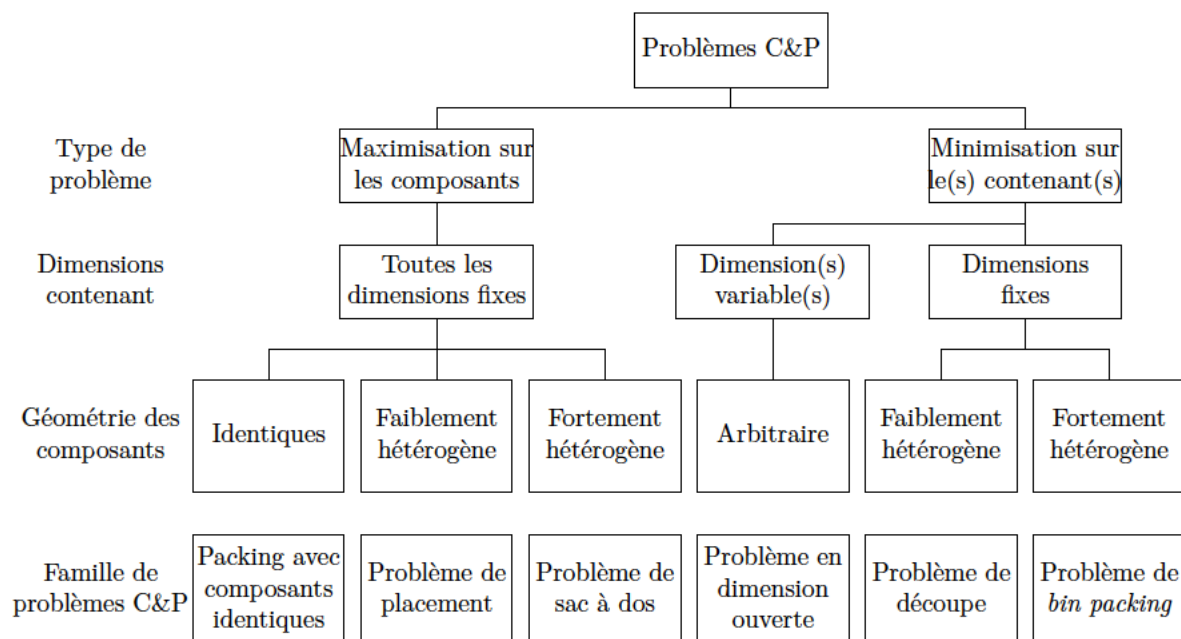


Figure 6. Classification des problèmes de découpe et de conditionnement proposée par Wäscher [84,40]

Dans cette classification, on retrouve le problème de placement, défini comme étant un problème de maximisation du nombre de composants faiblement hétérogènes dans un ou plusieurs contenants de tailles fixes - qui peut être considéré aussi comme étant un problème de minimisation de l'espace perdu -. Le problème traité dans cette thèse est un problème différent qui ne retrouve pas naturellement dans cette classe de problèmes ni dans les autres classes de la figure ci-dessus. Notre problème est un problème d'agencement dans lequel le nombre de contenants (un seul contenant qui est le contour extérieur du logement) ainsi que le nombre de composants (qui sont les pièces) sont fixes ; les dimensions des composants sont variables et la fonction objectif est exprimée comme étant la maximisation d'une fonction d'évaluation de la qualité de l'agencement. Néanmoins, chacune des classes de la famille des problèmes C&P présente plusieurs similitudes avec notre problème, qui nous seront utiles pour le modéliser et le résoudre. À titre d'exemple, l'écriture des contraintes de chevauchement et d'appartenance est presque la même pour l'ensemble de ces problèmes et les méthodes de résolutions pour les problèmes de C&P peuvent inspirer fortement les méthodes de résolution des problèmes d'agencement.

4.3. Définition

Un problème d'agencement est un problème NP-Difficile [76], dans lequel la position relative des composants dans les contenants est une inconnue du problème. Les composants sont reliés à la fois géométriquement (chevauchement, appartenance...) et fonctionnellement (proximité entre les composants, placement dans des zones en particulier...).

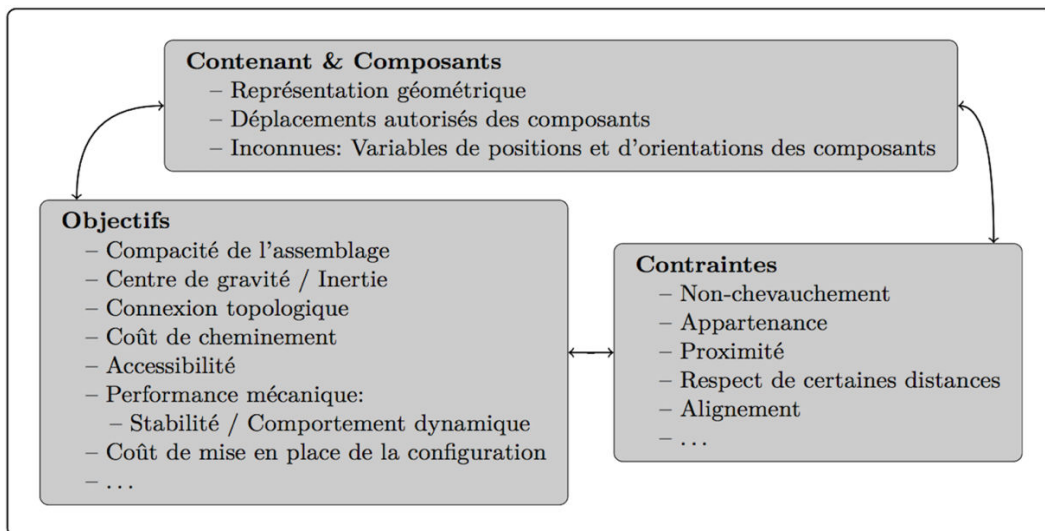


Figure 7. Constituants principaux d'un problème d'agencement [40]

La Figure 7 présente les différents éléments qui composent un problème d'agencement. Les objectifs sont différents et sont dans la majorité des cas multicritères, ce qui n'est pas le cas pour les problèmes de C&P qui ont souvent comme objectif l'amélioration de la compacité - soit il faut maximiser le nombre de composants dans le contenant ou minimiser le nombre de contenants pour un nombre de composants donné -.

Les méthodes de résolution proposées dans la littérature, pour résoudre un problème d'agencement, sont la plupart du temps des méthodes qui ramènent le problème multi-objectif à un problème mono-objectif, du fait que les critères d'évaluation sont souvent antagonistes, à l'aide de techniques de pondération des critères comme par exemple la méthode AHP (*Analytical Hierarchy Process*) [74] et MACBETH (*Measuring Attractiveness by Categorical-Based Evaluation Technique*) [4], qui sont des méthodes de comparaisons par paires [12,87], qui consistent à faire une analyse par paires des critères pour attribuer à chacun un poids selon son importance. On peut citer aussi la méthode des cartes Simos, qui ordonne les critères pour faire ressortir l'écart d'importance entre les critères [72,73,79].

Plusieurs exemples de cas pratiques d'agencements sont étudiés dans la littérature, parmi lesquels on retrouve :

- la conception de circuits électroniques VLSI (*Very Large Scale Integrated*), qui consiste à placer les composants (transistors, résistance, condensateur...) pour améliorer les performances globales du circuit, réduire sa taille... ;
- l'agencement d'ateliers (*Facility Layout problem*), qui consiste à optimiser les interactions entre les composants à agencer pour une meilleure fluidité et ainsi une meilleure exploitation des lieux ;
- le problème traité dans cette thèse, qui est l'aménagement d'un lieu de vie pour l'adapter aux personnes qui l'occupent ;
- ...

4.4. Formulation du problème et nature des contraintes à respecter

Nous verrons, dans les chapitres de propositions, que nous avons abordé le problème d'aménagement d'un logement de deux manières différentes. Dans la première, nous avons fait l'hypothèse que le logement et toutes ses pièces sont rectangulaires. Dans ce cas, le problème peut être formalisé de la manière suivante :

Soit C un contenant de dimensions fixes (DX, DY) et $I = \{1, \dots, n\}$ l'ensemble d'objets tel que pour chaque couple $(i, j) \in I^2$ les dimensions Dx_i et Dx_j sont définies respectivement dans les intervalles $[Dx_{iMin}, Dx_{iMax}]$ et $[Dx_{jMin}, Dx_{jMax}]$. Le problème d'agencement bidimensionnel avec un seul contenant consiste à agencer l'ensemble des objets I à l'intérieur du contenant en optimisant les interactions entre les objets tout en respectant les contraintes géométriques et fonctionnelles.

Les *contraintes géométriques* sont les contraintes élémentaires qu'on retrouve dans tous les problèmes de C&P, à savoir les contraintes de non-chevauchement entre les objets et d'appartenance des objets au contenant (Figure 8).

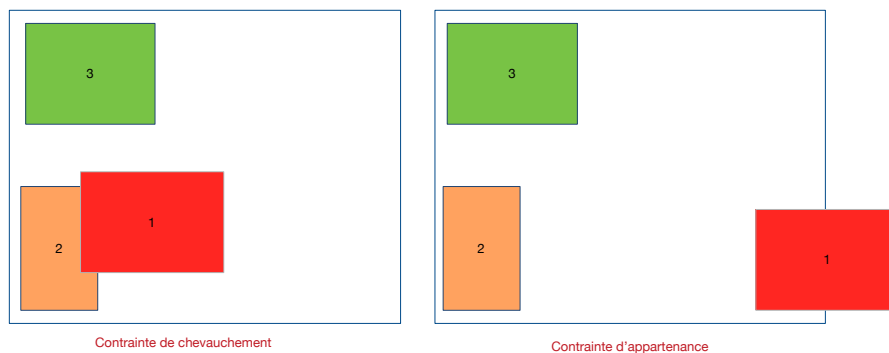


Figure 8. Exemples de contraintes géométriques

Les *contraintes fonctionnelles* sont les contraintes qui n'ont un sens que dans le contexte dans lequel elles sont appliquées, à savoir les contraintes de distances minimales et maximales entre les objets (figure de gauche ci-dessous, les contraintes de placement dans des zones en particulier...

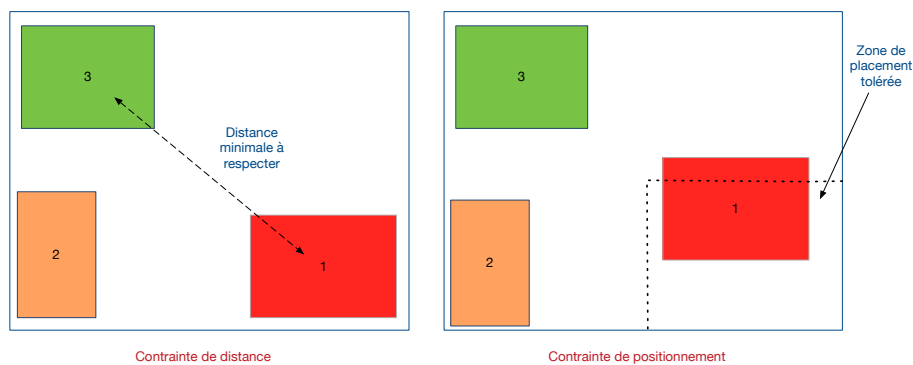


Figure 9. Exemples de contraintes fonctionnelles

Naturellement, la fonction objectif est la maximisation de la fonction d'évaluation de la qualité d'une solution. L'expression de cette fonction peut changer selon le paradigme et la méthode de résolution utilisés. En programmation mathématique par exemple, la difficulté d'exprimer certains critères de manière formelle (comme les critères de mobilité que nous verrons dans le Chapitre III lors du placement des portes) impliquera l'utilisation de paradigmes complémentaires, comme les graphes par exemple, et nécessitera une reformulation du problème pour considérer ces contraintes ainsi que la fonction objectif à optimiser. Un autre exemple de difficulté, la modélisation naturelle de certaines contraintes peut aboutir à des programmes mathématiques non-linéaires comme nous le verrons lors de la prise en compte de l'orientation dans le Chapitre III.1 et qui amène des contraintes quadratiques. Toujours dans la programmation mathématique, en particulier mono-objectif, une reformulation possible de la fonction objectif peut être la maximisation ou la minimisation d'un critère parmi

l'ensemble des critères d'évaluation, tout en ramenant l'ensemble des critères restant du problème initial au rang de contraintes avec des seuils à respecter. Par exemple, on peut chercher à maximiser la surface utile de l'aménagement et garder le coût des travaux comme une contrainte avec un seuil maximal à respecter.

Evidemment, dans certains cas, la réécriture du modèle ne garantit pas l'équivalence avec le problème initial ; dans d'autres cas, il n'est pas envisageable de se payer le luxe d'utiliser une méthode exacte pour résoudre un problème réputé NP-difficile de grande taille. Dans les deux situations, on procède à la relaxation de certaines contraintes.

5. Travaux existants

Les premières études que nous avons recensées concernant les problèmes d'agencement datent des années 70-80. Deux pistes ont été explorées [57] :

- travailler sur une grille et affecter des cellules élémentaires de la grille à telle ou telle pièce, en particulier en s'appuyant sur une approche gloutonne [77] ;
- placer des rectangles dans le plan, ou bien en étudiant parfois tous les placements relatifs, ce qui en fait une approche irréaliste quand le nombre de pièces croît [31], ou bien en s'appuyant sur des graphes [48].

Depuis, plusieurs autres approches ont été développées : des approches basées sur la satisfaction de contraintes comme dans le projet HM2PH ou dans les travaux de Medjdoub pour la conception du logiciel ARCHIPLAN [70,56], la programmation mathématique quadratique comme dans Michalek et al [59] etc. Quant à la résolution, elle a été faite avec des métaheuristiques comme l'algorithme génétique ou le recuit simulé [19,51,59,70,82], tantôt avec des heuristiques adaptées [74], tantôt avec des procédures de séparation et d'évaluation [51,55,70] etc. Tous les modèles ne sont pas compatibles avec toutes les méthodes de résolution. Par exemple, si l'on souhaite résoudre avec un algorithme génétique, il faut que le modèle permette la mise en place d'un opérateur de croisement, qui combine deux aménagements de la génération courante en deux nouveaux aménagements de la prochaine génération. C'est une opération compliquée, à tel point que l'on ne réussit pas à la réaliser sur certains modèles. D'ailleurs, beaucoup ne définissent pas cet opérateur et se contentent de l'opérateur de mutation, seul opérateur pour générer une population [51,59].

Toutes les approches manipulent des pièces de forme rectangulaire. Les couloirs sont parfois considérés comme des données du problème (on précise combien on en veut) et parfois comme une résultante d'un aménagement. Les contraintes sur les chemins, quand elles sont considérées, sont des contraintes qui permettent uniquement de vérifier qu'il existe des chemins directs pour relier les pièces sans traverser des pièces intermédiaires. Les portes et les fenêtres sont placées soit de manière arbitraire soit manuellement par l'architecte.

Parfois, l'agencement des pièces a été traité dans la littérature en s'inspirant des méthodes de conception de circuits VLSI. La ressemblance entre l'agencement d'un lieu et celle de composants électroniques est réelle. Néanmoins, le problème d'agencement reste un problème qui se divise plus naturellement en sous problèmes quasi-distincts que les problèmes VLSI et les degrés de liberté sont nettement plus nombreux en architecture qu'en VLSI [75].

Dans ce qui va suivre, nous allons parler brièvement de la conception de circuits VLSI avant d'examiner plus en détail quelques travaux sur l'agencement d'un environnement intérieur. Nous accordons une attention particulière au projet scientifique HM2PH car c'est un projet qui a inspiré nos travaux du fait qu'il est le seul projet scientifique qui traite de notre problématique, à savoir l'agencement d'un lieu de vie pour une personne handicapée.

5.1. Problème de conception de circuits VLSI

Le problème d'intégration à grande échelle (en anglais *Very Large Scale Integration*, abrégé *VLSI*) est un problème de conception de circuits qui consiste à placer les composants d'un circuit logique (diodes, transistors, résistances...) sur une puce avec, comme objectif, la minimisation de la surface utilisée.

Traditionnellement ce problème est découpé en deux étapes :

- le placement des composants : cet étape a pour objectif de placer les composants sur le support en respectant les contraintes telles que la distance des composants à l'alimentation, la logique du circuit, la connexité des composants, etc. Cette étape a été généralement traitée de manière heuristique car elle est difficile à modéliser. Citons tout de même les travaux de Lengauer et Lügering [52,86] qui ont proposé des formulations entières pour les problèmes de placement et de routage [29] ;
- le routage avec affectation des réseaux aux couches : à ce stade les composants sont déjà positionnés, le but est de placer les pistes reliant tous les composants d'un même réseau sans que deux pistes de deux réseaux différents ne se croisent. Dans ce cadre, on peut citer les travaux de Grötschel et al [34] et Jain et al [41] qui traitent ce problème comme un problème d'arbre de Steiner¹² pour minimiser la longueur des fils de cuivre utilisés pour connecter tous les réseaux. On peut évoquer aussi les travaux de Fouilloux qui a traité ce problème avec l'objectif de minimiser le nombre de vias¹³, en le ramenant à un problème de recherche d'un sous graphe k-parti induit dans un graphe [29,30].

Pour résumer :

- Il est difficile, lors de la résolution, de considérer simultanément tous les aspects de ce problème, d'où une résolution souvent en deux étapes. C'est un élément que nous avons retenu dans notre première méthode (Chapitre III) ;
- L'emplacement des entrées-sorties de chaque composant est connu, ce qui n'est pas le cas avec l'agencement d'un logement car le choix du nombre et des positions des portes dans chaque pièce fait partie de la conception de l'aménagement.

5.2. Problème d'agencement d'un environnement intérieur

5.2.1. Travaux de Schwarz et al

La méthode de Schwarz et al. [75,76] est inspirée d'une technique 2D de compression de VLSI [2,19]. Elle est plutôt orientée vers le bâtiment résidentiel mais, d'après les auteurs, elle est extensible aux hôpitaux, écoles...

Les données sont les suivantes : 1) les pièces souhaitées (noms et nombres, par exemple deux salles de bains), 2) un ensemble d'adjacences désirées (par exemple une adjacence entre une chambre

¹² L'arbre de Steiner est un arbre de poids minimal dans un graphe G tel que le sous graphe induit soit connexe et comporte tous les sommets de G.

¹³ via : un trou métallisé pour établir une liaison entre deux couches dans un circuit imprimé

donnée et une salle de bains donnée), 3) les orientations des murs pouvant comporter des portes d'entrée (i.e. le mur sud-est peut contenir une entrée), 4) les encadrements des surfaces des pièces.

La méthode pour produire l'agencement est en deux étapes :

1. les pièces du logement sont rectangulaires et décrites dans deux graphes pondérés orientés GX et GY : GX contient tous les murs verticaux, GY tous les murs horizontaux. Une arête de poids p entre deux murs indique la distance minimum entre les 2 murs qu'elle joint quand $p > 0$ et la distance maximum quand $p < 0$;
Les contraintes imposées au logement sont traduites en des associations entre des variables de décision (dont les positions et dimensions des pièces) et des groupes d'arêtes.

Exemple 1 : on veut une fenêtre de largeur L_f à plus de d des extrémités d'un mur de longueur lm : on introduit les trois variables de décision L_f , d et lm et on impose la contrainte : $L_f + 2d < lm$.

Exemple 2 : on veut que deux pièces soient adjacentes par le mur droit de la première et le mur gauche de la deuxième : on crée une arête de poids nul entre ces deux murs dans GX et des arêtes dans GY entre les murs haut/bas des deux pièces pour empêcher le recouvrement vertical.

Toutes les contraintes qui peuvent être exprimées en termes d'écart entre des murs parallèles peuvent être prises en compte.

Plusieurs contraintes (notamment pour éviter les recouvrements) sont ajoutées automatiquement ;

2. La résolution est assurée en combinant un branch-and-bound et une heuristique.

On constate qu'il n'est pas possible d'imposer des contraintes sur les itinéraires, sauf à les approcher par des DX et des DY entre les points de départ et d'arrivée, ce qui devient irréaliste dès l'instant que le chemin à une forme en U, par exemple.

Le plan suivant illustre l'exemple d'un aménagement généré à l'aide de la méthode proposée par Schwarz et al. Les entrées du problème sont:

- neuf pièces (Kitchen, Dining room, Living room, Entrance, kids' bedroom, master bedroom, Bathroom, Corridor);
- le couloir doit être adjacent aux deux salles de bains, à la salle à manger et à la cuisine. La cuisine doit être adjacente à l'entrée du logement.

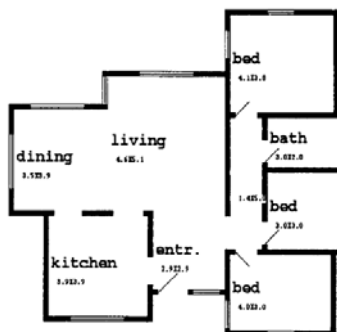


Figure 10. Exemple d'un plan généré

La méthode proposée permet de considérer l'entrée du logement comme une pièce. Le but est d'éviter qu'une porte d'entrée donne directement sur une chambre à coucher, par exemple. C'est un élément qui a été retenu dans nos travaux.

Le placement des portes est fait de manière arbitraire. Cela est dû au fait que les contraintes sur les itinéraires ne sont pas considérées. Dans notre cas, les contraintes sur les itinéraires sont indispensables du fait qu'ils permettent de prendre en compte certains besoins des personnes handicapées.

5.2.2. Travaux de Michalek et al.

Michalek et al. ont développé deux outils de génération de plans pour assister l'architecte : un outil automatique et un outil interactif. Les deux outils permettent de manipuler des pièces de forme rectangulaire et l'aménagement est fait sur un seul niveau [59].

Les entrées des deux outils sont : pour chaque pièce, la surface minimale et les dimensions minimales et maximales, et les contraintes liées à respecter.

L'outil de génération automatique des plans est un processus qui nécessite l'intervention de deux modules (Figure 11) : un module d'optimisation topologique, qui consiste à gérer des contraintes comme les contraintes d'adjacence entre les pièces et les contraintes de chemins, et un module d'optimisation géométrique qui consiste, à partir d'une topologie donnée, à chercher la meilleure position et la meilleure taille pour chaque pièce.

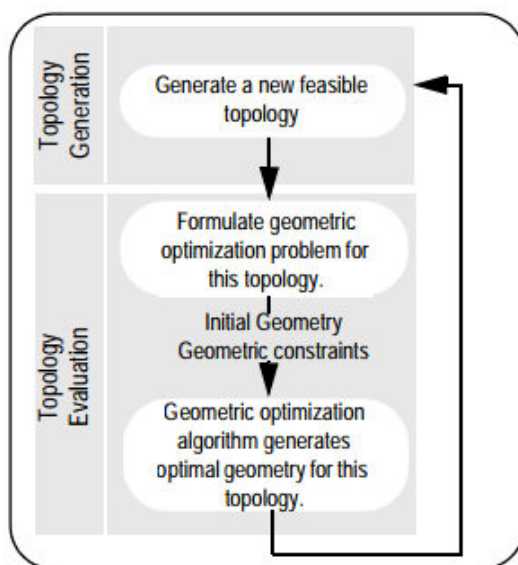


Figure 11. Processus de génération de plans automatique

Le processus utilisé dans l'outil interactif de génération de plans est très proche de celui utilisé dans la génération de plans automatique, la différence réside dans le fait que l'optimisation topologique, qui consiste à définir les placements relatifs des pièces, est gérée par l'architecte, comme le montre la figure suivante.

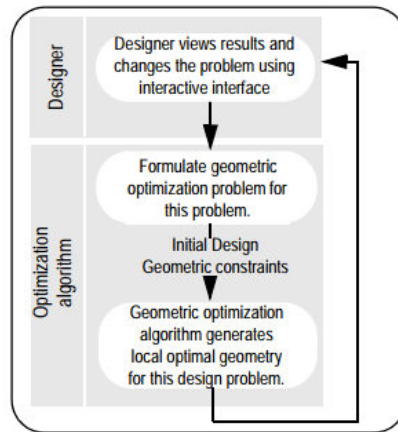


Figure 12. *Processus de génération de plans interactif*

5.2.2.1. *Module d'optimisation géométrique*

Dans ce module, Michalek et al. définissent *Unit* comme étant un rectangle, qui peut être le contour extérieur, une pièce, un couloir ou un accès (une zone pour garantir l'ouverture d'une porte). Les variables de décision sont les coordonnées du point bas gauche du rectangle et les dimensions.

5.2.2.1.1. *Les contraintes supportées*

Ce module supporte les contraintes suivantes :

- forcer à l'intérieur: pour forcer certains *Units* à se placer à l'intérieur d'autres *Units*. Par exemple, tous les *Units* doivent être à l'intérieur du contour extérieur, qui est aussi un *Unit* ;
- interdire/forcer une intersection: pour interdire ou forcer le chevauchement de deux *Units*. Forcer un chevauchement peut être utile dans le cas où un accès doit garantir l'ouverture d'une porte dans les deux sens, ce qui implique que l'*Unit* de l'accès doit être placé à cheval sur les deux *Units* qu'il faut communiquer (une chambre et un couloir par exemple) ;
- forcer contre un bord: pour forcer une pièce à se placer contre un mur du contour extérieur ;
- limites sur la largeur, la longueur, longueur/largeur et la surface des *Units* ;
- fenêtre réalisable: pour garantir que la position de la fenêtre est valide (pas de débordement) ;
- coût de construction: pour maîtriser les coûts de construction. Pour simplifier, le coût est une estimation qui est proportionnelle à la longueur des murs et des fenêtres ;
- limites du niveau d'éclairage : pour garantir qu'il y a un niveau de luminosité naturelle acceptable à l'intérieur des pièces. Le niveau de luminosité est une fonction qui prend en compte l'angle et l'altitude des pièces pour déterminer la quantité de lumière du jour qui traverse la pièce.

5.2.2.1.1. *Les fonctions objectifs utilisées*

Plusieurs fonctions objectifs ont été développées pour mesurer la qualité des aménagements générés. Une seule fonction est utilisée à la fois et toutes sont des fonctions quadratiques. Il s'agit de la :

- minimisation du coût du chauffage / Minimisation du coût de refroidissement : pour la première fonction, l'idée est de quantifier les déperditions thermiques pour estimer le coût nécessaire au maintien de la bonne température. Cette fonction dépend de la surface des fenêtres et de la largeur des murs extérieurs. La deuxième fonction est similaire à la première à un détail près, la fonction dépend en plus, de la surface des fenêtres et de la largeur des murs extérieurs, du gain en chaleur solaire à travers les fenêtres ;

- minimisation du coût de l'éclairage : l'objectif est d'encourager l'utilisation de l'éclairage naturel. Cette fonction s'inspire des fonctions précédentes et elle dépend de l'utilisation de l'occupant, qui est une donnée du problème fixée selon l'utilisateur ;
- minimisation de l'espace perdu / minimisation de la taille des couloirs et des accès: cela revient à maximiser la surface utile, c'est à dire la somme des surfaces des pièces.

5.2.2.1.2. Les méthodes de résolutions

Deux approches pour la résolution du problème ont été utilisées.

- Une première approche, basée sur une résolution locale, consiste à utiliser l'algorithme OQS (pour Optimisation Quadratique Successive). Cet algorithme est un algorithme primal-dual procédant par linéarisation des conditions d'optimalité de Karush, Kuhn et Tucker (KKT) [46]. Pour plus de détails concernant cet algorithme, nous invitons le lecteur à consulter le livre de Bierlaire (*Introduction à l'optimisation différentiable*) [7] ;
- La deuxième approche, basée sur une résolution globale, consiste à utiliser un recuit simulé (lui-même disponible en deux versions), un algorithme génétique et une méthode appelée MDDM (Maximum Distance Distribution Method) [44,45]. Pour le recuit simulé, la différence entre les deux versions réside dans la manière avec laquelle est généré un voisin. Il est soit construit aléatoirement soit en utilisant l'algorithme de OQS pour chercher à chaque itération un optimum local. Pour l'algorithme génétique, l'opérateur de mutation est défini comme étant une perturbation d'un aménagement avec une probabilité faible (changer les positions et les dimensions). L'opérateur de croisement par contre n'est pas évoqué par l'auteur. La dernière méthode, MDDM, qui est une séquence de recherches locales et globale, consiste à utiliser la recherche locale pour trouver un optimum local x^* qui permettra ensuite de définir un nouveau problème d'optimisation quadratique pour maximiser la distance entre la solution globale x et l'optimum local x^* ($Max (x - x^*)^2$).

5.2.2.2. Module d'optimisation topologique

Ce module permet de générer le meilleur ensemble de relations entre les pièces en fonction des contraintes imposées. La solution proposée par ce module est un graphe dans lequel les sommets sont les pièces et les arcs représentent les chemins directs entre chaque paire de pièces. Le graphe de la Figure 13 illustre une solution topologique.

Les pièces sont placées sur une grille. Une grille est définie par la subdivision de la surface du logement. La position des pièces est une variable du problème. La Figure 14 illustre deux interprétations géométriques d'une grille topologique.

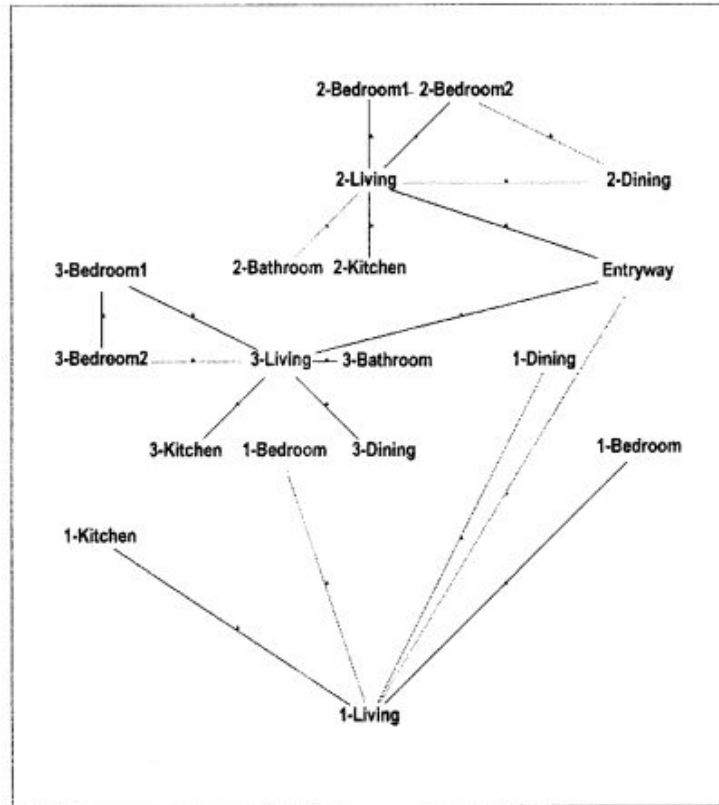


Figure 13. Exemple d'une topologie [59]

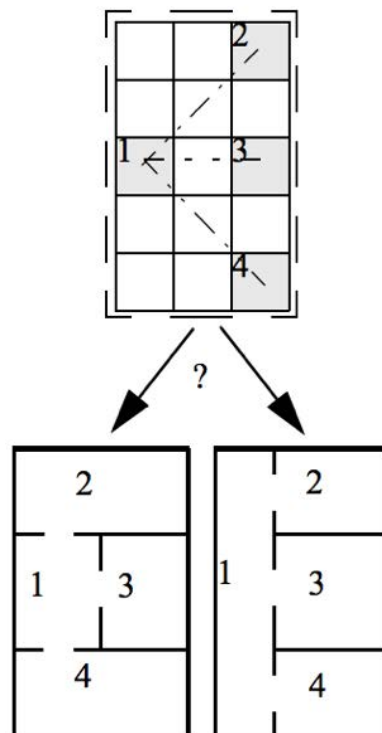


Figure 14. Deux interprétations géométriques d'une grille topologique

5.2.2.2.1. Les contraintes et les fonctions objectif supportées

Les contraintes supportées par ce module sont des contraintes de non chevauchement, de chemin et d'appartenance. Les contraintes de chemin, par exemple, permettent de garantir un chemin direct entre deux pièces sans traverser une pièce intermédiaire.

L'objectif de l'optimisation topologique est la minimisation de la valeur de la fonction objectif choisie pour l'optimisation géométrique.

5.2.2.2.2. Les méthodes de résolutions

La résolution a été faite avec un algorithme génétique. Deux fonctions de croisement ont été définies : sexué et asexué. Le croisement sexué croise deux parents. Les deux nouveaux enfants sont générés en utilisant la connexion mixte des deux parents (c'est à dire, en ne gardant que les arcs communs des deux parents). Le croisement asexué ne sélectionne qu'un parent de la population et génère un enfant soit en ajoutant ou en supprimant des arcs entre les pièces soit en échangeant la position des pièces. Pour la fonction de mutation, elle intervient après le croisement, elle se traduit soit par la modification de la position des pièces (on déplace légèrement les pièces) soit par l'inversion du sens des arcs du graphe.

5.2.2.3. Exemple

L'exemple suivant illustre l'aménagement généré, à l'aide de l'outil de génération automatique, d'une surface avec 19 pièces. Les entrées données à l'outil sont : les bornes inférieures des surfaces des *Units* et les bornes inférieures et supérieures de la largeur de chaque *Unit*. Les contraintes du problème sont : deux *Units* ne peuvent occuper le même espace ; l'entrée principale du logement doit être adjacente à un mur extérieur et doit être adjacente, aussi, à certaines pièces ; certaines pièces doivent être adjacentes à un mur extérieur ; certains itinéraires, pour aller d'une pièce à une autre, peuvent passer par une pièce intermédiaire, donnée, et toutes les pièces doivent être à l'intérieur du logement.

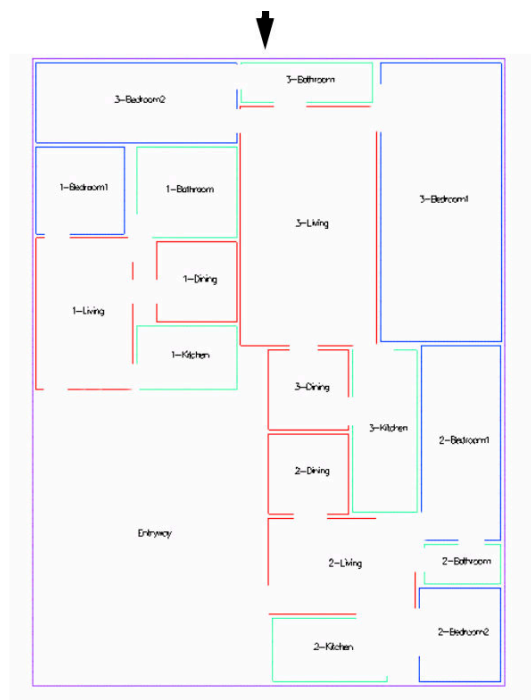


Figure 15. Exemple d'un plan généré avec l'outil automatique

Pour résumer, plusieurs méthodes de résolution ont été testées et plusieurs fonctions objectif, quadratique, ont été élaborées. L'objectif de l'élaboration de plusieurs fonctions, est de laisser le choix à l'utilisateur du critère qu'il veut optimiser. L'optimisation de plusieurs critères à la fois n'est pas considérée, par exemple, il n'est pas possible de réduire à la fois la consommation d'énergie et réduire l'espace non utile.

Les deux modules supportent des contraintes de bas niveau pour traduire toutes les contraintes fonctionnelles. Cet aspect de la modélisation des contraintes est repris et étendu dans le Chapitre IV.

Comme dans la méthode précédente, les pièces sont rectangulaires et le placement des portes n'est pas optimisé, du fait que les valeurs des itinéraires (les distances) entre les pièces ne sont pas considérées.

5.2.3. Projet HM2PH (Habitat Modulaire et Mobile pour Personnes Handicapées)

L'HM2PH est un projet scientifique réalisé au sein de l'université de Tours et qui avait pour objectif de favoriser l'autonomie des personnes handicapées en leur proposant un outil d'aide à la conception d'habitats adaptés à leurs besoins.

Ce projet a été développé dans le cadre de deux thèses de doctorat. La première thèse, réalisée par Leloup [49], a permis de développer : 1) deux bases de données pour analyser les déficiences et capacités du futur résident et proposer une liste de matériels spécifiques (aides techniques, équipements domotiques, etc.) pour compenser ses handicaps ; 2) un moteur de génération de plans adaptés à l'occupant avec une prise en compte de certaines règles architecturales, ainsi que de quelques souhaits qu'il a formulé ; 3) un module pour une visite virtuelle afin de simuler les solutions apportées. La continuité de ce projet a été assurée par les travaux de thèse de Puret [68]. Son approche est aussi composée de trois modules. Le premier module consiste à acquérir les besoins et souhaits de l'habitant. Le deuxième module est celui de la génération des plans adaptés, et pour valider les plans proposés, un module de visite virtuelle a été développé.

Dans les travaux de Leloup et Puret, deux hypothèses ont été émises. Il s'agit de :

- la forme des pièces, qui est rectangulaire pour ne pas se heurter à la modélisation des formes plus complexes ;
- les aménagements sont faits sur un étage à la fois.

5.2.4. Les travaux de Leloup [50]

Comme évoqué plus haut, les travaux de thèse de Leloup portent sur trois modules. Dans ce qui va suivre, nous détaillerons uniquement les deux premiers modules du fait qu'ils ont un lien direct avec nos travaux, à savoir le module concernant les deux bases de données et le module de génération des plans.

5.2.4.1. *Les bases de données pour une analyse des déficiences et capacités de l'habitant*

La base de données du matériel est destinée à aider les professionnels médicaux à prescrire des dispositifs d'aide technique [51]. Elle contient les références des différents matériels disponibles sur le marché pour améliorer la qualité de vie de l'habitant. Cette base de données permet de confronter les capacités et les limites de la personne en besoin de compensation avec les activités nécessitant une aide. Par exemple, on associe un tourne page à une personne qui a des difficultés à tourner les pages d'un livre ; à une personne muette, on associe un des logiciels de synthèse vocale que la base de données recense.

Cette base de données est accompagnée d'un formulaire dynamique pour sélectionner automatiquement les questions appropriées en fonction des réponses précédentes afin de dresser de manière efficace la liste des matériels dont l'habitant a besoin.

La base de données des pièces permet la collecte d'informations utiles à l'aménagement d'un lieu de vie pour une personne handicapée. Dans cette base de données, on retrouve les normes d'accessibilité comme par exemple, pour une personne en fauteuil roulant, les dimensions minimales des pièces et celles de la zone permettant de faire un demi-tour etc.

Une fois que le profil de l'habitant est dressé, c'est le deuxième module (MoGAP : Moteur de Génération Automatique de Plans) qui intervient pour générer plusieurs plans automatiquement afin de sélectionner le meilleur aménagement possible. Ce qui suit résume le fonctionnement de ce module.

5.2.4.2. *Moteur de génération Automatique de plans (MoGAP)*

Ce module permet la génération automatique de plans adaptés en prenant en compte les souhaits et les besoins de l'occupant, formulés à l'aide de la base de données des pièces, ainsi que les contraintes liées aux normes de constructions [49,50].

Ce module est divisé en deux parties : 1) la première partie consiste à créer *les règles de préférences*. Les règles de préférences sont les besoins et les souhaits de l'occupant (que nous détaillerons plus loin) ; 2) l'utilisation d'une des deux méthodes de résolution développées pour générer un plan adapté.

À noter que Leloup envisageait, en utilisant la base de données du matériel, de réadapter ce module pour réaménager aussi l'intérieur de chaque pièce afin de placer les appareils préconisés par l'analyse des déficiences et des capacités de l'habitant faite lors de la première étape.

Règles de préférences

Les deux méthodes utilisées pour générer les plans sont basées sur *les règles de préférences*. Par exemple, l'occupant souhaite ou a besoin que la cuisine soit la plus proche possible du salon ou que la chambre soit accessible par un couloir, etc. Ainsi, l'utilisateur peut créer ses règles de préférences avec les quatre opérateurs de bas niveau suivants :

« ne doit pas être agrandi(e) » et « doit être agrandie » spécifient si les dimensions de la pièce peuvent et/ou doivent être modifiées ;

« doit être contre un mur » et « doit être dans un coin » permettent de spécifier les préférences sur le placement des pièces dans l'espace de vie ;

« doit être accessible par le couloir » permet à l'utilisateur d'imposer des contraintes sur la circulation dans son espace de vie, cela empêche qu'une pièce d'être *enfermée* ou d'être accessible uniquement via une autre pièce ;

« est le plus loin possible de » et « est le plus près possible de » donnent la possibilité de créer ou d'empêcher des groupements entre les pièces.

Méthodes de génération automatiquement de plans

Deux méthodes de résolution du problème ont été utilisées pour générer plusieurs esquisses respectant au mieux les contraintes architecturales et les règles de préférences de l'habitant. La

première approche consiste à utiliser un algorithme génétique ; quant à la deuxième approche, l'idée est de mettre en œuvre une procédure de séparation et d'évaluation (PSE).

Avant de rentrer dans les détails de chaque méthode, précisons que chaque pièce est représentée par les coordonnées de son centre, par ses dimensions ainsi que son orientation (elle est tournée de 90°).

a. Méthode basée sur l'algorithme génétique

La population initiale est générée à l'aide d'une heuristique qui va tirer de manière aléatoire les coordonnées et l'orientation de chaque pièce. Si une paire de pièces se chevauchent, la pièce de rang inférieur - la pièce la moins importante - est retirée et une deuxième tentative est faite pour tenter de placer la pièce retirée dans l'espace libre restant. Si, après plusieurs tentatives, la pièce ne peut être placée, le plan est abandonné.

La fonction F d'évaluation permet d'évaluer la qualité d'un agencement selon les préférences de l'utilisateur :

$$F = 1 - \sum_{i=1}^N f_i * P_i$$

avec f_i le fitness d'une pièce, P_i son poids et N le nombre total de pièces.

Chaque pièce a un poids relatif à son importance qui est choisie arbitrairement par l'utilisateur.

P_i est calculé ainsi :

$$P_i = \frac{m - r_i}{m}$$

m étant le nombre de pièces soumises à une ou plusieurs conditions et r_i le rang de la pièce i (un rang de 0 pour la pièce la plus importante et de $n - 1$ pour la moins importante).

Soit q le nombre de contraintes auxquelles la pièce i est soumise, alors :

$$f_i = \frac{\sum_{j=1}^q \frac{e_{ij}}{E_{jMAX}}}{q}$$

avec e_{ij} l'évaluation de la contrainte j et E_{jMAX} la plus mauvaise valeur possible pour cette contrainte.

Plus e_{ij} est grande et plus f_i est grande, et donc plus la pièce i respecte les contraintes qu'elles lui sont associées.

La génération de la population suivante est basée uniquement sur l'opérateur de mutation. L'opérateur de croisement, comme il a été imaginé par Leloup consiste à inverser les gènes de deux individus, est jugé inutile du fait qu'il ne génère quasiment que des plans invalides.

L'utilisateur commence par choisir le ou les plans qui conviennent le plus parmi les meilleurs plans proposés (au sens de la fonction f_i). Une fois que le ou les plans sont sélectionnés, l'utilisateur choisit une mutation parmi trois :

- la mutation par amélioration, consiste à placer chaque pièce, traitée par ordre de rang décroissant, au mieux afin de satisfaire le maximum de préférences formulées ;
- la mutation totalement aléatoire, consiste à déplacer aléatoirement les pièces où à changer leurs orientations et dimensions ;
- la mutation aléatoire en suivant les règles de préférences, consiste à générer aléatoirement les distances de déplacements en abscisse et en ordonnée. La direction par contre est prédéterminée, elle est basée sur les règles de préférences.

À la *fin de l'algorithme* l'utilisateur peut demander à générer une autre population ou choisir un plan final parmi les plans générés.

b. Méthode basée sur la procédure de séparation et d'évaluation

Cette procédure, qui semble donner de meilleures solutions que la première, exploite un arbre de décision, qui est construit en ajoutant une pièce à chaque niveau. Si la pièce ne peut être placée alors la branche est coupée et le nœud parent est détruit. Afin de limiter le nombre de plans, un écart minimum de 10 cm a été fixé (un pas de déplacement) entre deux placements d'une même pièce sur deux plans (deux branches). L'évaluation des nœuds est faite de la même manière que pour l'évaluation dans l'algorithme génétique.

5.2.4.3. Validation du plan final

Une fois que les plans sont générés, une fonction pour l'évaluation de la circulation est appliquée (l'algorithme (A*)). L'objectif est de chercher un chemin entre chaque paire de pièces. La recherche des chemins se fait d'une porte à l'autre. Les portes sont placées automatiquement sur le mur le plus proche de la porte d'entrée. Le plan est jugé non conforme quand la largeur d'au moins un couloir est non réglementaire. Dans ce cas, une procédure de correction est lancée. Cette procédure consiste soit à supprimer le couloir, s'il n'est pas loin d'un mur du contour extérieur du logement, soit à rapprocher, agrandir ou éloigner des pièces pour rendre la circulation possible.

5.2.5. Quelques remarques

D'après Leloup, les résultats des deux méthodes de résolution sont satisfaisants. La deuxième méthode, basée sur la séparation et l'évaluation, a été implémentée pour minimiser le nombre d'interventions de l'utilisateur dans le processus de génération des plans. Les interventions nécessaires semblent cependant encore nombreuses.

La vérification de la possibilité de se rendre dans toutes les pièces est faite une fois que le plan est choisi et non pendant la phase de génération des plans. Si la vérification invalide le plan, la démarche peut être très coûteuse puisqu'il faut relancer le processus pour générer d'autres plans.

Le calcul des itinéraires se fait d'une porte à une autre, uniquement dans le but de vérifier la possibilité de circuler dans le logement. La maximisation ou la minimisation des itinéraires (que ce soit pour satisfaire une préférence du futur occupant ou une contrainte forte imposée par un handicap) ne sont pas considérées.

5.2.6. Les travaux de Puret [68]

La continuité de ce projet a été assurée par les travaux de thèse de Puret [68]. Son approche dans l'ensemble est différente de celle développée par Leloup, notamment son choix d'utiliser des

techniques de satisfaction de contraintes pour la génération des plans. Elle est composée de quatre modules, qui sont : la modélisation des contraintes liées aux besoins et souhaits de l'habitant, la génération des plans, la classification de plans et la visite virtuelle pour simuler la solution finale dans des conditions très proches de celles du futur occupant.

Dans la continuité de cette section, nous détaillerons les deux premiers modules, à savoir la modélisation des contraintes et la génération des plans du fait qu'ils ont un lien direct avec nos travaux.

5.2.6.1. Modélisation des besoins et souhaits de l'habitant

La position des pièces est exprimée par les coordonnées du point supérieur gauche et inférieur droit. Les dimensions sont exprimées par la longueur et la largeur de la pièce. Une variable, valant 0° ou 90°, définit l'orientation de la pièce. Un couloir est considéré comme une pièce.

Pour une modélisation informatique de tous les besoins et souhaits de l'occupant, Puret utilise la liste des opérateurs de bas niveau suivants :

Opérateur	Signification
PositionX(P,X) et PositionY(P,Y)	Position occupée par une pièce P (le point d'application est le coin supérieur gauche). Les variables X, Y sont représentées dans des intervalles fermés et continus de nombres entiers.
DimensionL(P,L) et DimensionW(P,W)	Dimension d'une pièce P (L est la dimension horizontale et W la dimension verticale). Les variables L, W représentent des intervalles fermés et continus de nombres entiers.
Surface(P,S)	La surface est un opérateur portant sur les variables L et W d'une même pièce P. La variable représente un intervalle fermé et continu de nombres entiers.
Angle(P,A)	Orientation d'une pièce P dans le plan selon l'axe vertical. Deux orientations sont autorisées : 0° (pièce « horizontale ») et 90° (pièce « verticale »). Si aucune orientation n'est préférée, A = {0°,90°}.
Adjacence(P1,P2,D,C,O)	L'adjacence indique que la pièce P1 doit avoir un mur en contact avec la pièce P2. Les paramètres sont : <ol style="list-style-type: none"> 1) D (distance entre les pièces, intervalle continu et borné d'entiers) ; 2) C (longueur du mur en contact, intervalle continu et borné d'entiers) ; 3) O (orientation du mur de contact sur P1, liste de constantes dans : {Nord, Sud, Est, Ouest}).

AdjContour(P,O)	Cette adjacence est une version simplifiée de la précédente, elle impose un contact direct entre une pièce P et le mur extérieur de l'habitat (le contour). Le seul paramètre est l'orientation (liste de constantes dans : {Nord, Sud, Est, Ouest}) correspondant au mur extérieur à mettre en contact avec la pièce.
NonRecouvrement(P1,P2)	Deux pièces P1 et P2 ne doivent pas se recouvrir mutuellement (elles ne peuvent prendre un même emplacement) Cette contrainte est placée automatiquement par le logiciel de génération de plans.

Tableau 1 Liste des opérateurs de bas niveau

Le tableau suivant donne les modélisations adoptées pour trois contraintes de haut niveau.

Contrainte de haut niveau	Modélisation adoptée
Avoir la porte d'entrée de la maison dans la pièce P	AdjContour (P, {Nord, Sud, Est, Ouest})
Les pièces {P1,P2...} qui ont un besoin en eau / assainissement sont sur un même mur extérieur	(AdjContour(P1, {Nord}) ET AdjContour(P2,{Nord}) ET...) OU (AdjContour(P1, {Sud}) ET AdjContour(P2,{Sud}) ET...) OU...
Il est possible d'aller de la pièce P1 à la pièce P3 en passant par P2	Adjacence (P1, P2, [0], [1,...], {Nord, Sud, Est, Ouest}) ET Adjacence (P2, P3, [0], [1,...], {Nord, Sud, Est, Ouest})

Tableau 2 Exemples de modélisations des contraintes de haut niveau

5.2.7. Module de génération de plans

Pour une génération automatique des plans, l'auteur s'est inspiré de l'approche utilisée par Medjdoub et Yannou dans le développement du logiciel ARCHIPLAN¹⁴ [55,56]. Cette approche est divisée en deux phases, il s'agit de :

¹⁴ Un Outil développé dans la thèse de Medjdoub (1996) et amélioré par la suite avec Yannou (2000). Il a pour but d'assister l'architecte dans son processus de création des plans en lui proposant plusieurs plans qui prennent en compte le cahier des charges du projet

- la phase de recherche topologique, qui consiste à définir la position relative des pièces les unes par rapport aux autres (recherche exhaustive de toutes les solutions) ;
- la phase d'optimisation géométrique, qui optimise la position et les dimensions des pièces afin de satisfaire les critères de l'habitant.

5.2.7.1. Recherche topologique

La première phase de la génération des plans consiste à générer un maximum de solutions topologiques. Ces solutions doivent satisfaire les contraintes de positionnement relatif des pièces imposées par les contraintes de type *Adjacence* et *AdjContour*. À ce stade, les dimensions et la position exactes des pièces ne sont pas figées, elles sont définies dans un intervalle min/max.

La construction des solutions se fait avec des techniques de satisfaction de contraintes. L'algorithme sélectionne une pièce et évalue les contraintes qui la concernent. Si ses contraintes sont non satisfiables, l'algorithme en cherche d'autres parmi celles décrites dans la spécification de l'habitat. Sinon, il sélectionne une autre pièce et reprend la même procédure jusqu'à l'instanciation de toutes les pièces (ou échec de la construction de la solution).

La démarche utilisée pour construire une solution est fortement influencée par l'ordre dans lequel les pièces sont instanciées et l'ordre des contraintes à évaluer.

En ce qui concerne l'ordre d'instanciation des pièces, Puret utilise une heuristique nommée « *dynamic space ordering* », issue des travaux de Medjdoub et Yannou [56], et qui permet d'attribuer à une pièce une note. Cette note correspond au degré de contraintes d'adjacence auxquelles une pièce doit se soumettre. Plus la pièce est contrainte plus elle est prioritaire.

L'ordre de vérification des contraintes est fait selon la nature des contraintes. Les contraintes unaires sont vérifiées en premier (celles ne portant que sur une seule pièce) pour réduire les domaines des variables. Les contraintes d'orientation sont parmi les premières contraintes à être vérifiées. Ce sont ces contraintes qui donnent aux pièces leur longueur et leur largeur pour ensuite être positionnées correctement.

5.2.7.2. Optimisation géométrique

Ce module a deux objectifs : fournir un plan valide pour la disposition des pièces (à partir des plans topologiques générés avec l'outil précédent) et optimiser la position et les dimensions afin de satisfaire un critère (ex. coût de construction, facilité de circulation, etc.).

La technique utilisée pour optimiser les plans est basée sur la procédure de séparation et évaluation.

La fonction d'évaluation choisie par Puret consiste à optimiser le coût de construction. En reprenant les calculs menés par Medjdoub et Yannou [56], le coût de construction de l'habitat est considéré comme proportionnel aux dimensions des murs intérieurs. Le coût estimé est donc la somme des longueurs des murs intérieurs multipliée par le prix unitaire d'un mètre de cloison.

La procédure de séparation et d'évaluation est aussi influencée par l'ordre des variables et l'instanciation des valeurs. Une pièce est décrite à l'aide de quatre variables qui sont les coordonnées du point haut à gauche et du point bas droit de la pièce. Puret choisit d'instancier les pièces les plus contraintes en premier pour réduire le temps d'exploration de l'arbre. Pour classer les pièces de la plus contrainte à la moins contrainte, l'auteur comptabilise les contraintes d'adjacence aux murs extérieurs auxquelles, il ajoute les contraintes d'adjacence entre chaque pièce et le reste des pièces déjà instanciées. La pièce qui obtient le plus gros score sera instanciée en premier. Pour chaque variable à instancier, l'auteur choisit une valeur parmi les trois choix suivants : valeurs minimales en premier

(min) ; valeurs du milieu de l'intervalle en premier (middle) ; valeurs des bornes de l'intervalle en premier (limites).

Du fait que les couloirs sont traités comme des pièces et que leur nombre est figé, les solutions générées peuvent contenir des espaces libres (des trous). La suppression de ces espaces est faite en augmentant la taille des pièces adjacentes, de manière progressive, tout en respectant les limites Min/Max des variables.

5.2.8. Quelques remarques

L'approche développée par Puret est plus autonome que celle développée par Leloup. L'utilisateur peut n'intervenir qu'à la fin du processus, pour corriger les imperfections, et non pendant le processus de génération des plans, pour ajuster les paramètres au fur et à mesure des itérations. Les solutions générées semblent être satisfaisantes.

Le nombre de couloirs ainsi que leurs intervalles de dimensions sont gérés comme ceux des autres pièces et sont donc des données du problème. L'utilisateur de la méthode doit donc avoir une idée préalable du nombre de couloirs nécessaires.

La porte d'entrée du logement peut donner directement sur une chambre, il n'y a donc aucune garantie qu'elle donne sur un séjour par exemple.

6. Conclusion

Dans ce chapitre, nous avons fait un tour d'horizon des différentes problématiques du handicap et de l'aménagement. L'aménagement d'un lieu de vie est un problème complexe qui donne lieu à plusieurs sous problèmes d'optimisation combinatoire. Tous les travaux étudiés décomposent le problème maître en plusieurs sous problèmes.

Nous remarquons que tous les travaux que nous avons étudiés émettent l'hypothèse que les pièces sont rectangulaires. Certains considèrent les couloirs comme des pièces, d'autres comme une résultante d'une disposition des pièces de vie. Parmi les travaux qui permettent de prendre en compte les contraintes de chemins, aucun ne considère la valeur des itinéraires pour aller d'une pièce à une autre, tous vérifient uniquement l'existence d'un chemin. Ce qui va les inciter à poser les portes soit manuellement soit de manière arbitraire alors que les positions des portes ont une réelle influence sur la longueur des itinéraires dans le logement.

En conclusion de cette étude bibliographique, voici deux listes : 1) celle des points que nous estimons particulièrement intéressants dans ces études et que nous réutilisons dans nos travaux et 2) celle des aspects que nous nous efforçons d'améliorer ou pour lesquels nous apportons une toute première solution.

Propositions réutilisées :

- contraindre une pièce à contenir telle zone intéressante et non mobile d'un logement (évacuation d'eaux, conduit de cheminée...);
- produire un aménagement en plusieurs étapes car la prise en compte de contraintes très nombreuses et très variées en une seule fois est trop compliquée. Nous procédons ainsi dans la synthèse d'un aménagement ;
- nous avons particulièrement apprécié la possibilité, offerte dans le projet HM2PH, de traduire des spécifications de haut niveau, proches d'un langage utilisateur, en des concepts qui peuvent être gérés par un logiciel. Nous reprenons ce principe en l'élargissant à une plus

grande panoplie de souhaits et de besoins, pour couvrir davantage de situations handicapantes (vieillesse, grossesse, surdité, handicap visuel...);

- maîtriser les coûts de constructions en les estimant proportionnels à la longueur totale des murs intérieurs ;
- contraindre les dimensions des pièces en permettant à l'utilisateur de fournir des intervalles de validité.

Points améliorés ou nouveaux apports :

- faire en sorte que les contraintes sur les chemins aient un impact sur la position des pièces et des portes ;
- étudier l'opportunité d'autoriser des pièces de formes non rectangulaires ;
- agrandir la panoplie des contraintes qui permettent au futur occupant de spécifier son futur logement ;
- proposer des méthodes exactes et approchées, pour la production d'un logement satisfaisant un ensemble de contraintes variées et nombreuses ;
- ne pas obliger l'utilisateur à fixer le nombre de couloirs et à choisir quels couloirs desservent telles ou telles pièces ;
- éviter d'impliquer l'utilisateur pendant l'élaboration (phase de calcul) du plan du logement.

Chapitre II --- Modélisation et Algorithmes géométriques

1 Introduction

Cette étude traite d'un problème pluridisciplinaire. Cela s'est traduit, entre autres, par la nécessité de développer un certain nombre d'algorithmes géométriques, dont les principaux sont présentés dans ce qui suit. Les calculs des intersections (en particulier entre deux segments) présentent peu d'intérêt donc ils ne sont pas présentés.

La structure de données d'un logement utilise des murs sans épaisseurs, appelés murs *filaires* dans la suite. À chaque mur filaire sont associées diverses informations, parmi lesquelles :

- l'origine et l'extrémité du mur filaire. Ces deux points définissent un sens de parcours du mur que l'on qualifie d'*intrinsèque*. Une question récurrente est de savoir si un mur est utilisé par une pièce dans son sens intrinsèque ou pas. Dans la Figure 16 les quatre murs de la pièce sont numérotés de 0 à 3, ils forment un rectangle et, d'après ce que l'on voit, ce rectangle tourne dans le sens horaire ; les murs 0 et 3 sont parcourus par la pièce dans leur sens intrinsèque mais pas les murs 1 et 2.

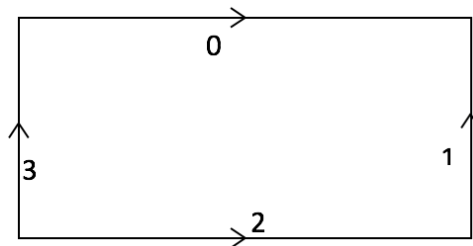


Figure 16. Sens de parcours des murs

Il est aisé de constater qu'un mur est utilisé dans son sens intrinsèque par une pièce : il faut que l'extrémité du mur soit l'un des deux points du prochain mur de la pièce : dans notre exemple, l'extrémité du mur 0 est l'extrémité du mur 1 et l'extrémité du mur 3 est l'origine du mur 0.

Une autre question récurrente est de savoir, pour un mur donné d'une pièce donnée, si la pièce est à gauche ou à droite du mur lorsqu'on le parcourt dans son sens intrinsèque. Considérons par exemple le mur 2 dans les quatre cas de la Figure 17 : on constate que, bien que les dispositions relatives des murs 1 et 3 soient identiques dans les deux premiers cas et dans les deux derniers cas, la pièce est respectivement à droite, à gauche, à droite et à gauche du mur 2. La réponse à cette question est moins simple à formuler et fait l'objet de l'une des sections de ce chapitre.

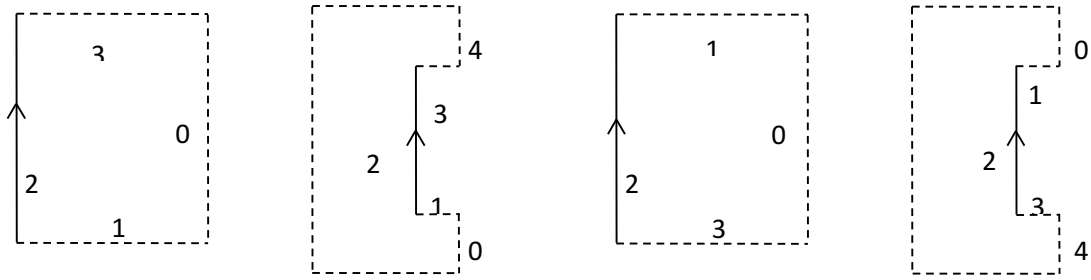


Figure 17. Cas de figures de la position relative d'un mur par rapport à une pièce

- l'épaisseur du mur,
- la manière dont la matière du mur est placée par rapport au mur filaire lorsqu'on le parcourt dans son sens intrinsèque : du côté gauche, du côté droit ou de part et d'autre (Figure 18).

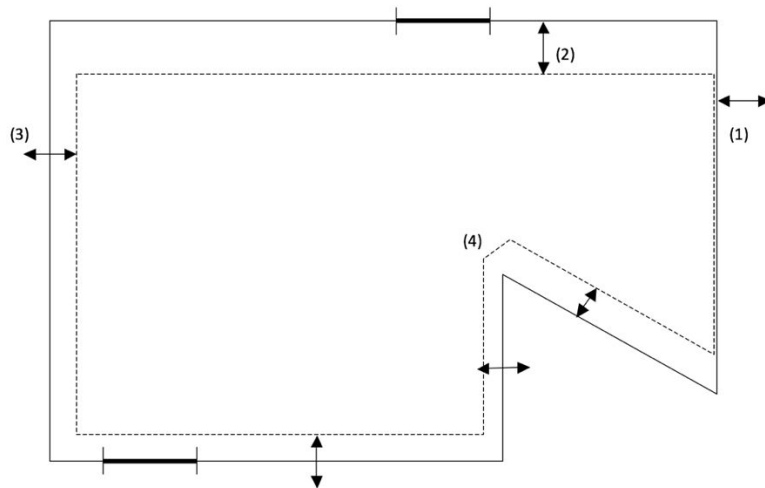


Figure 18. En trait plein, les murs filaires ; les doubles flèches donnent la position de la matière sur les murs filaires ; en pointillés, le pourtour de la pièce

La plupart des algorithmes présentés dans ce chapitre sont tirés du livre de Gardan [32].

Les algorithmes ne sont pas présentés par difficulté croissante mais en fonction de leurs dépendances. Quand un algorithme est utile à un autre, le premier est présenté d'abord.

Avertissement : les flèches au-dessus des vecteurs ne sont tracées que lorsqu'il pourrait y avoir équivoque.

2 Produit scalaire et produit vectoriel

Une des applications du produit scalaire est l'évaluation du fait qu'un vecteur va globalement dans le même sens qu'un autre.

Appelons Ref le vecteur de référence et U un autre vecteur, supposons que les deux sont normés. Alors, le produit scalaire $Ref \cdot U$ varie, en fonction de l'orientation et du sens de U , de 1 (les deux vecteurs sont colinéaires et de même sens) à -1 (Figure 19).

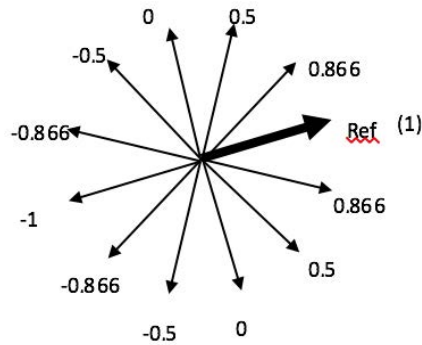


Figure 19. Variation du produit scalaire de deux vecteurs

Rappelons aussi que le produit vectoriel de deux vecteurs U et V fournit, à la condition que les deux vecteurs soient non colinéaires, un troisième vecteur W orthogonal aux deux autres et formant avec eux un système direct.

3 Calcul du sens de rotation d'un polygone / La normale orientée à un polygone

Définition

Nous verrons que les pièces sont, entre autres, décrites par un polygone. Une des requêtes de base est le calcul du sens de rotation de ce polygone. Etant donné qu'il est dans un plan parallèle à (xOy) , cela consiste à dire s'il tourne dans le sens trigonométrique ou dans le sens horaire. Nous calculons plutôt, ce qui revient au même, une normale orientée ; cette normale vaut \vec{K} si le polygone tourne dans le sens trigonométrique, $-\vec{K}$ sinon.

Motivation

Le calcul du sens de rotation d'un polygone est utilisé à plusieurs occasions :

- Comme nous le verrons dans le Chapitre IV, nous serons amenés à modifier la forme des pièces d'un aménagement (par exemple déplacer un sommet). Une telle modification peut engendrer un changement du sens de rotation d'une pièce d'où la nécessité de recalculer le sens de rotation de son polygone à chaque modification de sa forme ;
- Nous avons souvent besoin de savoir de quel côté d'un mur se trouve la pièce : soit trois points consécutifs A, B, C du pourtour. La pièce est à gauche du segment $[B, C]$ si la normale orientée est $+\vec{K}$, à droite sinon ;
- Pour calculer l'angle entre deux murs qui se rejoignent en un sommet, il faut connaître le sens de rotation du polygone.
- Pour savoir si un sommet est convexe ou concave.

Algorithme

Soit S_k un sommet d'un polygone. On obtient une normale (non orientée) au polygone en faisant le produit vectoriel du vecteur arrivant et du vecteur repartant :

$$\vec{n} = \overrightarrow{S_{k-1}S_k} \times \overrightarrow{S_kS_{k+1}}$$

En effet, à la condition que deux vecteurs soient non colinéaires, leur produit vectoriel est non nul et orthogonal à ces deux vecteurs. Il faut donc choisir k tel que S_{k-1} , S_k et S_{k+1} soient non alignés. La Figure 20 montre les normales obtenues sur cinq des six sommets d'un hexagone (en S_1 , le calcul est inutile car ce sommet est aligné avec S_0 et S_2).

Pour que le vecteur obtenu soit une normale *orientée*, S_k doit être un sommet convexe, c'est à dire un sommet où la matière du polygone fait un angle de moins de 180° . Dans la figure, S_0, S_2, S_4, S_5 conviennent. Pour trouver un tel sommet, aucun calcul complexe n'est nécessaire : il suffit de choisir un sommet extrême dans une direction quelconque, par exemple celui de plus grande abscisse (soit S_0, S_1 ou S_2). S'ils sont plusieurs, on choisit celui de moindre ordonnée pour assurer la condition de non alignement.

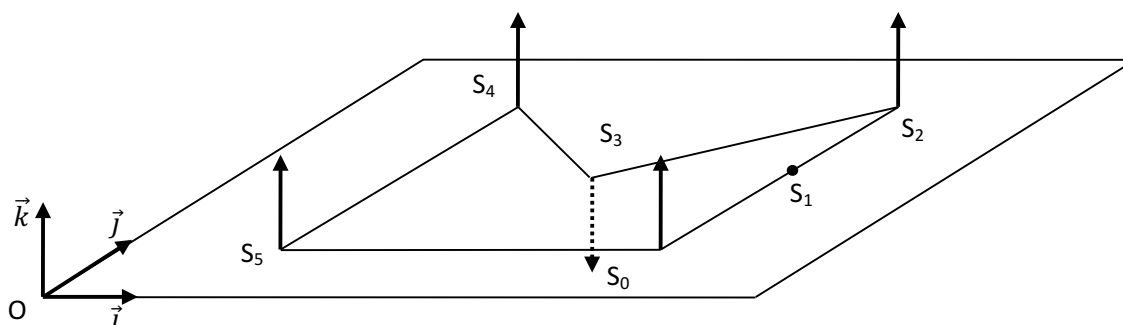


Figure 20. Exemples de vecteurs de produits vectoriels

4 Un sommet d'un polygone est-il convexe ou concave ?

Sous l'hypothèse que la normale orientée \vec{n} du polygone est connue, on vérifie qu'un sommet S_k est convexe en vérifiant que le produit vectoriel du vecteur arrivant et du vecteur repartant a même sens que la normale orientée, c'est à dire que :

$$(\overrightarrow{S_{k-1}S_k} \times \overrightarrow{S_kS_{k+1}}) \cdot \vec{n} > 0$$

A titre d'illustration, revenons à la figure précédente : le polygone a une orientation trigonométrique donc sa normale orientée va vers le haut. On constate qu'en tous les sommets convexes, le produit vectoriel est effectivement orienté vers le haut alors qu'il est orienté vers le bas en S_3 .

5 Normale extérieure / intérieure à un segment d'un polygone

Sous l'hypothèse que la normale orientée \vec{n} d'un polygone est connue, on obtient une normale extérieure au côté $[S_k, S_{k+1}]$ avec le produit vectoriel :

$$\overrightarrow{S_kS_{k+1}} \times \vec{n}$$

La normale intérieure est l'opposée de la normale extérieure, elle est égale à $-\overrightarrow{S_kS_{k+1}} \times \vec{n}$ c'est à dire $\vec{n} \times \overrightarrow{S_kS_{k+1}}$.

6 Calcul du pourtour d'une pièce

Définition

Le pourtour d'une pièce est le polygone qui entoure la pièce, au pied des murs (Figure 21). Certains des segments du pourtour sont superposés à des murs filaires, dans le cas où la matière d'un mur est du côté opposé au côté de la pièce (cas des murs 1 et 2). Les autres segments du pourtour sont strictement parallèles aux murs filaires qui les engendrent (murs 0 et 3).

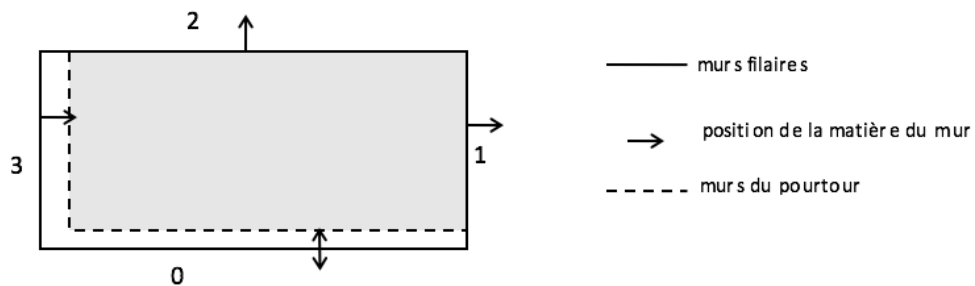


Figure 21. Pourtour d'une pièce

Motivation

On se rend bien compte que la seule donnée des murs filaires ne suffit pas à évaluer précisément la surface d'une pièce. Par exemple, dans le cas où, pour tous les murs filaires d'une pièce, la matière du mur est du côté de la pièce, il y a une différence significative entre la surface du polygone formé par les murs filaires et celle du polygone qui correspond au pourtour réel de la pièce. Pour les pièces de la Figure 22, dont les murs sont épais de 20 cm, la différence est respectivement de 3,04 m² et 4,64 m².

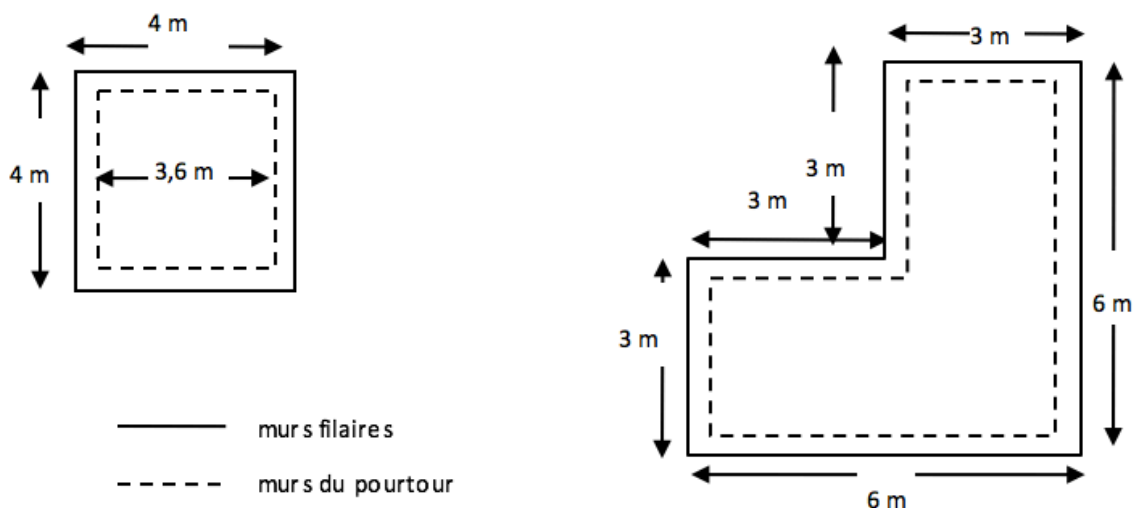


Figure 22. La différence des dimensions entre le polygone d'une pièce et le polygone de son pourtour

Il faut donc calculer le pourtour *réel* de chaque pièce.

Algorithme

L'algorithme procède comme suit :

- on part de la séquence de murs filaires qui entourent la pièce (Figure 23.a)
- un premier polygone, Pol1, est superposé aux murs filaires de la pièce. Les sommets de ce polygone sont ceux des murs filaires à deux différences près :
- les sommets ne sont plus dupliqués (dans la structure de données initiale, les deux murs qui partagent un sommet dupliquent ce sommet)
- quand plusieurs sommets sont alignés, seuls les deux sommets extrêmes sont conservés (dans la Figure 23.b, on constate qu'aucun sommet de Pol1 ne correspond au raccord entre les murs filaires 2 et 3) ;
- le sens de rotation de Pol1 est calculé ;
- chaque sommet convexe ou peu concave (l'angle de matière est inférieur à 225°) de Pol1 engendre un *unique sommet* dans le pourtour (sommets Pol1(0), Pol1(1), Pol1(2), Pol1(4), Pol1(5) dans la Figure 23.b) ; chaque sommet très concave de Pol1 (sommet Pol1(3)) engendre *deux sommets* dans le pourtour ; cette précaution est nécessaire étant donné que les formes des pièces sont parfois très inhabituelles, certains sommets étant effectivement très concaves. On "casse" alors le coin, afin d'éviter de longues pointes qui s'étendent très loin dans la pièce.

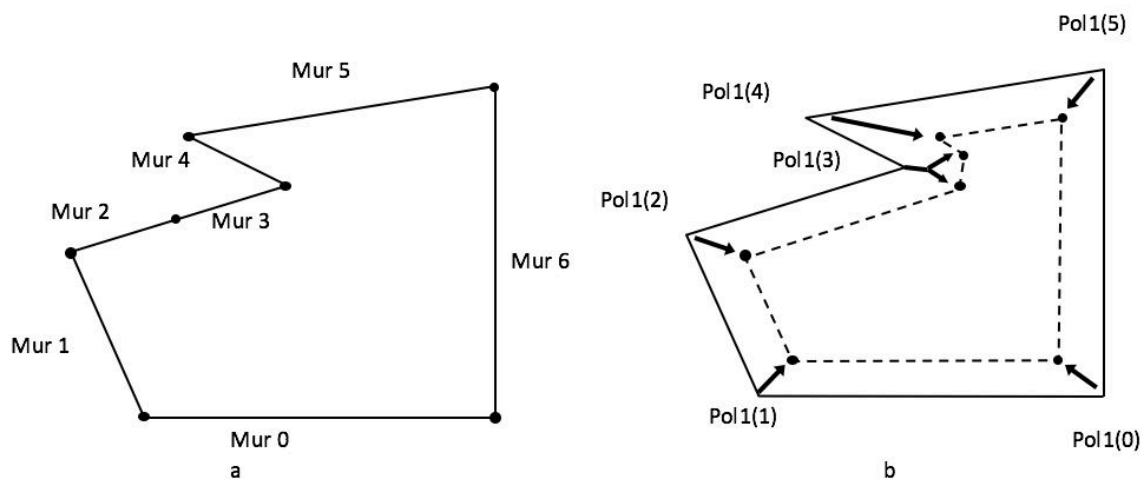


Figure 23. Calcul du pourtour

Le reste de cette section est consacré à la présentation détaillée de cette dernière étape.

Calcul du point en vis à vis d'un sommet convexe ou peu concave

Afin de ne pas aller trop loin dans les aspects mathématiques, nous avons placé en annexe 2 l'opération géométrique de très bas niveau qui calcule le point d'intersection I_k entre les parallèles Par1 et Par2 à deux droites $D1$ et $D2$ sécantes en S_k (Figure 24) :

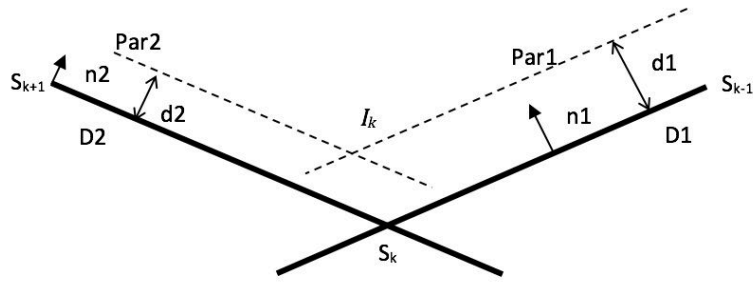


Figure 24. Calcul du point en vis à vis d'un sommet convexe ou peu concave

C'est cette opération qui est utilisée pour calculer un point du pourtour en vis à vis d'un sommet *convexe ou peu concave* de Pol1 : S_k est un sommet du polygone Pol1. $D1$ et $D2$ sont respectivement la droite qui porte le segment $[S_{k-1}, S_k]$ de Pol1 et la droite qui porte le segment $[S_k, S_{k+1}]$ de Pol1. Les autres données de l'opération sont :

- les vecteurs $n1$ et $n2$, respectivement orthogonaux à $D1$ et $D2$. Le premier indique de quel côté de $D1$ doit se trouver la parallèle $P1$; dans notre contexte, $n1$ doit donc être orienté vers l'intérieur de la pièce ; $n2$ fait de même pour $D2$.

Or, si $[S_{k+1}, S_k]$ est un segment d'un polygone de normale *orientée* \vec{n} alors, on obtient un vecteur orthogonal à ce segment et orienté vers l'intérieur du polygone avec :

$$\text{On a donc } n1 = \vec{n} \times \overrightarrow{S_{k-1}S_k}$$

$$\text{et } n2 = \vec{n} \times \overrightarrow{S_kS_{k+1}}$$

- $d1$ et $d2$ indiquent à quelles distances se trouvent les parallèles. Chacune de ces deux valeurs dépend de trois facteurs : 1) la manière dont la matière du mur est placée relativement au mur (à gauche, à droite, à cheval), 2) le fait que le mur filaire soit utilisé dans son sens intrinsèque et 3) le fait que Pol1 tourne dans le sens trigonométrique ou horaire, soit $3 \times 2 \times 2 = 12$ combinaisons possibles et, à chaque fois, si on appelle E l'épaisseur du mur, les trois valeurs possibles pour une distance sont 0 , $E/2$ et E . Trois des configurations sont illustrées par les murs 0, 1 et 2 de la Figure 25, dans laquelle est dessinée une pièce orientée dans le sens horaire :
 - o le mur 0 est à contre sens et sa matière est à sa gauche (quand il est parcouru dans son sens intrinsèque). La parallèle doit se trouver à une distance égale à l'épaisseur du mur ;
 - o le mur 2 n'est pas à contre sens et sa matière est à sa gauche (quand il est parcouru dans son sens intrinsèque). La parallèle doit se trouver à une distance égale à 0 ;
 - o la matière du mur 1 est équitablement répartie de part et d'autre. Peu important le sens du mur et le sens de rotation de la pièce, la parallèle doit se trouver à une distance égale à la moitié de l'épaisseur du mur.

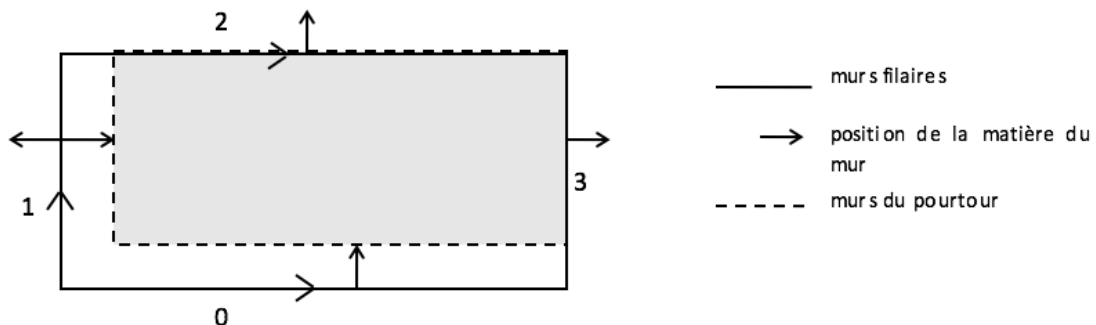


Figure 25. Les trois valeurs possibles pour la distance entre le mur filaire et le mur du pourtour

Les autres configurations sont semblables et ne sont pas représentées.

Calcul des deux points en vis à vis d'un sommet très concave

Les données sont les mêmes que dans le premier cas :

- S_k est le sommet très concave ;
- $D1, n1, d1$ sont les données du mur arrivant sur S_k ;
- $D2, n2, d2$ sont les données du mur repartant de S_k .

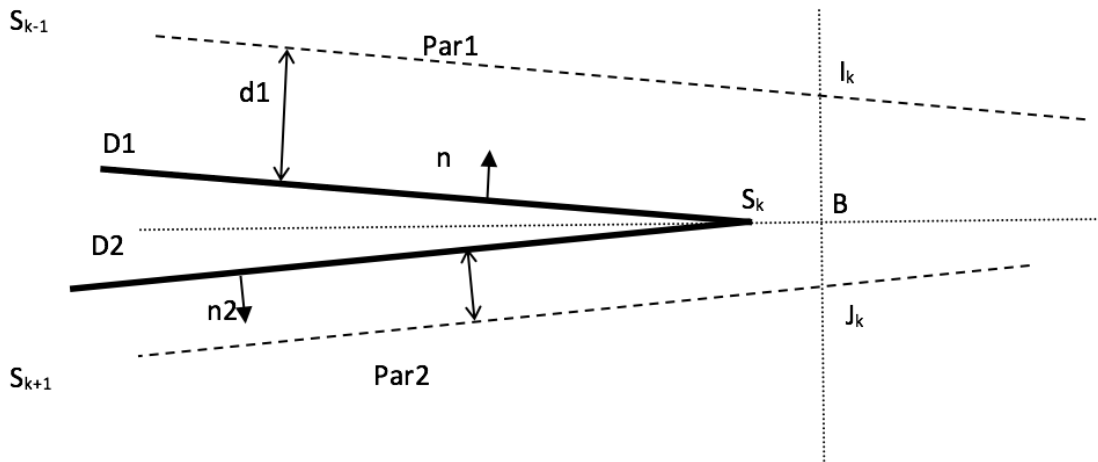


Figure 26. Calcul des deux points en vis à vis d'un sommet très concave

Si $d1$ et $d2$ sont nulles, il est inutile de créer deux points en vis-à-vis de S_k : un vis-à-vis suffit et c'est S_k lui-même. Nous supposons donc dans la suite que l'une des distances est non nulle et supposons que $d2$ est la plus petite des deux.

Les deux vis-à-vis I_k et J_k sont calculés en 2 étapes :

- B est le point de la bissectrice des droites $D1, D2$ situé à $d2$ de S_k ;
- I_k et J_k sont les intersections de $Par1$ et $Par2$ avec la perpendiculaire à la bissectrice en B .

7 Une pièce est-elle orientée vers une direction donnée ?

Définition

Cette opération doit par exemple permettre de vérifier si une pièce est orientée au sud. Il ne s'agit donc pas de l'opération de même nom utilisé dans le Chapitre III visant à vérifier si un mur d'une pièce rectangulaire est parallèle à O_x ou à O_y .

Les données sont la pièce, le vecteur Nord et un angle indiquant la direction souhaitée, avec la convention que 0° représente l'est et 90° le nord. La Figure 27 montre par exemple quelle est la direction souhaitée si l'angle est de 0° .

À noter que si la pièce n'a aucun mur qui donne sur l'extérieur, elle sera considérée automatiquement comme une pièce qui n'est orienté vers aucune direction.

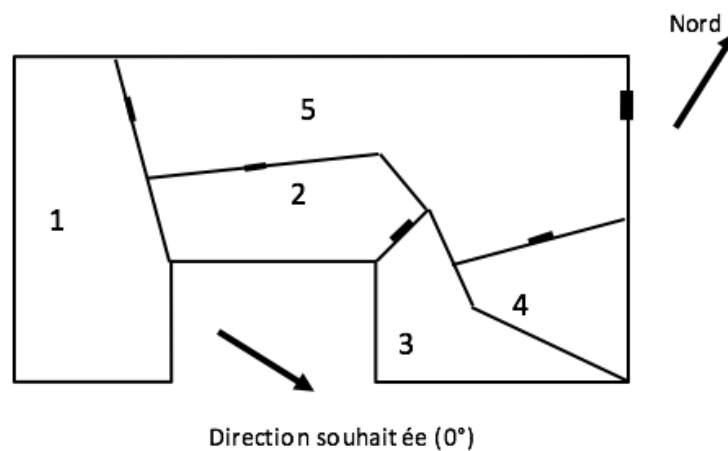


Figure 27. Exemple d'une direction souhaitée si l'angle est à 0°

Motivation

Cet opérateur est utile à plusieurs reprises. Avant toute chose, il permet de vérifier si les souhaits du futur occupant concernant l'orientation des pièces sont satisfaits (on veut le soleil du matin dans la cuisine, ce qui se traduit par une orientation de la cuisine à l'est - sauf si le bâtiment est sur un pôle ...). L'opérateur permet aussi d'agir sur la consommation énergétique de l'habitation : selon l'orientation et la fréquentation d'une pièce, le chauffage ou la climatisation seront plus ou moins utilisés ; il est donc utile de permettre ce type de spécification à l'utilisateur du logiciel.

Algorithme

Les informations disponibles dans la structure de données ne permettent de répondre à la question que de manière incomplète : on peut vérifier si la pièce est adjacente à un mur *extérieur* qui fait globalement face à la direction souhaitée mais, pour que la vérification soit parfaite, il faudrait aussi savoir si le mur est mitoyen (auquel cas on ne peut pas y mettre une fenêtre), s'il y a un bâtiment en vis-à-vis, pondérer le résultat par la longueur du mur convenablement orienté (un mètre de mur orienté au sud n'est pas suffisant pour affirmer qu'une pièce est orientée au sud) etc ... Nous avons estimé qu'il était inutile d'aller aussi loin car cela alourdisait considérablement la construction des jeux de données alors que la faisabilité est déjà établie.

Pour une pièce et un angle donnés, la vérification est réalisée comme suit :

- on calcule le vecteur normé « Direction », qui correspond à l'angle fourni (en tenant compte du décalage introduit par le fait que le nord n'est pas forcément le vecteur $J(0, 1, 0)$)
- on parcourt les murs de la pièce ;
- quand un mur se superpose à un mur extérieur (il y a une potentielle orientation de la pièce vers la direction souhaitée), on calcule une normale normée au mur, orientée vers l'extérieur de la pièce ;
- on calcule le produit scalaire de cette normale avec le vecteur direction et on conserve le plus grand nombre positif ainsi obtenu (0 à défaut).

On obtient ainsi une quantité qui varie de 0 à 1, 1 signifiant qu'un mur extérieur de la pièce fait parfaitement face à la direction demandée.

8 Une pièce ne donne pas vers une direction donnée ?

Définition

Il s'agit de l'opérateur symétrique du précédent : on souhaite par exemple vérifier qu'une pièce ne donne pas vers l'est, donc si aucun mur de la pièce n'est à la fois extérieur et face à la direction interdite.

Motivation

Les motivations sont identiques à celles de l'opérateur précédent : répondre aux demandes du futur occupant et permettre aussi la gestion énergétique du logement.

Algorithme

Il est symétrique à l'algorithme précédent. Si la pièce ne donne pas du tout vers la direction à éviter la note est alors égale à 1 sinon si, elle donne totalement ou partiellement vers la direction non souhaitée la note est alors égale à $1 - \text{la valeur de l'évaluation si le souhait était plutôt d'orienter la pièce vers la direction à éviter}$.

9 Calculer un point à l'intérieur d'un polygone

Définition

Cet algorithme calcule un point dans la région du plan incluse dans un polygone. Quelle que soit la forme du polygone, le point n'est pas sur le polygone mais bien à l'intérieur. La Figure 28 montre un polygone ayant la forme d'un U et quelques points qui conviendraient.

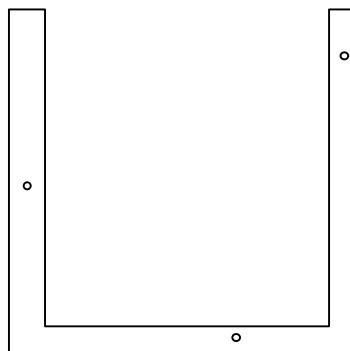


Figure 28. Quelques points à l'intérieur d'un polygone

Motivation

Cet algorithme est utilisé à deux reprises :

- pour savoir où afficher une information concernant une pièce, par exemple son nom ;
- dans le test d'inclusion d'un polygone dans un autre polygone (voir plus bas).

Algorithme

L'idée est de partir d'une arête du polygone et de s'en éloigner vers l'intérieur du polygone perpendiculairement à l'arête ; l'amplitude du déplacement doit être limitée, de manière à garantir de ne pas franchir une autre arête du polygone.

Le point de départ *Dep* est obtenu avec le milieu d'une quelconque arête.

La direction selon laquelle on s'éloigne est fournie par l'opposé du vecteur normal extérieur à l'arête (voir plus haut).

Ce point et ce vecteur définissent une droite. On calcule les intersections de cette droite avec le polygone et on les ordonne selon le sens défini par le vecteur.

Une de ces intersections est le point de départ et on obtient un point situé, de manière certaine, dans le polygone en calculant le milieu du segment allant de *Dep* à l'intersection qui le suit.

10 Vérifier si un point est dans la région du plan entourée par un polygone

Définition

Cet opérateur vérifie si un point *P* est sur ou dans un polygone. Dans la figure ci-dessous, dont le polygone a été rendu déraisonnablement compliqué pour être une pièce de forme tolérable, les points intérieurs sont matérialisés par des carrés et les points extérieurs par des cercles. Le choix de ce polygone est motivé par la volonté d'illustrer le fait que le test est moins simple que ce que l'on pourrait penser après l'observation d'un cas simpliste.

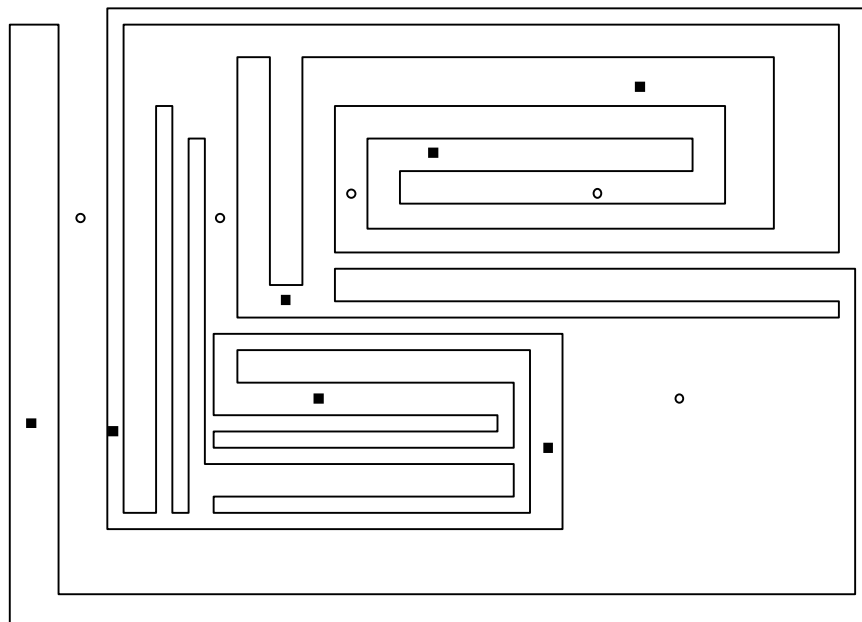


Figure 29. Exemple de plusieurs points à l'intérieur d'un polygone

Motivation

Le test d'appartenance d'un point à un polygone est utilisé à deux occasions :

- il permet une réalisation imparfaite du test d'inclusion d'un polygone dans un autre: on vérifie pour cela que chaque sommet du premier est à l'intérieur du deuxième. Il est aisé de trouver des contre-exemples à cette méthode mais, quand les formes des polygones sont simples, les résultats sont rarement faux. Comme le but premier de cette thèse n'était pas de faire de l'algorithmique géométrique, nous nous sommes contentés de cette approximation ;
- il permet une réalisation exacte du test d'inclusion d'un polygone dans un autre quand on sait que les deux polygones n'ont pas d'intersections franches mais sont éventuellement adjacents, comme cela arrive souvent dans le cas d'un ensemble de pièces.
 - o Exemple : dans la figure ci-dessous, il n'y a qu'une seule relation d'inclusion entre deux pièces : 2 est incluse dans 1.

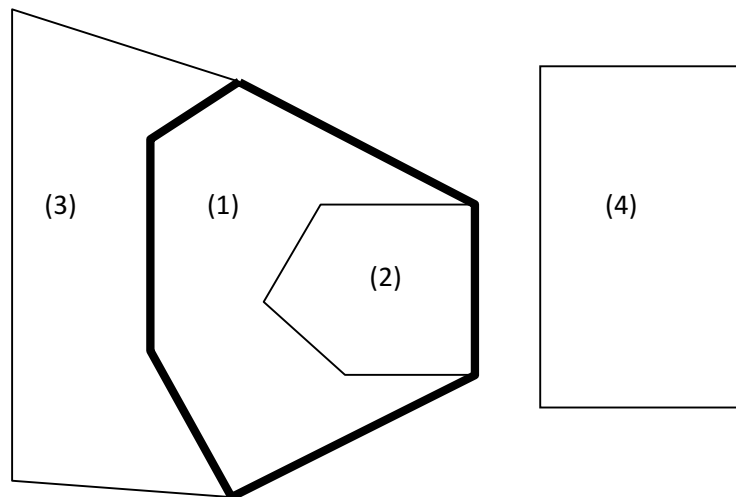


Figure 30. Exemples de Relations d'inclusions sans intersections franches

L'algorithme du test d'inclusion d'un polygone dans un autre polygone est détaillé plus loin dans ce chapitre.

Algorithme [62]

Pour vérifier l'appartenance d'un point à un polygone, une méthode, dite des *angles capables* [65], a le double mérite d'être très facile à implanter et insensible aux imprécisions inévitablement engendrées par le codage des réels. Elle ne peut être utilisée que dans le cas où la ligne fermée à laquelle on teste l'appartenance est effectivement un polygone, peu importe que ce dernier soit convexe ou concave.

Une autre méthode, dite de la *demi-droite* [71], est applicable à toute forme de ligne fermée mais, bien qu'un peu plus rapide, elle est d'une mise en œuvre sensiblement plus compliquée. Compte tenu des courts délais dont nous disposons, nous avons écarté cette méthode.

Un angle capable est défini par un point A et un segment $[P1, P2]$. Il s'agit de l'angle *orienté* formé par les segments $[A, P1]$ et $[A, P2]$ (Figure 31).

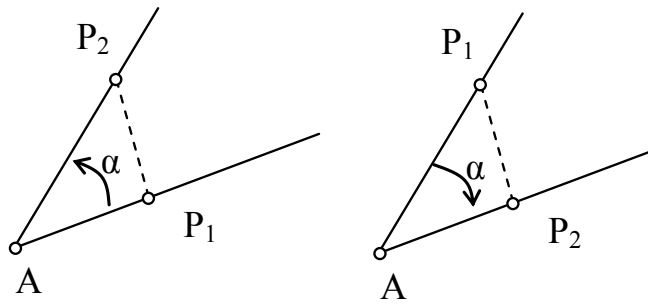


Figure 31. Exemple d'angle capable

Il s'agit donc d'un angle susceptible de varier de $-\pi$ (exclu) à π (inclus). Il peut être calculé au prix d'un seul appel à une fonction trigonométrique, ce qui rend son obtention moins coûteuse que ce qu'un examen inattentif pourrait laisser penser. Les détails calculatoires sont présentés dans l'annexe 3. Précisons simplement que :

- si A est égal à P_1 ou P_2 , l'angle capable ne peut pas être calculé ;
- si A est strictement inclus dans le segment $[P_1, P_2]$, l'angle capable est π ;
- si A est sur la droite (P_1, P_2) et en dehors de $[P_1, P_2]$, l'angle capable est nul.

Ceci étant posé, on vérifie l'appartenance d'un point A à un polygone en deux étapes :

- on vérifie d'abord si A est égal à l'un des sommets S_k du polygone, auquel cas A est *sur* le polygone ;
- si ce n'est pas le cas, on calcule les angles capables α_k formés par A et chaque segment $[S_k, S_{k+1}]$ du polygone. Dans la figure suivante, on voit deux points à tester (cercles grisés) par rapport au même polygone ; chaque angle capable est matérialisé par un arc de cercle orienté par une flèche.

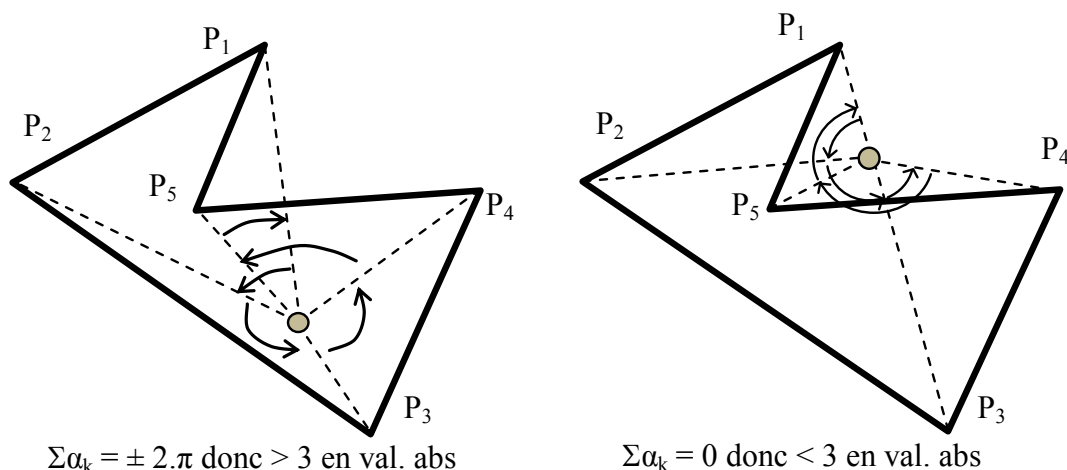


Figure 32. Vérification de l'inclusion d'un point dans un polygone à l'aide de la méthode dite des angles capables

Si l'un de ces angles est égal à π , A est *sur* le polygone et on arrête tout ;
 sinon, on calcule la somme des angles capables, qui vaut soit 0, soit $\pm 2. \pi$. Dans le premier cas, le point est *extérieur*, dans le deuxième, il est *intérieur*.

C'est en cela que la méthode n'est absolument pas sensible aux erreurs causées par le codage des réels : la somme ne pouvant prendre que deux valeurs (en valeur absolue), il suffit de vérifier si sa valeur absolue est supérieure à 3 (auquel cas elle vaut exactement $\pm 2. \pi$ et le point est intérieur) ou pas (auquel cas elle vaut 0 et A est extérieur).

11 Aire d'un polygone

Définition

Il s'agit de l'aire de la zone du plan incluse dans le polygone.

Motivation

Il s'agit d'une mesure utile au moment de vérifier si l'aire d'une pièce vérifie telle ou telle contrainte de majoration, minoration ou d'encadrement et au moment de vérifier si la forme d'une pièce est irréaliste.

Algorithme [62]

L'algorithme consiste à évaluer la formule suivante, sous la convention que (x_i, y_i) soient les coordonnées du $i^{\text{ème}}$ sommet et que le $(N + 1)^{\text{ème}}$ sommet du polygone soit le premier. Il ne s'agit aucunement d'une formule approchée ou réservée aux polygones convexes :

$$Aire = \frac{1}{2} \left| \sum_1^N (x_{i+1} - x_i)(y_{i+1} - y_i) \right|$$

Notons (cela sera utile plus tard) que le calcul de l'aire d'un polygone se fait donc en un temps linéaire en le nombre N de sommets.

12 Angle d'un sommet d'un polygone

Définition

Il s'agit de l'angle formé par le segment arrivant et par le segment repartant d'un sommet. On le mesure à l'intérieur du polygone (Figure 33). Le sommet est donc convexe si cet angle est inférieur à 180° , concave sinon. La valeur obtenue varie dans $]0, 2. \pi[$ (la valeur 0 témoigne d'un polygone invalide et ne devrait jamais apparaître).

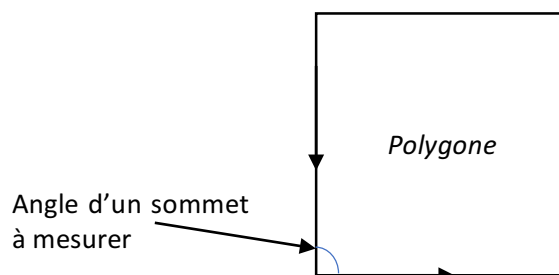


Figure 33. Exemple d'un angle d'un sommet à mesurer

Motivation

Comme on le verra plus tard, les pièces subissent des déformations très variées pendant le mécanisme d'optimisation, ce qui rend indispensable de calculer les angles en tous leurs sommets, pour détecter les valeurs incompatibles avec une construction réelle.

Algorithme

Les raisonnements sont semblables à ceux qui sont menés pour le calcul d'un angle capable (calcul du sinus et cosinus, et un appel à la fonction arc tangente).

13 Calcul de l'offset à un polygone

Définition

L'offset est une courbe située à distance constante d'une autre courbe. Dans notre étude, la courbe de référence est un polygone (le pourtour de la pièce) et l'offset est toujours situé du côté *intérieur* (figure suivante). L'offset est globalement un polygone, sauf au niveau des sommets concaves où il est arrondi (sommets 4).

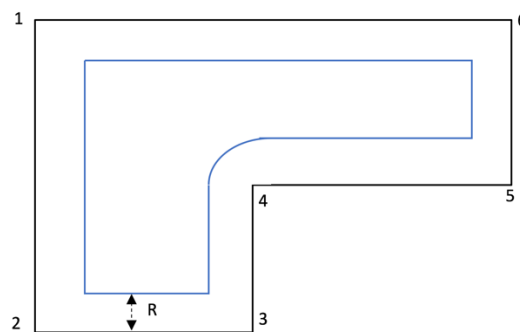


Figure 34. Offset à un polygone

Motivation

L'offset à la distance R est l'ensemble des positions les plus proches du polygone pour le centre d'un cercle de rayon R qui se déplacerait dans le polygone. Concrètement, l'occupant du logement étant approché par un cercle de rayon R , l'offset à la distance R représente le périmètre de la partie de la pièce que le centre du cercle peut atteindre.

C'est sur cet offset que sont choisis les points qui marqueront le début ou la fin d'un chemin entre deux pièces (voir Chapitre IV.2.3.2.2 Points de départ d'une pièce). On garantit ainsi deux choses : aux positions initiale et finale, la personne n'est pas en collision avec un mur de la pièce et la longueur du chemin n'est pas sous-estimée.

Algorithme

Le principe de construction de l'offset est la suivante : chaque sommet du polygone de référence engendre un sommet dans l'offset ; ce dernier est l'intersection des parallèles intérieures aux deux segments qui partagent ce sommet.

Plusieurs considérations compliquent théoriquement l'opération :

- comme on l'a vu plus haut, l'offset n'est pas constitué que de segments de droite (Figure 34) : il y a des arcs de cercle en face des sommets concaves du polygone de référence, ce qui devrait contraindre à un traitement différent des sommets concaves. Nous avons pourtant décidé de traiter les sommets concaves comme les sommets convexes. En effet, nous avons vu plus haut (voir Chapitre II.6 - l'algorithme de calcul d'un pourtour -) que lorsqu'un sommet du pourtour aurait dû être fortement concave, il est biseauté et remplacé par deux autres sommets. Illustrons ceci avec la figure ci-dessous : les murs *filaires* sont en trait épais et le pourtour de la pièce est en trait fin ; dans la zone de la flèche, on voit que le pourtour est biseauté. Les deux sommets du biseau sont encore concaves mais ils le sont peu.

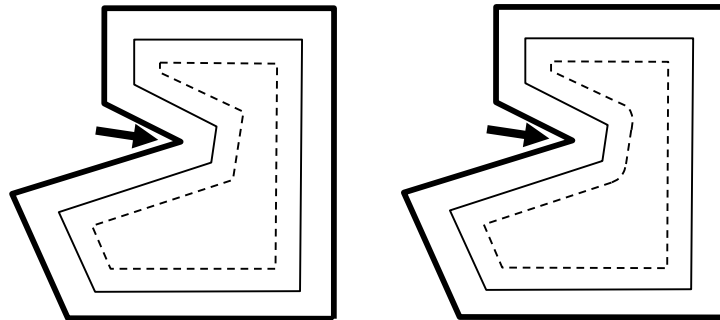


Figure 35. Comparaison entre l'offset approché (à gauche, en pointillés) et l'offset théorique (à droite, en pointillés)

En pointillés, on voit, à gauche, l'offset obtenu si on applique aux deux sommets concaves la même procédure qu'aux sommets convexes. On constate que cet offset est très semblable à l'offset théorique, qui comporterait des arcs de cercle en face des sommets concaves. Cette faible perte de précision est d'autant plus tolérable quand on connaît l'usage qui est fait des offsets calculés (voir par exemple : Calcul du plus court chemin entre deux pièces dans Chapitre IV.3.1.5) au paragraphe sur le calcul de la distance entre deux pièces).

- l'offset peut être croisé, ce qui n'est évidemment pas tolérable dans notre contexte (figure ci-dessous).

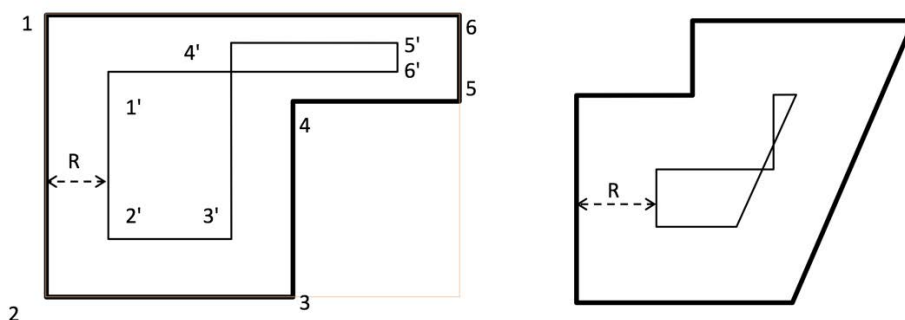


Figure 36. Deux cas de croisement d'un offset à une distance R

Les croisements sont détectés en deux temps. Un filtre est d'abord appliqué : à chaque fois qu'un sommet O_k est ajouté à l'offset, on vérifie que le segment $[O_{k-1}, O_k]$ a le même sens que le segment analogue sur le polygone de référence. Par exemple, dans la figure ci-dessus (cas de gauche), le segment des sommets 5 à 6 monte alors que le segment $[5', 6']$ descend.

Ceci traduit nécessairement la présence d'un croisement et est détecté très rapidement (le produit scalaire des vecteurs directeurs des deux segments est négatif). Si le filtre ne détecte rien d'anormal (c'est le cas avec le croisement de droite dans la figure de droite, ci-dessus), on applique un traitement plus coûteux, consistant à calculer les intersections du segment $[O_{k-1}, O_k]$ avec les segments précédents de l'offset. Les seules intersections trouvées doivent être des points (si une intersection est un segment, des arêtes de l'offset sont superposées) et ces points doivent être O_{k-1} (puisque $[O_{k-1}, O_k]$ partage le point O_{k-1} avec $[O_{k-2}, O_{k-1}]$) et, si $[O_{k-1}, O_k]$ est le dernier segment de l'offset, O_1 (puisque le dernier segment de l'offset partage le point O_1 avec le premier).

Quand un croisement est détecté, nous n'essayons pas d'appliquer un traitement visant à supprimer les boucles de l'offset, ce qu'il faudrait faire si nous voulions un opérateur d'offset valide d'un point de vue géométrique. En effet, un croisement traduit le fait que le polygone de référence (la pièce) est localement d'une dimension insuffisante pour permettre le passage de l'occupant et le polygone de référence (c'est-à-dire le pourtour de la pièce) est alors simplement invalidé.

14 Distance d'un point à un segment

Définition

La distance d'un point P à un segment $[A, B]$ est le minimum des distances de P aux points du segment.

Motivation

Un point n'est une position valide pour l'occupant du logement (c'est-à-dire une position valide pour le centre du cercle de rayon R qui approche l'occupant) qu'à la condition que la distance de ce point à tous les murs de la pièce soit supérieure ou égale à R .

Algorithme

Trois cas sont possibles : le projeté orthogonal de P sur la droite (A, B) est :

- avant A , ce que l'on détecte en vérifiant que le produit scalaire de AP et AB est négatif ;
- après B , ce que l'on détecte en vérifiant que le produit scalaire de BP et AB est positif ;
- entre A et B

La distance de P à $[A, B]$ est respectivement la distance de P à A , la distance de P à B ou la distance de P à son projeté sur (A, B) .

15 Inclusion d'un polygone dans un autre

Définition

Un polygone est inclus dans un autre quand tous les points du premier (intérieurs et périphériques) sont inclus dans l'autre. Notons que le fait qu'un polygone $Pol1$ soit inclus dans un polygone $Pol2$ n'interdit pas qu'ils aient des intersections et que ces dernières peuvent être des points ou des segments (Figure 37).

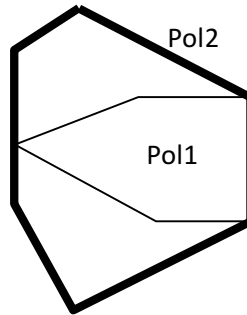


Figure 37. Exemple d'un polygone inclus dans un autre

Motivation

La position d'une pièce est parfois contrainte par la nécessité, pour cette pièce, de contenir une zone d'intérêt comme un conduit de cheminée ou des arrivées/évacuations d'eau (Chapitre IV.3.2.1.7 - Zone d'intérêt -). Vérifier qu'une telle contrainte est satisfaite équivaut à vérifier que le polygone qui limite la zone d'intérêt est inclus dans le pourtour de la pièce. Pour vérifier ce type d'inclusion, nous appliquons un algorithme approché, qui teste l'appartenance de chaque sommet du polygone d'intérêt au pourtour de la pièce. Il est aisé de trouver des contre-exemples à cette méthode mais, quand les formes des polygones sont simples, les résultats sont rarement faux. Comme le but premier de cette thèse n'était pas de faire de l'algorithmique géométrique, nous nous sommes contentés de cette approximation et ne revenons pas sur ce cas dans ce qui suit.

Deux autres situations, relativement rares, exigent aussi de vérifier l'appartenance d'un polygone dans un autre ou, pour être plus précis, la non inclusion d'un polygone dans un autre :

- la première situation est illustrée ci-dessous (Figure 38) :

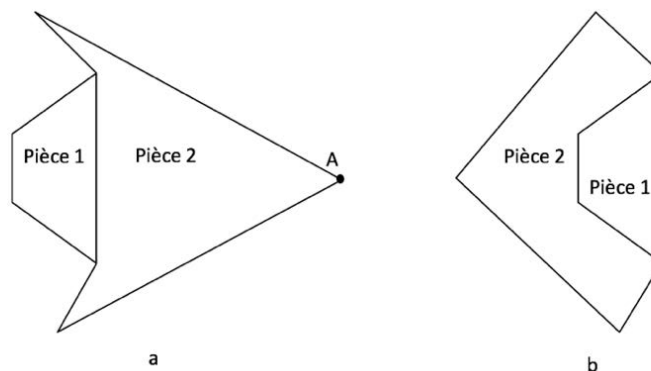


Figure 38. Cas particulier qui exige la vérification de l'appartenance d'une pièce dans une autre

Il arrive, durant l'amélioration d'un aménagement, que l'on déplace un sommet d'une pièce. Dans l'hypothèse où l'on déplacerait le sommet A vers la gauche de la pièce 1 (Figure 38.a), on obtiendrait une nouvelle disposition dans laquelle la pièce 1 serait maintenant incluse dans la pièce 2 (Figure 38.b), ce qui est interdit dans notre mise en œuvre. Il importe donc de détecter cette inclusion et d'invalider la modification ;

- symétriquement, Il arrive aussi que l'on échange deux pièces. Si la pièce 1 ci-dessous est échangée avec un couloir, on retrouve deux couloirs adjacents et, dans cette situation, nous avons convenu de supprimer les murs partagés par les couloirs, ce qui mène à l'agencement de droite. Ce cas de figure est illustré dans l'exemple suivant :

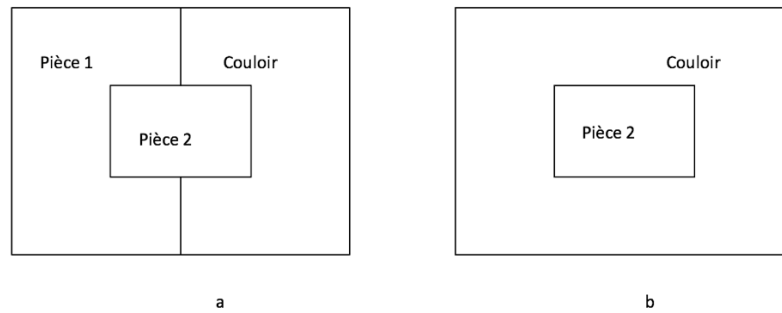


Figure 39. Une permutation qui provoque l'inclusion d'une pièce dans un couloir

A nouveau, cela mène à l'inclusion d'une pièce dans une autre, chose que nous voulons détecter et interdire.

Ces deux dernières situations ont en commun que les pièces à tester sont sans intersections franches, tout au plus peuvent-elles avoir des contacts ponctuels ou rectilignes.

Algorithme [66]

Quand deux Pol1 et Pol2 n'ont pas d'intersection franche, trois cas sont possibles : ils sont disjoints (Figure 40.b), Pol1 est inclus dans Pol2 (Figure 40.c) ou Pol2 est inclus dans Pol1 (Figure 40.a).

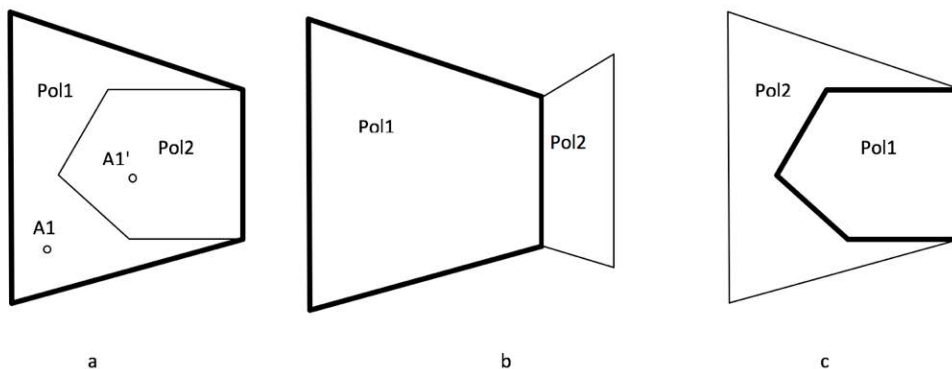


Figure 40. Les trois cas possible quand deux polygones n'ont pas d'intersection franche

Pour isoler un des trois cas, on procède comme suit :

- on calcule un point A1 dans Pol1 (voir Chapitre II.9 - Calcul d'un point à l'intérieur d'un polygone -) ; la Figure 40 montre deux positions possibles de A1 : A1 et A1' ;
- si A1 n'est pas dans Pol2, on sait que Pol1 ne peut pas être inclus dans Pol2 donc soit ils sont disjoints, soit Pol2 est dans Pol1. Pour trancher, on calcule un point A2 dans Pol2 :
 - o si A2 n'est pas inclus dans Pol1, Pol2 ne peut pas être inclus dans Pol1 donc ils sont disjoints ;
 - o sinon, Pol2 est inclus dans Pol1.
- Si A1 est dans Pol2, il ne faut pas conclure que Pol1 est inclus dans Pol2 (A1' en est une preuve) ; on peut juste conclure que Pol1 et Pol2 ne sont pas disjoints donc que l'un est inclus dans l'autre. Pour savoir lequel, l'intuition nous incite, à tort, à procéder comme dans le cas précédent : calculer un point A2 dans Pol2 et, selon qu'il appartient à Pol1 ou pas, décider que l'incluant est Pol1 ou Pol2 : la Figure 41 nous montre deux cas dans lesquels A1 est dans Pol2, A2 est dans Pol1 mais l'incluant est tantôt Pol1, tantôt Pol2.

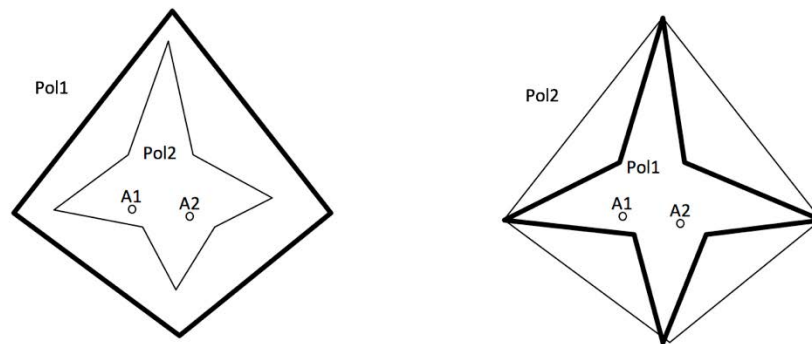


Figure 41. Deux cas dans lesquels l'incluant de A1 et A2 est différent

Une autre approche pour trouver l'englobant est de chercher celui qui s'étend le plus loin dans une quelconque direction, par exemple vers la droite. Celui-ci est forcément l'englobant. Malheureusement le cas de droite illustre le fait que le choix de la direction n'est pas immédiat puisque l'inclus s'étend aussi loin dans les quatre directions principales que l'incluant. Il faudrait donc avoir recours à des raisonnements plus subtils, soit pour choisir la bonne direction, soit pour étudier les positions respectives des segments des deux polygones quand ils s'étendent aussi loin dans la direction étudiée. Et encore cela ne réglerait-il pas le cas où les deux polygones sont exactement superposés.

Finalement, le fait que les contours que nous manipulons soient des polygones nous offre une solution très simple : nous calculons l'aire des deux polygones. Si les deux aires sont égales, les deux polygones sont superposés (parce que l'un inclut l'autre !), sinon celui de plus grande aire est l'incluant. Notons que si les contours manipulés n'étaient pas des polygones, le calcul des aires ne serait pas forcément possible (en tout cas pas de manière exacte) et cette approche ne serait plus forcément applicable.

Cette solution a aussi le mérite d'être dans la même classe de complexité que la précédente : cette dernière, puisqu'elle oblige à parcourir les deux polygones pour trouver le point le plus éloigné, est linéaire en le nombre de sommets des polygones et nous avons vu que c'est aussi le cas du calcul de l'aire d'un *polygone*. L'utilisation des aires amène donc un gain en simplicité mais sans perte en performances.

16 Forme raisonnable

Un polygone a une forme raisonnable si un architecte ou un futur occupant accepterait qu'il soit le pourtour d'une pièce. C'est donc un critère subjectif mais, comme nous le verrons, le processus d'amélioration d'un agencement produit parfois des formes très inhabituelles et un mécanisme d'évaluation de la forme des pièces est absolument nécessaire.

Certaines personnes attachent de l'importance au fait de pouvoir placer des meubles le long des murs (alors que d'autres préfèrent ne pas avoir de meubles mais plutôt des placards encastrés dans les murs), certains ne veulent que des angles multiples de 90° (alors que d'autres apprécient des formes globalement circulaires), certains préfèrent des pièces rectangulaires (d'autres recherchent les pièces en L ou en U). Il faut donc veiller à ce que la manière d'évaluer la forme ne privilégie aucun de ces scénarios.

La Figure 42 (une cuisine en étoile) donne l'exemple d'une cuisine qui répond aux critères :

- partager une porte avec le salon ;
- avoir une surface d'au moins 8 m^2 ;
- être à moins de 4 m de la porte d'entrée ;
- ne pas être orientée au sud.

Pourtant, personne ne voudrait d'une telle cuisine, en particulier parce qu'il est impossible d'y placer de l'électroménager.

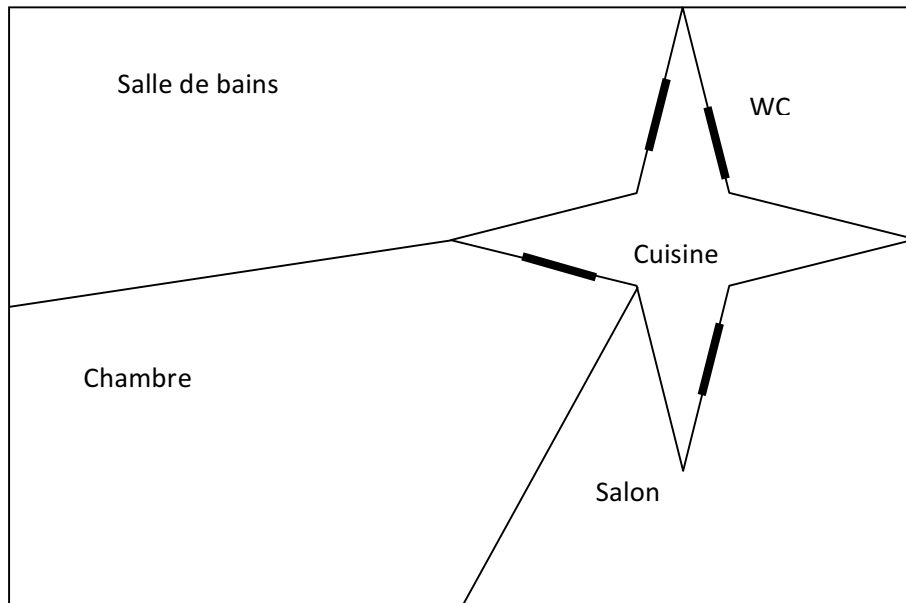


Figure 42. *Un aménagement adapté aux exigences de l'occupant mais avec une cuisine en étoile*

Des critères doivent donc éviter de retenir des formes aussi déraisonnables. En l'occurrence, nous en utilisons deux : "avoir une forme raisonnable" et "maîtrise des angles". Seul le premier de ces deux aspects, qui fait l'objet du paragraphe suivant, présente des difficultés d'ordre géométrique et mérite de figurer dans ce chapitre.

La première mesure qui vient à l'esprit quand on veut estimer la forme d'une pièce est le rapport de ses deux dimensions. Mais ceci ne se conçoit que pour des formes rectangulaires, ce qui est très improbable dans notre cas car nos pièces peuvent avoir plus de quatre côtés et des angles différents de 90° .

Faire le rapport des dimensions d'un englobant rectangulaire n'est pas plus utile car la cuisine en étoile ci-dessus, qui s'inscrit dans un carré, serait alors bien notée.

Nous avons constaté que les formes aberrantes étaient caractérisées par un périmètre excessif, rapporté à la surface de la pièce. Nous avons donc utilisé, comme indicateur d'une forme non souhaitable, le rapport entre le périmètre de la pièce et le périmètre d'un carré qui aurait la même aire que la pièce. $\sqrt{\text{Aire}}$ correspondant à l'arête d'un carré de même aire que la pièce, ce rapport est traduit par la formule suivante :

$$\text{Aspect} = \frac{\text{Périmètre}}{4 \cdot \sqrt{\text{Aire}}}$$

Plus ce rapport s'éloigne de 1, moins la forme de la pièce est acceptable.

Ce bref développement (décalé par rapport au reste du texte) vise à justifier l'utilisation du rapport d'aspect comme une mesure du caractère raisonnable de la forme d'une pièce.

Observons d'abord le comportement du rapport d'aspect sur des rectangles (et non des pièces de formes quelconques) ayant tous la même aire A et des rapports longueur/largeur variés : dans la table 1, ils sont triés du moins allongé - le carré - au plus allongé.

Si on appelle δ le rapport de la plus grande dimension d'un rectangle à la plus petite, le rapport d'aspect devient :

$$Aspect = \frac{1 + \delta}{2 \cdot \sqrt{\delta}}$$

On peut noter que cette quantité ne dépend que des proportions du rectangle, ce qui vient à l'appui du fait que le rapport d'aspect caractérise bien l'allure de la forme évaluée (ici un rectangle).

Pour aller plus loin dans cet exemple, la première ligne de la table 1 présente des rectangles de plus en plus disproportionnés (d'une à quatre fois plus larges que hauts). La deuxième ligne donne leurs rapports d'aspect, qui varient de 1 à 1.25. On conclut que plus le rapport d'aspect grandit, plus le rectangle est disproportionné.




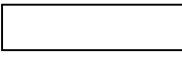
<i>DimMax/DimMin</i> (δ)				
	1	2	3	4
Aspect	1	$3\sqrt{2}/4 \approx 1.06$	$2/\sqrt{3} \approx 1.15$	$5/4 = 1.25$

Tableau 3 Quelques rapports d'aspect pour des rectangles de même aire

Enfin, pour la cuisine en étoile, le rapport d'aspect est de $\sqrt{17/5}$, soit environ 1.84, il est donc bien plus mauvais que celui du quatrième rectangle de la table. Pour qu'un rectangle ait une si mauvaise note, il faudrait que sa longueur soit plus de 11 fois sa hauteur. Le rapport d'aspect qualifie donc de manière satisfaisante la forme de cette cuisine.

Ce sont ces constats qui nous ont amené à choisir le rapport d'aspect comme une mesure de la qualité intrinsèque de la forme d'une pièce. Evidemment, d'autres mesures, dépendant cette fois des préférences de l'architecte ou de l'occupant, peuvent venir compléter cette évaluation.

17 Détection des couloirs

Le logiciel d'optimisation d'un plan part d'un plan existant, qui doit être disponible dans un fichier de données. Pour faciliter la création de ce fichier, on n'impose pas de décrire les couloirs : le fichier contient un ensemble de segments numérotés qui correspondent aux murs puis, pour chaque pièce,

on énumère les numéros de ses murs. Les couloirs ne sont pas explicites et il est donc nécessaire de découvrir les murs qui les limitent ; à partir de là, ils peuvent être traités comme toute autre pièce (calcul d'aire, d'itinéraires ...). Cette opération est appelée la détection des couloirs.

Elle est également utile *pendant* la phase d'optimisation d'un aménagement. En effet, l'optimisation consiste à modifier un grand nombre de fois le logement et certaines de ces modifications ont pour but de faire disparaître ou apparaître des couloirs (voir Chapitre IV.4.1 - les opérateurs de modifications de bas niveau -). Il faut donc souvent lancer l'opération de détection des couloirs.

Appelons *orphelin* un mur qui n'appartient qu'à une seule pièce (éventuellement la pièce fictive qui correspond au pourtour du logement, voir Chapitre IV - Amélioration d'un aménagement - pour plus de détails concernant la structure de données utilisée pour décrire le logement. L'algorithme de base pour détecter un couloir est de trouver un premier mur orphelin, puis un deuxième qui commence là où le premier s'arrête et de poursuivre jusqu'à refermer. On a alors reconstitué un couloir et l'opération est répétée tant qu'il existe des murs orphelins (Figure 43).

Deux situations exigent davantage d'explications : 1) il y a plusieurs segments orphelins à l'extrémité du segment courant et 2) la détection de la fermeture.

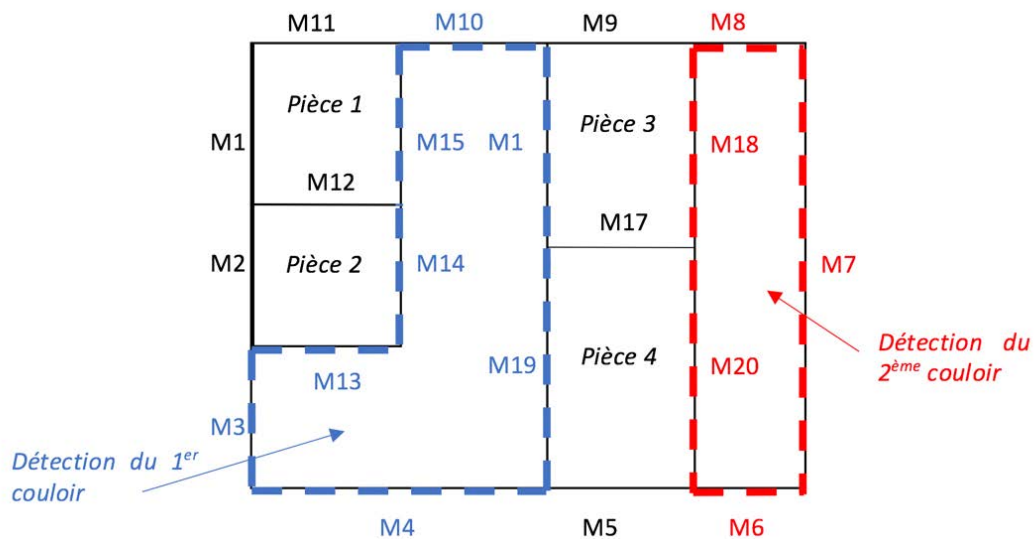


Figure 43. Exemple de détection de deux couloirs

Choix du prochain segment parmi plusieurs segments orphelins

La situation est illustrée par le plan suivant. Il s'agit d'un logement à quatre pièces (grisées) et 17 segments/murs : M0 .. M16.

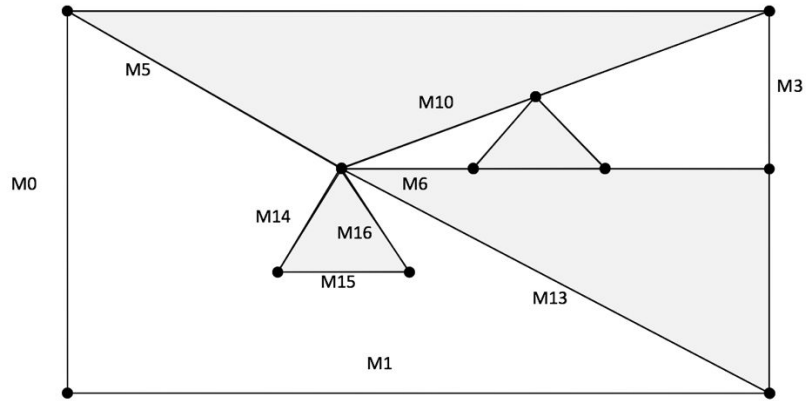


Figure 44. *Choix du prochain mur dans la reconstruction d'un couloir*

Imaginons que la reconstruction d'un couloir a pour l'instant sélectionné M1 et M13. Les possibilités de poursuite sont M16, M14, M6, M10 et M5. Le bon choix, dans ce cas, est M16 (et il devra être suivi de M15, M14, M5 et M0). Ce choix est illustré par la figure suivante et dicté par l'algorithme suivant :

Appelons A le point où il y a indécision

Appelons U le vecteur issu de A et revenant sur le dernier segment sélectionné

Appelons V_i les vecteurs issus de A et s'éloignant sur les divers segments orphelins possibles

Appelons θ_i les angles formés par U et les V_i , exprimés dans $[0, 2. \pi[$ (en pointillés dans la figure ci-dessous)

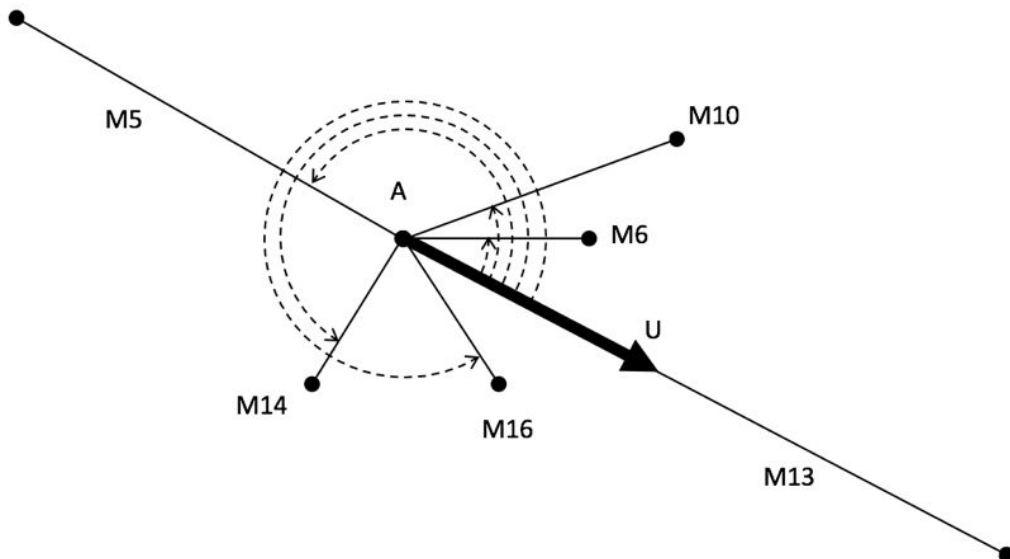


Figure 45. *Choix du prochain mur dans la reconstruction d'un couloir*

Si, le long du dernier segment parcouru (M13 sur l'exemple), le couloir est à gauche (c'est le cas sur l'exemple) alors, on choisit le segment candidat de plus grand θ_i (M16 sur l'exemple), sinon, on choisit le segment candidat de plus petit θ_i (M6 sur l'exemple).

Détection de la fermeture

On ne peut se contenter de considérer que la reconstruction d'un couloir est terminée quand l'extrémité du dernier mur choisi coïncide avec l'origine du tout premier segment.

Reprenons l'exemple précédent et supposons cette fois que la reconstruction a commencé avec M5, suivi de M0, M1, M13. On conclurait alors à la fermeture du couloir (car M13 se termine bien sur M0) alors qu'il faut encore ajouter M16, M15 et M14. Le critère d'arrêt convenable est donc que le prochain segment choisi soit le segment de départ.

18 Distance d'un segment à un obstacle

Définition

La distance évoquée dans ce paragraphe est la distance maximale de translation d'un mur perpendiculairement à son axe. Il faut donc bien avoir à l'esprit que la distance évoquée ici n'est pas la distance au sens géométrique habituel. La figure suivante illustre cette différence.

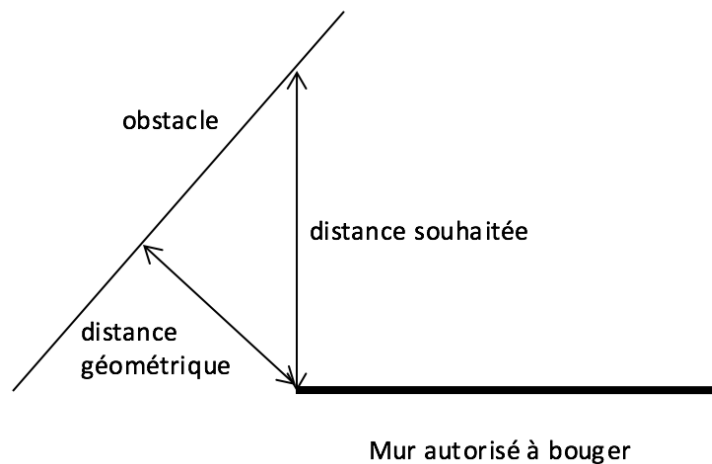


Figure 46. Distance d'un segment à un obstacle

Motivation

Nous avons besoin de cet opérateur lors de la procédure de suppression des couloirs inutiles, évoquée dans la synthèse d'un aménagement (Chapitre III – Synthèse d'un aménagement -). Dans cette procédure, on a besoin de savoir de combien il est possible d'éloigner le mur d'une pièce pour agrandir cette pièce autant que possible. On dispose d'une liste d'obstacles et il faut calculer la distance du mur à chaque obstacle.

Algorithme

On remarque qu'une donnée indispensable est la direction dans laquelle le mur est autorisé à translater. Dans la figure précédente, il était sous-entendu que le mur devait aller vers le haut. S'il devait aller vers le bas, l'obstacle représenté ne limiterait pas le mouvement du mur.

Pour rappel et comme nous l'avons vu au début de ce chapitre, les vecteurs v qui ont un produit scalaire positif avec le vecteur u sont ceux qui vont globalement dans le même sens que u , c'est-à-dire

les vecteurs v qui font un angle de moins de 90° avec u (Figure 47). Ceci nous permettra d'établir la position d'un obstacle relativement au mur.

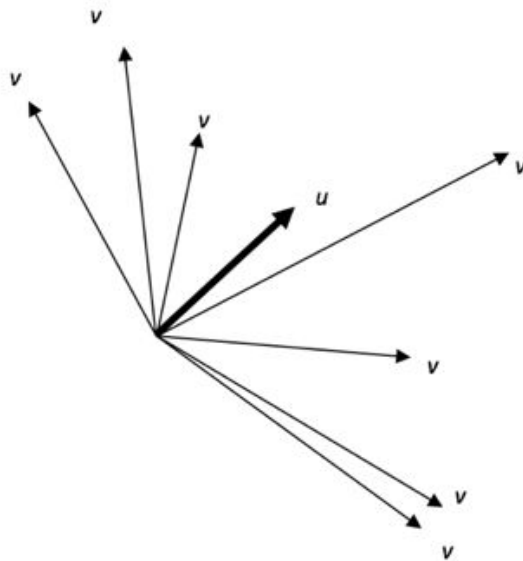


Figure 47. Les vecteurs v ayant un produit scalaire positif avec u

Fixons quelques notations.

La direction de déplacement d'un mur est indiquée par un vecteur unitaire appelé Dir ; dans la Figure 46 Dir serait $(0, 1)$. Les extrémités du mur sont les points A et B et on suppose que A et B sont placés de sorte que, lorsqu'on va de A vers B , la direction indiquée par Dir est le côté gauche du mur. Les extrémités d'un obstacle sont C et D . La figure suivante montre un mur en trait épais et plusieurs obstacles en traits fins.

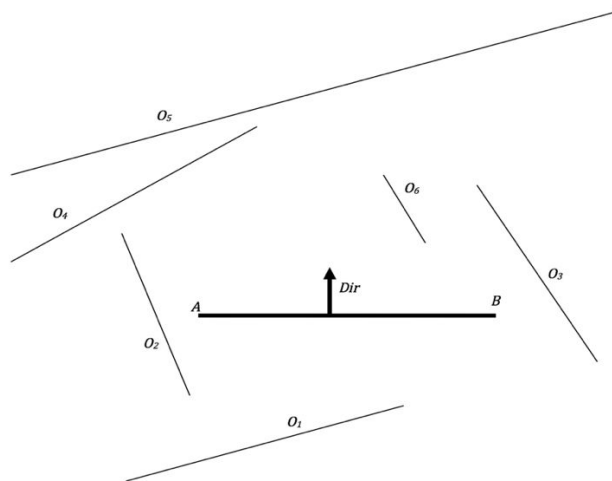


Figure 48. Un mur et des obstacles

Pour savoir si un obstacle potentiel limite le déplacement d'un mur, sa partie qui n'est pas " devant " le mur est d'abord supprimée. La Figure 49 montre en pointillés les parties supprimées des obstacles de la figure précédente.

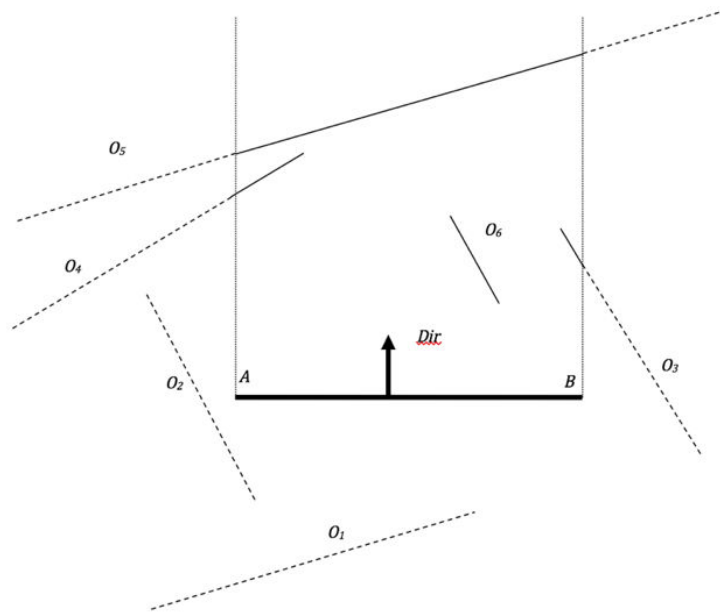


Figure 49. Les obstacles réduits à leur partie utile

Cette opération de troncature d'un obstacle est analogue au découpage (ou clipping) [38] utilisé en algorithmique géométrique pour réduire un segment quelconque (pour nous ce segment est un obstacle) à sa partie incluse dans un rectangle (pour nous, le rectangle est limité en bas par le mur, à gauche et à droite par les perpendiculaires au mur issues de A et B et il est infiniment haut donc il n'a pas de limite supérieure). Dans les lignes qui suivent, cet algorithme de découpage est adapté au fait que le rectangle n'est pas limité vers le haut.

Pour faciliter la présentation, on dira qu'un point est "derrière" (comme les extrémités de O_1) ou "à gauche" (comme une des extrémités de O_1) ou "à droite" du mur s'il est situé dans les zones correspondantes quand le mur est dessiné horizontalement, Dir pointant vers le haut. Toutefois, les formules présentées ci-dessous sont valides quelle que soit l'orientation du mur. Par ailleurs, nous appelons droites *limites* les trois droites qui limitent le rectangle en bas, à gauche et à droite.

Pour savoir si un point P est "derrière" le mur $[A, B]$ (donc sous la première droite limite), on teste si

$$\text{Derriere } (P, A, \text{Dir}) \text{ c'est-à-dire si } AP \cdot \text{Dir} < 0$$

Pour savoir si un point P est "à gauche" du mur $[A, B]$ (donc à gauche de la deuxième droite limite), on teste si

$$\text{AGauche } (P, A, B) \text{ c'est-à-dire si } AP \cdot AB < 0$$

Pour savoir si un point P est "à droite" du mur $[A, B]$, on teste si

$$\text{ADroite } (P, A, B) \text{ c'est-à-dire si } BP \cdot AB > 0$$

Pour réduire un obstacle $[C, D]$ à sa partie utile, on supprime successivement sa partie "derrière" puis sa partie "à gauche" et enfin sa partie "à droite" du mur à translater jusqu'à ce qu'il ne reste rien ou jusqu'à ce que ce qui reste soit entièrement dans la bonne zone (devant le mur donc "dans" le rectangle de hauteur infinie) :

Exprimé en pseudo code, nous avons :

```
Si C et D sont tous les deux à droite du rectangle, ou tous les
deux à gauche ou tous les deux derrières,

    l'obstacle est entièrement extérieur donc il n'en reste rien

Sinon

    Si C et D sont dans le rectangle (ni derrière, ni à gauche, ni à
    droite)

        l'obstacle est intérieur, sa réduction est terminée

    Sinon

        on choisit une extrémité extérieure au rectangle, on la
        déplace sur l'une des droites limites au-delà de laquelle
        elle se trouve (le segment est ainsi raccourci) et on
        recommence

    FSi

FSi
```

Algorithme 1 *Troncature d'un obstacle*

Une fois qu'un obstacle est réduit à sa partie utile, celle qui est "devant" le mur à bouger (à supposer qu'il en reste une) et qui va effectivement limiter la translation du mur, on obtient la "distance" du mur à l'obstacle en prenant le minimum des distances (au sens géométrique, cette fois) des deux extrémités de l'obstacle au mur. Ces deux distances sont obtenues avec deux derniers produits scalaires mais on utilise cette fois une autre propriété du produit scalaire : la distance d'un point P à son projeté orthogonal sur la droite (A, B) de normale normée *Dir* est donnée par la formule qui suit, car *Dir* est normé :

$$Distance(P, (A, B)) = |AP \cdot Dir|$$

Chapitre III --- Synthèse d'un aménagement

1. Introduction

Ce chapitre présente une méthode basée sur la programmation linéaire mixte pour concevoir un aménagement d'un logement adapté au futur occupant de ce logement, en particulier quand ce dernier présente un handicap. Les entrées de la méthode sont de deux sortes :

1. des données de nature numérique, comme les dimensions du logement à agencer, le nombre et la nature des pièces ;
2. un ensemble de contraintes. Ces contraintes sont soit géométriques et architecturales (pas de chevauchements entre les pièces, une zone donnée doit être à l'intérieur d'une pièce, ...), soit des souhaits exprimés par le futur occupant (cuisine orientée à l'Est) ou des besoins imposés par son état de santé (maximiser la distance entre le salon et la chambre à coucher). La Figure 50 montre un aménagement en 2D produit par l'application qui implémente cette méthode.

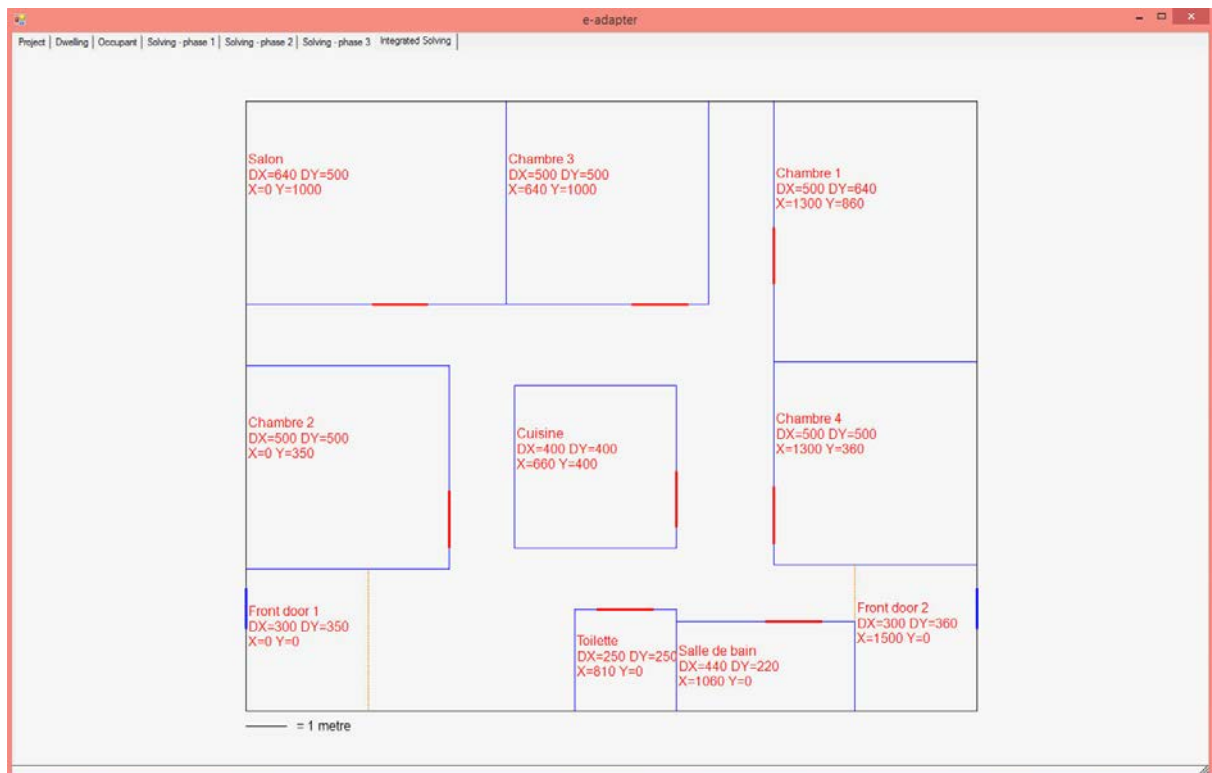


Figure 50. Aménagement généré automatiquement pour un habitat de dimensions 10x15 mètres

À noter que cette étude n'inclut pas le mobilier. Le but est le placement et le dimensionnement des pièces, des portes, et des couloirs.

La méthode proposée dans ce chapitre se focalise sur la *synthèse* d'un aménagement (et non l'amélioration d'aménagement existant, qui fait l'objet du chapitre suivant). Cette approche consiste à élaborer une nouvelle solution d'aménagement améliorant la qualité du logement en satisfaisant les contraintes liées à la situation de l'usager et en respectant le mieux possible des combinatoires de type bin-packing avec contraintes additionnelles réputées NP-difficiles dont la résolution est appuyée sur des méthodes exactes et dont l'objectif est la recherche d'une seule solution pour une instance de petite taille.

Dans la littérature, étudier le processus de synthèse est souvent réalisé par des méthodes heuristiques, où la solution courante est évaluée et modifiée de manière itérative. Ici, nous explorons une voie alternative basée sur une approche globale et plus directe.

L'approche basée sur la programmation mathématique offre plusieurs avantages :

- la possibilité de produire une solution optimale directement en utilisant un solveur dédié ;
- la possibilité de traduire différentes contraintes énumérées ci-dessus et de les considérer de manière globale ;
- la possibilité d'exploiter des résultats formels de la littérature compte tenu de certaines similitudes que présent ce problème avec ceux de la conception de circuits intégrés [52,34,25], de placement, de découpe ou de bin packing [37,42,85] ; dans ces domaines, la programmation mathématique a souvent produit des résultats intéressants et prometteurs.

La synthèse d'un aménagement est un problème complexe :

- la position des portes dépend à la fois de la position et le placement des pièces, et des murs sur lesquels les portes peuvent être placés (les murs des pièces communs aux murs du pourtour extérieur ne sont pas concernés) ;
- les couloirs sont des zones qui ne peuvent pas être considérés comme des pièces car leur nombre et leurs dimensions ne sont pas connus par avance.

Pour ces raisons, on propose de décomposer le problème en trois étapes :

1. la première étape consiste à gérer les contraintes structurelles qui consiste à placer et dimensionner les pièces, avec une distance minimale entre chaque paire de pièces de $2 \cdot r$ (r étant le rayon du cercle représentant l'habitant), qui correspond à un couloir potentiel ; elle est exprimée comme un problème d'optimisation linéaire en variables mixtes de type découpe ;
2. la deuxième étape part du résultat de la première étape pour ajouter les portes et elle est aussi exprimée comme un problème d'optimisation linéaire en variables entières de type calculs de chemins ;
3. la troisième étape supprime les couloirs inutiles et ainsi, elle agrandit les pièces adjacentes à ces couloirs ; cette étape est géométrique.

Ce chapitre est structuré comme suit. Nous commençons par détailler la première étape, qui a été publié dans la conférence Codit'16 [11], avec une amélioration de ces résultats et quelques commentaires supplémentaires. Il s'agit d'un modèle mathématique sous-forme d'une variante du modèle de bin-packing en 2-dimensions. Un premier modèle quadratique en variables mixtes a été élaboré dont la résolution est rendue très difficile par la non convexité et la non concavité des contraintes. Un deuxième modèle exploitant les méthodes de linéarisation est proposé pour remédier aux limites du premier modèle. La section suivante détaille la deuxième étape qui s'appuie sur un modèle de calculs de chemins dans un graphe de mobilité intégrant des caractéristiques du futur occupant. La troisième étape, objet de la section suivante, présente une heuristique d'optimisation de

l'espace utile. Et pour finir, nous montrons que la méthode permet de produire des aménagements adaptés à des personnes handicapées.

Pour éviter un peu de lourdeur dans la rédaction, nous emploierons indifféremment les termes mur ou coin gauche, droit, haut, bas et mur Nord, Sud, Est, Ouest pour désigner le placement relatif des murs ou des coins. Par exemple, un mur situé à gauche du plan est un mur qui est à l'ouest.

2. Placement et dimensionnement des pièces

La première étape du processus de synthèse d'un agencement est le placement et le dimensionnement des pièces. Ceci est fait avec l'objectif principal de garantir que chaque pièce peut être atteinte depuis n'importe quelle autre. Pour cela, on libère une bande d'épaisseur $2 \cdot r$ autour de chaque pièce ; on garantit ainsi qu'il est possible de circuler autour de chaque pièce et, de proche en proche, depuis toute pièce vers toute autre pièce. A terme, on constatera peut-être que certaines de ces zones sont inutiles ; elles seront supprimées et la place ainsi récupérée sera affectée aux pièces adjacentes [11].

Ce problème peut être vu comme variante du problème de bin-packing dans le sens où les tailles et les dimensions des objets sont variables (elles sont définies dans un intervalle au lieu d'être fixées au préalable) alors que le contenant est de taille et de dimensions imposées.

Pour simplifier la formulation du problème, les conventions suivantes ont été adoptées : le logement et les pièces sont tous de forme rectangulaire. Les côtés des pièces et du logement sont soit horizontaux soit verticaux; le bord droit est à l'Est et le bord supérieur est au nord ; le coin en bas à gauche est l'origine; les distances sont en centimètres et aucune distance ou coordonnée n'est inférieure à un centimètre; la porte d'entrée est associée à une pièce spéciale (une entrée) qui est considérée comme toute autre pièce; l'habitant est approché par un cercle dont le diamètre dépend de l'encombrement de la personne ($0.3 m$ pour une personne de corpulence dite normale, $0.75 m$ pour une personne en fauteuil roulant...).

2.1. Première approche de modélisation mathématique (modèle 1)

2.1.1. Les entrées

Les entrées de ce modèle sont des données et des contraintes. Les contraintes seront détaillées plus loin.

Les propriétés évidentes concernant les données suivantes ne sont pas énumérées explicitement. Par exemple, le nombre de pièces doit être positif, la largeur maximale d'une pièce doit être inférieure ou égale à la largeur du logement, etc. Les données sont :

n : le nombre de pièces à placer.

DX, DY : la longueur et la largeur du logement.

r : le rayon du cercle représentant l'habitant. Chaque couloir doit avoir une largeur minimale de $2 \cdot r$ afin de permettre à une personne par exemple en fauteuil roulant de faire demi-tour [28].

$MinDim1_i, MinDim2_i, MaxDim1_i, MaxDim2_i$: les bornes des deux dimensions de la pièce i . Ces valeurs peuvent être fournies par l'habitant ou par la réglementation en termes de construction (Par exemple, on souhaite que la chambre ait une surface de $9m^2$ et que chacune des deux dimensions soit supérieure à $2.5m$). On ne fixe pas a priori, parmi les deux dimensions $Dim1$ et $Dim2$ d'une pièce,

laquelle sera la largeur et laquelle sera la longueur. Le système d'optimisation est libre de choisir l'orientation. Ce qui constitue d'ailleurs une source de difficulté supplémentaire dans le modèle puisque la prise en compte des conditions de disjonctions associées introduit des contraintes de type quadratique. Les valeurs $MinDim1_i$ et $MinDim2_i$ permettent également de répondre à des conditions en lien avec le handicap du futur occupant pour garantir l'accessibilité et la mobilité au sein des différentes pièces.

$MinX_i, MinY_i, MaxX_i, MaxY_i$: respectivement, les bornes inférieures du coin inférieur gauche et les bornes supérieures du coin supérieur droit de la pièce i . La pièce i doit commencer au-delà des deux premières valeurs et doit terminer avant les deux dernières. Ces quatre valeurs permettent dans le cas où, une contrainte de positionnement sur la pièce i est formulée (Par exemple si on veut une pièce le long du mur Est), de limiter la zone dans laquelle cette pièce peut être placée. Si aucune contrainte de positionnement sur la pièce i n'est formulée, la zone de placement considérée est tout le logement.

$MinEndX_i, MinEndY_i, MaxStartX_i, MaxStartY_i$: respectivement, les bornes inférieures du coin supérieur droit et les bornes supérieures du coin inférieur gauche de la pièce i . La pièce i doit terminer au-delà des deux premières valeurs ($MinEndX_i, MinEndY_i$) et doit commencer avant les deux dernières valeurs ($MaxStartX_i, MaxStartY_i$). Ces quatre valeurs sont utilisés pour s'assurer qu'une pièce contient une zone donnée, tel qu'une évacuation d'eau ou le conduit de cheminée. Si une pièce doit contenir plusieurs zones d'intérêt, on utilise ces valeurs pour décrire un englobant rectangulaire de toutes les zones.

2.1.2. Les variables

Les variables sont :

X_i, Y_i : les coordonnées du coin inférieur gauche de la pièce i .

$Dim1_i, Dim2_i$: les deux dimensions de la pièce i sans la considération de l'orientation. Ainsi, **$Dim 1_i$** peut-être la dimension le long de l'axe x ou la dimension le long de l'axe y, de même pour **$Dim 2_i$** .

DX_i, DY_i : les dimensions de la pièce i le long des axes x et y .

$L_{ij}, L_{ji}, U_{ij}, U_{ji}$: des variables binaires en $(0, 1)$ qui définissent la position relative de la pièce i par rapport à la pièce j dans le logement (L comme Left et U comme Up). $L_{ij} = 1$ si et seulement si la pièce i est à gauche de la pièce j c'est à dire : $X_i + DX_i < X_j$ et $U_{ij} = 1$ si et seulement si la pièce i est au-dessus de la pièce j c'est à dire $Y_j + DY_j < Y_i$ [42].

$LE_i, RE_i, TE_i, BE_{ji}$: des variables binaires en $(0, 1)$. Elles expriment le fait que la pièce est adjacente au bord gauche (Left Edge) au bord droit (Right Edge) au bord haut (Top Edge) ou au bord bas (Bottom Edge) du logement. Par exemple, $LE_i = 1$ si la pièce i est adjacente au bord gauche du logement ; si ce n'est pas le cas, sa valeur peut être 0 ou 1.

$NLC_i, NRC_i, NBC_i, NTC_i$: des variables binaires en $(0, 1)$ qui signifient : pas de recouvrement à gauche (No Left Cover), à droite (No Right Cover), en bas (No Bottom Cover) et en haut (No Top Cover) du logement. Elles expriment le fait que la pièce i ne recouvre pas la totalité d'un des quatre côtés du logement. Par exemple, $NLC_i = 0$ si la pièce i relie les deux murs horizontaux du logement (ce qui revient à dire que sa largeur est égale à la largeur du logement) et se trouve contre le mur gauche de l'habitation. Sinon sa valeur peut être 0 ou 1 (voir les justifications de contraintes un peu plus loin).

O_i : variable binaire en $(0, 1)$ qui définit l'orientation de la pièce i . Quand $O_i = 0$, la pièce n'est pas tournée, c'est sa première dimension $Dim1_i$ qui est le long de l'axe des abscisses et $Dim2_i$ qui est le long de l'axe des ordonnées. Sinon, la pièce est tournée de 90° et $Dim1_i$ et $Dim2_i$ sont respectivement sur le long de l'axe des ordonnées et de l'axe des abscisses.

$Aux1_i, Aux2_i$: Ces variables sont utilisées pour linéariser quelques contraintes quadratiques. Elles remplacent respectivement les expressions $Dim1_i * O_i$ et $Dim2_i * O_i$ dont la nécessité est justifiée un peu plus loin.

2.1.3. Le modèle mathématique

Au final, le modèle mathématique est le suivant :

$$\text{Maximiser } f = \sum_1^n DX_i + DY_i$$

sous les contraintes

$$\left\{ \begin{array}{ll} (1) X_i \geq MinX_i & (*) \\ (2) Y_i \geq MinY_i & (*) \\ (3) X_i + DX_i \leq MaxX_i & (*) \\ (4) Y_i + DY_i \leq MaxY_i & (*) \\ \\ (5) X_i \leq MaxStartX_i & (*) \\ (6) Y_i \leq MaxStartY_i & (*) \\ (7) X_i + DX_i \geq MinEndX_i & (*) \\ (8) Y_i + DY_i \geq MinEndY_i & (*) \\ \\ (9) L_{ij} + L_{ji} + U_{ij} + U_{ji} \geq 1 & (**) \\ (10) X_i + DX_i + 2r * L_{ij} \leq X_j + DX(1 - L_{ij}) & (**) \\ (11) Y_i + DY_i + 2r * U_{ij} \leq Y_j + DY(1 - U_{ij}) & (**) \\ \\ (12) X_i \geq 1 - LE_i & (*) \\ (13) X_i \leq DX(1 - LE_i) & (*) \\ (14) X_i + DX_i \leq DX + RE_i - 1 & (*) \\ (15) X_i + DX_i \geq DX * RE_i & (*) \\ (16) Y_i \geq NBC_i & (*) \\ (17) Y_i \leq DY * NBC_i & (*) \\ (18) Y_i + DY_i \leq DY - NTC_i & (*) \\ (19) Y_i + DY_i \geq DY(1 - NTC_i) & (*) \\ (20) LE_i + RE_i + NTC_i + NBC_i \geq 1 & (*) \\ (21) X_i \geq NLC_i & (*) \\ (22) X_i \leq DX * NLC_i & (*) \\ (23) X_i + DX_i \leq DX - NRC_i & (*) \\ (24) X_i + DX_i \geq DX(1 - NRC_i) & (*) \\ (25) Y_i \geq 1 - BE_i & (*) \\ (26) Y_i \leq DY(1 - BE_i) & (*) \\ (27) Y_i + DY_i \leq DY + TE_i - 1 & (*) \\ (28) Y_i + DY_i \geq DY * TE_i & (*) \\ (29) BE_i + TE_i + NRC_i + NLC_i \geq 1 & (*) \end{array} \right.$$

$$\begin{cases}
(30) DX_i = Aux2_i - Aux1_i + Dim1_i & (***) \\
(31) DY_i = Aux1_i - Aux2_i + Dim2_i & (***) \\
(32) Aux1_i \leq Dim1_i & (***) \\
(33) Aux1_i \geq Dim1_i - DX \cdot (1 - O_i) & (***) \\
(34) Aux1_i \leq DX \cdot O_i & (***) \\
(35) Aux1_i \geq 0 & (***) \\
(36) Aux2_i \leq Dim2_i & (***) \\
(37) Aux2_i \geq Dim2_i - DY \cdot (1 - O_i) & (***) \\
(38) Aux2_i \leq DY \cdot O_i & (***) \\
(39) Aux2_i \geq 0 & (***)
\end{cases}$$

(*) pour chaque pièce contrainte

(**) pour chaque paire pièces i et j ($i \neq j$)

(***) pour chaque pièce i

2.1.4. Explication de la fonction objectif

La fonction coût a pour objectif de minimiser les surfaces perdues. Ces surfaces perdues sont les zones qui ne sont affectées ni à une pièce, ni à un couloir, c'est-à-dire à la bande laissée vide autour de chaque pièce pour garantir l'accès à toutes les pièces. Une manière intuitive de minimiser ces surfaces perdues aurait été de maximiser la somme des aires des pièces : l'aire d'une pièce est le produit des deux dimensions de la pièce et ces deux dimensions sont disponibles dans le modèle (DX_i et DY_i). Mais cela aurait produit une fonction objectif quadratique. Pour éviter cela, nous avons préféré maximiser le périmètre des pièces (plus exactement les demi-périmètres).

Considérant un rectangle avec une aire donnée A et étudions l'effet, sur la forme du rectangle, d'une augmentation de son périmètre sans variation de son aire. Soit w et h ses dimensions ; on a :

$$h = \frac{A}{w}$$

le périmètre P du rectangle est

$$P = 2 \cdot \frac{(w^2 + A)}{w}$$

et sa dérivée par rapport à w est

$$P' = 2 \cdot \left(w - \frac{A}{w} \right) = 2 \cdot \left(1 - \frac{A}{w^2} \right) = 2 \cdot \frac{(w^2 - A)}{w^2}$$

qui est négative jusqu'à \sqrt{A} puis positive. P est décroissant jusqu'à $w = \sqrt{A}$ et croissant ensuite. En d'autres termes, pour une aire donnée, maximiser le périmètre d'un rectangle entraîne une augmentation de la largeur (ou de la longueur puisque le rôle de w et h dans les formules qui précèdent sont symétriques) ce qui engendre des chambres infiniment minces et longues.

Dans notre modèle, la conséquence de ce phénomène est limitée par deux facteurs : premièrement, une valeur minimale est requise pour les deux dimensions de chaque pièce, il y a donc une valeur seuil en dessous de laquelle une dimension ne peut aller, ce qui l'empêche de converger vers 0 ; deuxièmement, la largeur et la longueur d'une pièce ne peuvent pas infiniment croître car elles sont bornées par les dimensions du logement. Cependant, une fois que les deux dimensions sont au-dessus

de leurs seuils, rien n'empêche l'une d'elles de croître jusqu'à atteindre la hauteur ou la largeur de l'habitation, ce qui, en effet, produit parfois des formes allongées. Une mesure facile à implanter pour corriger ce défaut serait de contraindre le rapport des deux dimensions de chaque pièce, de manière à empêcher qu'une pièce ne soit jamais plus de K fois plus large que haute :

$$\frac{1}{K} \leq \frac{DX_i}{DY_i} \leq K$$

Cette formule se linéarise facilement en deux inégalités, en multipliant les trois termes par DY_i .

2.1.5. Justification des contraintes

Comme évoqué ci-dessus, les entrées du problème sont des données et des contraintes. Ces contraintes sont de trois types : contraintes architecturales, souhaits exprimés par l'occupant et contraintes imposées par son état de santé. Pour éviter toute confusion, nous les appelons *SContraintes* (Contraintes Sémantiques) tandis que les inéquations (de (1) à (39)) dans le modèle ci-dessus, qui sont leurs traductions en langage mathématique, sont appelées *MContraintes* (Contraintes Mathématiques). Ce paragraphe présente la liste des *SContraintes* que le modèle peut prendre en compte et la manière dont elles sont traduites en *MContraintes*.

- **Eloignement** : repousse une pièce loin d'un mur extérieur (*MContrainte* (1) à (4)). Par exemple, la *MContrainte* (2) impose que l'ordonnée de début de la pièce i soit supérieure à $MinY_i$, ce qui a pour effet d'éloigner la pièce i du mur sud. Ainsi, si, dans un pays chaud, on souhaite ne pas avoir le salon au sud, il est possible de le spécifier. En fonction de la valeur donnée à la $MinY_i$, le système intercalera soit un couloir, soit une ou plusieurs pièces entières entre le mur sud et le salon ;
- **Pas de chevauchement** : chaque paire de pièces i et j ne doit pas se chevaucher. La *MContraintes* (9) s'assure que la pièce i et la pièce j respectent cette condition : comme la somme est supérieure ou égale à 1, au moins une de ces opérandes doit être égale à 1. En d'autres termes, i doit être à gauche de j ($L_{ij} = 1$), à droite de j ($L_{ji} = 1$), au-dessus de j ($U_{ij} = 1$) ou au-dessous de j ($U_{ji} = 1$), qui est en effet une condition nécessaire et suffisante pour que deux rectangles ne se chevauchent pas [37,85]. Les *MContraintes* (10) et (11) expriment le fait que si i est à gauche (respectivement au-dessus) de j alors, il doit y avoir un espace d'au moins $2.r$ qui les sépare, pour un éventuel futur couloir. Ces deux contraintes garantissent que toute pièce est accessible depuis n'importe quelle autre pièce. Ainsi, la *MContrainte* (10) fonctionne comme suit : si $L_{ij} = 0$ (signifie qu'il n'y a pas de contrainte « à gauche » entre i et j), la formule devient ainsi :

$$X_i + DX_i \leq X_j + DX_j$$

ce qui est toujours vrai. En d'autres termes, la contrainte est redondante. Si $L_{ij} = 1$ (i est à gauche de j), la formule devient :

$$X_i + DX_i + 2.r \leq X_j$$

ce qui signifie que la pièce j ne doit pas être uniquement à droite de la pièce i mais, elle doit être aussi à une distance $2.r$ loin d'elle pour créer un couloir au milieu.

La *MContrainte* (11) est similaire mais sur l'axe Y .

- **Inclusion** : oblige une pièce à contenir une zone rectangulaire donnée (*MContraintes* (5) à (8)). Rappelons que cette zone peut être, par exemple, un mur externe (« je veux que ma cuisine soit orientée côté Est ») ou un englobant d'une arrivée/ évacuation d'eau, d'un conduit de cheminée ;
- **Pas de scission** : une pièce ne doit pas couper le logement en deux régions (Figure 51– à gauche -). Sinon l'occupant est obligé de traverser cette pièce pour passer d'une région à l'autre (*MContraintes* (12) à (29)).

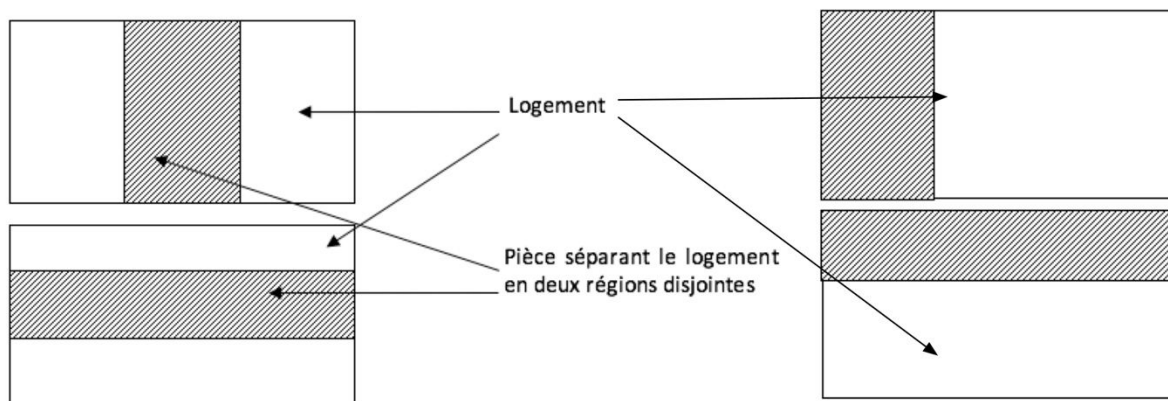


Figure 51. Configuration non souhaitable : pièces scindant le logement (à gauche), une des configurations souhaitable (à droite)

Considérant par exemple les *MContraintes* (12) à (20), qui garantissent que la pièce i n'est pas dans la situation de la première de ces deux figures, à savoir que la pièce i ne sépare pas le logement en deux zones disjointes, l'une à gauche de la pièce, l'autre à droite. Pour que cette situation ne se présente pas, il faut soit que la pièce soit moins large que le logement, soit qu'elle soit aussi large que le logement mais qu'elle soit alors collée contre le bord gauche ou le bord droit du logement (Figure 51- à droite -). Mathématiquement cela se traduit comme suit : soit les *MContraintes* (12) et (13), si $LE_i = 1$, X_i est nécessairement égale à 1, c'est à dire que la pièce i est adjacente au bord gauche du logement. Sinon, X_i doit juste être inférieure à DX . Les *MContraintes* (14) et (15) sont similaires : si $RE_i = 1$, la pièce i est en contact avec le bord droit du logement. Les *MContraintes* (16) et (19) signifient que si $NBC_i = 1$ (respectivement $NTC_i = 1$) alors DY_i est nécessairement plus petit que DY (la pièce n'est pas aussi haute que le logement). Finalement, la contrainte *MContrainte* (20) exige qu'au moins une de ces opérandes soit égale à 1. En d'autres termes, si toutes les opérandes sont à 0, la pièce alors scinde le logement en deux régions.

La section 5 (Gestion du handicap) de ce chapitre explique comment exprimer des contraintes de niveau encore plus élevé (imposées par l'état de santé du futur occupant par exemple) en des *SContraintes* qui, à leur tour, peuvent être traduites en des *MContraintes*, comme expliqué ci-dessus.

Les *MContraintes* de (30) à (39) ne traduisent pas directement des *SContraintes*, elles ont pour but d'empêcher certaines des *MContraintes* déjà citées de devenir quadratiques, ce que nous devons éviter puisque notre objectif est d'obtenir un modèle linéaire de notre problème. Cette élévation du degré survient quand on souhaite laisser le système décider de l'orientation des pièces. Cette orientation consiste à laisser la pièce dans son orientation de base, auquel cas sa première dimension est sa portée en x et sa deuxième est sa portée en y , ou bien à la tourner d'un quart de tour ; dans ce cas, les deux dimensions s'inversent. L'orientation d'une pièce est considérée en ajoutant, pour chaque pièce i une variable binaire O_i , où $O_i = 1$ quand la pièce est tournée de 90° à 0° sinon, et deux variables, $Dim1_i$ et $Dim2_i$ qui correspondent aux deux dimensions de la pièce. Le système est alors libre de choisir $Dim1_i$ ou $Dim2_i$ comme en x sur l'axe des abscisses de la pièce i : la dimension x (largeur) d'une pièce est $Dim1_i$ si la pièce n'a pas été tournée, sinon c'est $Dim2_i$:

$$DX_i = Dim1_i \cdot (1 - O_i) + Dim2_i \cdot O_i$$

$$= Dim2_i \cdot O_i - Dim1_i \cdot O_i + Dim1_i \quad (a)$$

et sa dimension y est :

$$\begin{aligned} DY_i &= Dim2_i \cdot (1 - O_i) + Dim1_i \cdot O_i \\ &= Dim1_i \cdot O_i - Dim2_i \cdot O_i + Dim2_i \quad (b) \end{aligned}$$

La conséquence de cette extension est que la *MContrainte* (3), par exemple, qui utilise DX_i , devient quadratique comme suit :

$$X_i + Dim2_i \cdot O_i - Dim1_i \cdot O_i + Dim1_i \leq MaxX_i$$

Pour éviter cela, les inéquations (a) et (b) sont linéarisées. La linéarisation est faite en utilisant des techniques de base qu'on peut retrouver en détails dans [8,33,26]: chaque produit est remplacé par une nouvelle variable ($Aux1_i$ remplace $Dim1_i * O_i$, $Aux2_i$ remplace $Dim2_i * O_i$) et quatre nouvelles contraintes sont ajoutées au modèle pour que la nouvelle variable ait la même valeur que le produit remplacé ; par exemple (a) et (b) deviennent :

$$DX_i = Aux2_i - Aux1_i + Dim1_i \quad (a')$$

$$DY_i = Aux1_i - Aux2_i + Dim2_i \quad (b')$$

où $Aux1_i$ est contraint par :

$$Aux1_i \leq Dim1_i \quad (A1)$$

$$Aux1_i \geq Dim1_i - DX \cdot (1 - O_i) \quad (B1)$$

$$Aux1_i \leq DX \cdot O_i \quad (C1)$$

$$Aux1_i \geq 0 \quad (D1)$$

et $Aux2_i$ est contraint par :

$$Aux2_i \leq Dim2_i \quad (A2)$$

$$Aux2_i \geq Dim2_i - DY \cdot (1 - O_i) \quad (B2)$$

$$Aux2_i \leq DY \cdot O_i \quad (C2)$$

$$Aux2_i \geq 0 \quad (D2)$$

Quand O_i est égale à 0 (la pièce i n'est pas tournée), (C1) et (D1) forcent $Aux1_i$ à prendre la valeur 0, (A1) et (B1) deviennent redondantes. En effet, $Aux1_i$ est alors nulle donc A1 est nécessairement satisfaite ; de même, $Dim1_i$ est alors la dimension en x de la pièce i , (puisque $O_i = 0$) et cette dimension est nécessairement inférieure à DX (la largeur du logement) ; le terme droit de (B1) est donc négatif et B1 revient à dire que 0 est supérieur ou égal à un nombre négatif, ce qui est toujours vrai. (C2) et (D2) forcent $Aux2_i$ à prendre la valeur 0, (A2) et (B2) sont forcément satisfaites aussi et, finalement, (a') et (b') deviennent alors respectivement $DX_i = Dim1_i$ et $DY_i = Dim2_i$, ce qui est le résultat escompté. Quand O_i est égale à 1, il est facile de constater que DX_i et DY_i sont respectivement égales à $Dim1_i$ et $Dim2_i$ ce qui est aussi le résultat escompté. De ce fait (a') et (b') donnent le même résultat que (a) et (b) mais elles sont maintenant linéaires.

Ainsi, on aboutit à un modèle mathématique en variables mixtes proche du problème de bin-packing, avec des contraintes additionnelles et une spécificité, à savoir des objets à tailles variables. Pour la suite de l'étude, nous avons retenu la version linéarisée.

Dans la section suivante, nous proposons un deuxième modèle mathématique basé sur une discrétisation de l'espace. Cette approche de modélisation évite de générer des expressions quadratiques et permet de pallier la principale source de difficulté de résolution du modèle précédent.

2.2. Deuxième approche de modélisation mathématique (modèle 2)

Cette approche se déroule en deux étapes :

- étape 1 : consiste à générer, pour chaque pièce un ensemble de formes potentielles, tout en respectant sa dimension minimale et sa dimension maximale. La distance entre deux formes consécutives générées, pour une pièce donnée, est régie par un paramètre appelé *pas de discrétisation* (Figure 52).

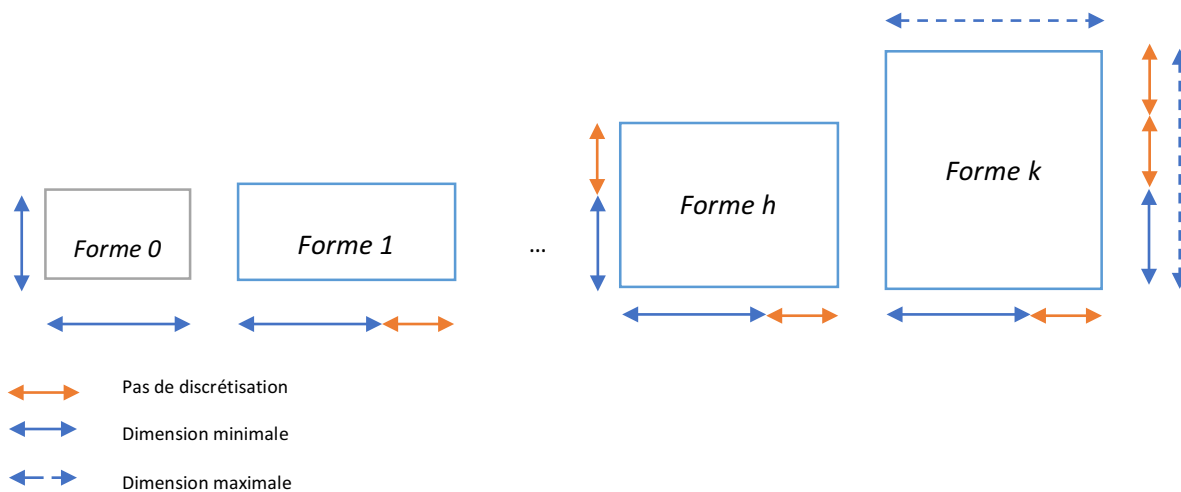


Figure 52. Pas de discrétisation et formes potentielles pour une pièce

- étape 2 : consiste à choisir, à l'aide du modèle mathématique, en respectant toutes les contraintes du problème, une forme par pièce afin de maximiser la surface utile.

2.2.1. Les entrées

Les entrées sont les formes des pièces générées, exprimées comme suit :

- DX_i^k, DY_i^k : les dimensions de la forme k de la pièce i le long des axes x et y ;
- S_i^k : la surface de la forme k de la pièce i .

2.2.2. Les variables

Les variables sont les mêmes pour les deux approches à deux différences :

- DX_i, DY_i : ne sont plus nécessaires car elles sont remplacées par les DX_i^k, DY_i^k pour chaque forme k générée de chaque pièce i ;
- F_i^k : 1 si et seulement si la forme k de la pièce i est retenue ; 0 sinon.

2.2.3. Le modèle mathématique

Au final, le modèle mathématique est le suivant :

$$\text{Maximiser } f = \sum_{i=1}^n \sum_{k=1}^{NF_i} F_i^k * S_i^k \text{ Avec } S_i^k = DX_i^k * DY_i^k$$

sous les contraintes

$$\left\{ \begin{array}{l} (1) \sum_{k=1}^{NF_i} F_i^k = 1 \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (2) X_i \geq MinX_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (3) Y_i \geq MinY_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (4) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) \leq MaxX_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (5) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \leq MaxY_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (6) X_i \leq MaxStartX_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (7) Y_i \leq MaxStartY_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (8) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) \geq MinEndX_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (8) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \geq MinEndY_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (9) L_{ij} + L_{ji} + U_{ij} + U_{ji} \geq 1 \end{array} \right. \quad (**)$$

$$\left\{ \begin{array}{l} (10) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) + 2r * L_{ij} \leq X_j + DX(1 - L_{ij}) \end{array} \right. \quad (**)$$

$$\left\{ \begin{array}{l} (11) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) + 2r * U_{ij} \leq Y_j + DY(1 - U_{ij}) \end{array} \right. \quad (**)$$

$$\left\{ \begin{array}{l} (12) X_i \geq 1 - LE_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (13) X_i \leq DX(1 - LE_i) \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (14) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) \leq DX + RE_i - 1 \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (15) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) \geq DX * RE_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (16) Y_i \geq NBC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (17) Y_i \leq DY * NBC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (18) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \leq DY - NTC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (19) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \geq DY(1 - NTC_i) \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (20) LE_i + RE_i + NTC_i + NBC_i \geq 1 \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (21) X_i \geq NLC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (22) X_i \leq DX * NLC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (23) X_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \leq DX - NRC_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (24) X_i + \sum_{k=1}^{NF_i} (F_i^k * DX_i^k) \geq DX(1 - NRC_i) \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (25) Y_i \geq 1 - BE_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (26) Y_i \leq DY(1 - BE_i) \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (27) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \leq DY + TE_i - 1 \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (28) Y_i + \sum_{k=1}^{NF_i} (F_i^k * DY_i^k) \geq DY * TE_i \end{array} \right. \quad (*)$$

$$\left\{ \begin{array}{l} (29) BE_i + TE_i + NRC_i + NLC_i \geq 1 \end{array} \right. \quad (*)$$

(*) pour chaque pièce contrainte

(**) pour chaque paire pièces i et j ($i \neq j$)

(***) pour chaque pièce i

2.2.4. Explication des contraintes

La contrainte (1) indique qu'il faut choisir une seule forme par pièce.

Les autres contraintes sont similaires aux contraintes de la première approche. Par contre, l'écriture de ces contraintes est légèrement différente car les variables DX_i la DY_i ne font plus parti de ce deuxième modèle, elles sont remplacées par $\sum_{k=1}^{NF_i} (F_i^k * DX_i^k)$ et $\sum_{k=1}^{NF_i} (F_i^k * DY_i^k)$, avec NF_i le nombre de formes générés pour la pièce i .

Les contraintes pour gérer l'orientation ne sont plus nécessaires ici car l'orientation est implicitement gérée lors de la génération des formes.

2.2.5. Explication de la fonction objectif

La fonction objectif permet de sélectionner le meilleur ensemble des formes qui maximise la surface utile (minimise la surface perdue).

Une étude expérimentale de ces deux modèles, pour le placement et le dimensionnement des pièces, est proposée dans la section 6, à la fin de ce chapitre.

Les étapes 2 et 3 de la synthèse d'un logement, à savoir le placement des portes et la gestion des couloirs, sont détaillées dans les deux sections suivantes (3 et 4) avant d'explicitier la gestion et la prise en compte du handicap dans le cadre global de notre approche (section 5).

3. Placement des portes

Une fois que les dimensions et les positions sont figées, le placement des portes peut commencer. Deux principaux critères impactent cette opération.

- Le premier est la distance minimale (appelée $MinDistPorteMur$ dans Figure 53) autorisée entre une porte et un mur autre que celui qui contient la porte ; ceci est particulièrement utile lorsque les handicaps sont pris en compte. Il sera détaillé dans la section (5 - Gestion du handicap -).

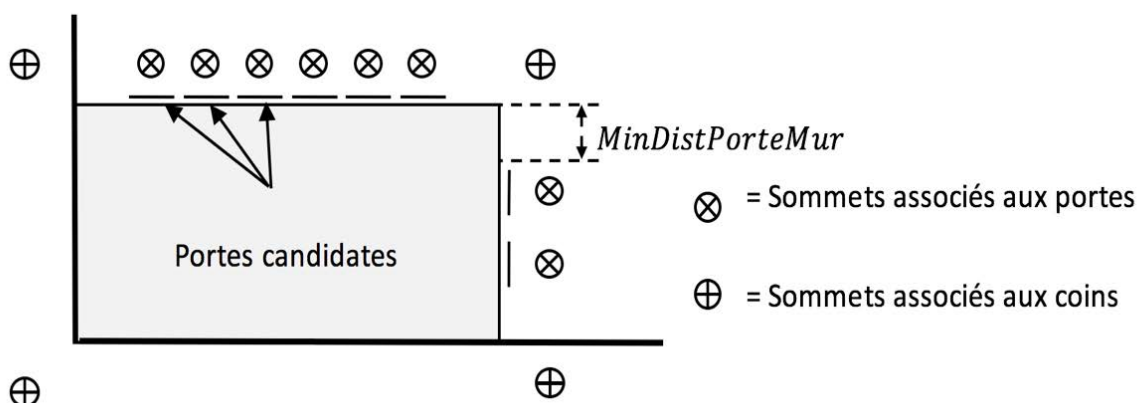


Figure 53. $MinDistPorteMur$ et sommets du graphe

- Le second critère est l'obligation de satisfaire des contraintes sur la longueur ou la valeur des chemins et cycles fréquents (comme le chemin du salon à la cuisine ou le cycle matinal : *chambre à coucher – toilettes – salle de bain – cuisine – chambre à coucher*).

Note : pour simplifier les explications, nous utilisons le terme « chemin » pour désigner un chemin, une chaîne, un circuit ou un cycle habituellement différenciés dans la théorie des graphes.

Optimiser un chemin ne se réduit pas à minimiser sa valeur : dans certains cas, il est souhaitable d'éloigner une pièce d'une ou plusieurs autres pièces. Par exemple, il peut être souhaité de ne pas placer les toilettes à côté du salon. Par conséquent, une contrainte de chemin est l'agrégation de trois informations : le chemin lui-même (la séquence des pièces, notée $(a_i)_{1 \leq i \leq A}$, où A est le nombre de pièces du chemin), la nature de l'optimisation (\leq pour une minimisation et \geq pour une maximisation) et un seuil S , qui est sa valeur minimale tolérée (respectivement maximale) dans le cas d'une maximisation (respectivement minimisation).

L'approche de placement de portes a été conçue avec l'hypothèse selon laquelle une seule porte est nécessaire pour chaque pièce. Pour la plupart des appartements, c'est une condition acceptable. Cela peut éventuellement être gênant pour un grand salon ou une chambre ouvrant à la fois sur un couloir et sur un jardin mais, pour notre première version du logiciel développé, nous avons considéré que c'était suffisant. De plus, cette hypothèse réduit les chances d'avoir un aménagement avec des chambres en enfilade. L'enfilade est souhaitable dans un musée ou un magasin parce qu'elle facilite la circulation d'un grand nombre de personnes mais elle est habituellement mal acceptée dans un logement.

Les quatre principales entrées de la procédure de placement des portes sont l'aménagement provenant de la première étape, la distance *MinDistPortMur*, la largeur des portes, qui est déduite du diamètre de l'occupant ou des normes locales de construction et les contraintes sur les chemins à minimiser et à maximiser. La méthode commence par créer plusieurs portes candidates sur chaque mur qui n'est pas superposé à un mur extérieur. La densité des portes est dictée par le pas de discrétisation, la distance entre les centres des portes consécutives ; les portes candidates peuvent donc se chevaucher. Dans Figure 54, les portes candidates sont marquées en trait rouge pour les rendre plus visibles. Une approximation polygonale du plus court chemin est alors calculée entre toutes les paires de portes candidates. À cette fin, un graphe est calculé. Dans ce graphe, un sommet est créé pour chaque porte candidate (sur un point situé en face du milieu de la porte, à une distance de r - le rayon de l'occupant -). Quatre sommets extrêmes sont aussi créés pour chaque pièce ; ils sont positionnés à l'extérieur des pièces, sur les bissectrices des coins à une distance de $\sqrt{2} * r$. Les arcs du graphe sont tous les segments de droite qui relient deux sommets différents en n'étant jamais à une distance inférieure à r d'une quelconque pièce du logement (le pourtour du logement inclus). Cette contrainte garantit que l'occupant, s'il suit un segment, ne se heurte à aucun mur du logement. La Figure 54 suivante montre un tel graphe. Le plus court chemin entre deux portes est obtenu en calculant, dans le graphe, le plus court chemin entre les sommets associés à ces deux portes et c'est parce que tous les arcs du graphe sont des segments que les plus courts chemins calculés sont des approximations polygonales des plus courts chemins réels. Cette approximation est une majoration : la plupart des composants d'un plus court chemin dans un environnement où les obstacles sont des rectangles sont des segments ; c'est seulement au moment de tourner autour d'un coin de pièce que des arcs de cercles sont nécessaires. En ne calculant pas ces arcs et en les remplaçant par des segments qui joignent le sommet placé en vis à vis du coin à contourner, on surestime donc la longueur du plus court chemin. Cette majoration est cependant très faible et nous avons estimé qu'elle ne faussait pas significativement les valeurs.

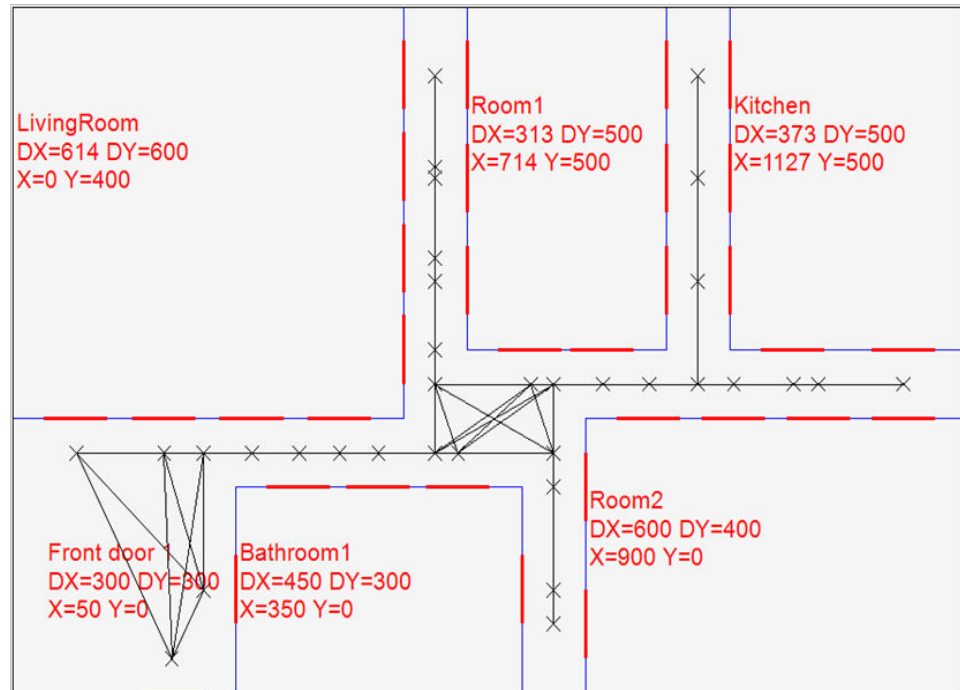


Figure 54. Le graphe utilisé pour les calculs des itinéraires

Une porte est alors choisie pour chaque pièce avec le modèle linéaire suivant. L'idée générale, derrière ce modèle, est de sélectionner parmi toutes les configurations de portes qui satisfont toutes les contraintes d'optimisation de chemins (incluant les contraintes de minimisation et de maximisation), celle où les distances entre les pièces sont minimisées.

Note : le graphe de mobilité construit plus haut permet d'évaluer les valeurs des chemins entre toute paire de portes potentielles. Pour la conception du modèle mathématique pour le choix des portes à installer, le graphe est simplifié en remplaçant chaque chemin par un arc direct portant la valeur du chemin auquel il est substitué.

3.1. Les entrées

N : le nombre de pièces.

M_i : le nombre de portes candidates pour la pièce i .

$f_{i,j}$: la fréquence de déplacement journalière de la pièce i vers la pièce j .

$d_{i,j,k,l}$: la distance entre la porte k , de la pièce i et la porte l de la pièce j .

Notre : Dans un souci de généralité et en prévision de futurs développements, il n'est pas supposé que $d_{i,j,k,l} = d_{j,i,l,k}$. Par exemple, il se peut que le chemin allant de la pièce i à la pièce j contienne une rampe ou des marches et, si l'effort est considéré à la place de la distance pour évaluer la valeur d'un chemin, il est clairement différent dans un sens et dans un autre.

3.2. Les variables

$x_{i,j,k,l}$: 1 si et seulement si le chemin retenu pour aller de la pièce i à la pièce j commence à la porte candidate k de la pièce i et arrive à la porte candidate l de la pièce j ; 0 sinon. Il est pratique d'imaginer $x_{i,j,k,l}$ comme une valeur booléenne indiquant s'il existe (1) ou non (0) un arc entre la porte k de la pièce i et la porte l de la pièce j .

3.3. Le modèle mathématique

Avec ces conventions, le modèle mathématique pour le placement des portes est le suivant :

$$\text{Minimiser } f = \sum_{i=1}^{N-1} \sum_{j=i+1}^N \sum_{k_i=1}^{M_i} \sum_{k_j=1}^{M_j} (x_{ijk_i k_j} \cdot d_{ijk_i k_j} \cdot f_{ij} + x_{jik_j k_i} \cdot d_{jik_j k_i} \cdot f_{ji})$$

S. C

$$\left\{ \begin{array}{ll} (1) x_{ijk_i k_j} = x_{jik_j k_i} & \forall (i, j, k_i, k_j) \in [1, N]^2 \times [1, M_i] \times [1, M_j], i \neq j \\ (2) \sum_{k_i=1}^{M_i} \sum_{k_j=1}^{M_j} x_{ijk_i k_j} = 2 & \forall (i, j) \in [1, N]^2, i \neq j \\ (3) \sum_{k_i=1}^{M_i} x_{ijk_i k_j} = \sum_{k_h=1}^{M_h} x_{hjk_h k_j} & \forall (i, j, h, k_j) \in [1, N]^3 \times [1, M_j], i \neq j \neq h \\ (4) \sum_{i=1}^{A-1} \sum_{k_1=1}^{M_{a_i}} \sum_{k_2=1}^{M_{a_{i+1}}} x_{a_i, a_{i+1}, k_1, k_2} \cdot d_{a_i, a_{i+1}, k_1, k_2} \begin{cases} \leq \\ \geq \end{cases} T, \text{ pour chaque chemin} \end{array} \right.$$

3.4. Explications des contraintes

La contrainte (1) indique que si le chemin allant de la pièce i à la pièce j emprunte la porte k_i et la porte k_j , alors le chemin inverse doit utiliser les mêmes portes sinon on risque, pour chacune des deux pièces, d'utiliser une porte différente pour la quitter et pour y entrer, ce qui veut dire une sélection de plus d'une porte par pièce.

La contrainte (2) exige le choix d'exactly un arc (chemin) pour l'aller et un arc pour le retour pour chaque couple de pièces i et j ($x_{ijk_i k_j} = x_{jik_j k_i} = 2$).

Ces deux contraintes ne sont pas suffisantes pour garantir que chaque pièce a seulement une porte. Par exemple, trois pièces avec deux portes candidates chacune, pourraient être connectées de cette façon : $x_{1,2,1,1} = x_{2,1,1,1} = x_{1,3,2,1} = x_{3,1,1,2} = x_{2,3,2,1} = x_{3,2,1,2} = 2$ (Figure 55). Dans ce cas, il y a bien un unique chemin de chacune des trois pièces vers chaque autre pièce donc (2) est satisfaite ; de plus, pour chaque couple de pièces, le chemin aller et le chemin retour utilisent les mêmes portes donc (1) est satisfaite. Pourtant, deux portes sont utilisées pour la pièce 1 et la pièce 2. C'est la raison pour laquelle la contrainte (3) est nécessaire : i, j et h sont trois pièces différentes, k_j est une porte candidate de la pièce j . Le terme gauche de (3) est le nombre total d'arcs (chemins) depuis toutes les portes de la pièce i vers la porte k_j de la pièce j . D'après la contrainte (2), la valeur de ce terme ne peut être que 0 ou 1 ; si cette valeur est de 0, alors le terme de droite de cette contrainte doit être nul aussi, ce qui interdit à tout chemin de h à j d'emprunter k_j ; en d'autres termes, si le chemin entre i et j n'emprunte pas la porte k_j alors cette porte ne peut être utilisée pour d'autres arcs entre la pièce j et le restant des pièces. Symétriquement, si la valeur du terme gauche est 1, alors il existe un chemin allant de i à j en empruntant la porte k_j et, d'après la contrainte (2), ce chemin est le seul chemin de i à j ; et donc, le second terme doit aussi valoir un, ce qui veut dire qu'il y a un seul et unique chemin de h à j qui utilise la porte k_j aussi. Autrement dit, tous les chemins allant de n'importe quelle pièce à

la pièce j doivent utiliser la même porte pour accéder à j ; ainsi, une seule porte sera choisie pour la pièce j et c'est évidemment vrai pour toutes les autres pièces.

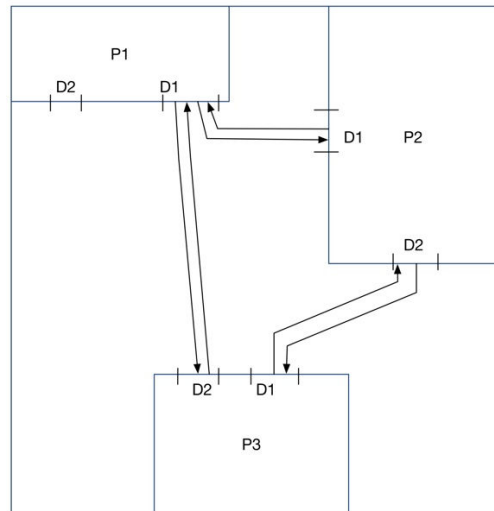


Figure 55. Exemple d'une sélection de portes sans considérer la troisième contrainte

La contrainte (4) permet de traduire les contraintes sur l'optimisation des chemins. Les deux sommations intérieures calculent les distances entre chaque paire de pièces consécutives d'un chemin. La sommation principale est la valeur du chemin.

Note : une modélisation mathématique alternative et peut être plus intuitive aurait été d'utiliser des variables de décisions binaires pour chaque porte j de chaque pièce i : $U_{i,k}$ (1 si la porte k de la pièce i est choisie, 0 sinon). La contrainte pour choisir uniquement une porte par pièce serait évidente mais, d'autre part, la formule pour calculer la distance entre chaque paire de pièces consécutives serait quadratique :

$$\sum_{k_1=1}^{M_{a_i}} \sum_{k_2=1}^{M_{a_{i+1}}} U_{a_i, k_1} \cdot U_{a_{i+1}, k_2} \cdot d_{a_i, a_{i+1}, k_1, k_2}$$

3.5. Explications de la fonction objectif

Comme expliqué au-dessus, la fonction objectif permet de sélectionner l'ensemble de portes qui minimise la somme des distances entre toutes les paires de pièces. Autrement dit, dans la limite où cela est compatible avec les contraintes sur les longueurs des chemins, on réduit le plus possible les itinéraires dans le logement.

4. Suppression des couloirs inutiles

Une fois que les portes sont placées, certains couloirs n'ont plus aucune utilité (voir la zone grise de la Figure 56.a). Il s'agit des régions qui ne sont utilisées par aucun itinéraire entre deux pièces. Il n'y a donc pas d'intérêt à les conserver alors qu'elles peuvent servir à agrandir les pièces adjacentes. Cette section présente l'heuristique développée à cette fin.

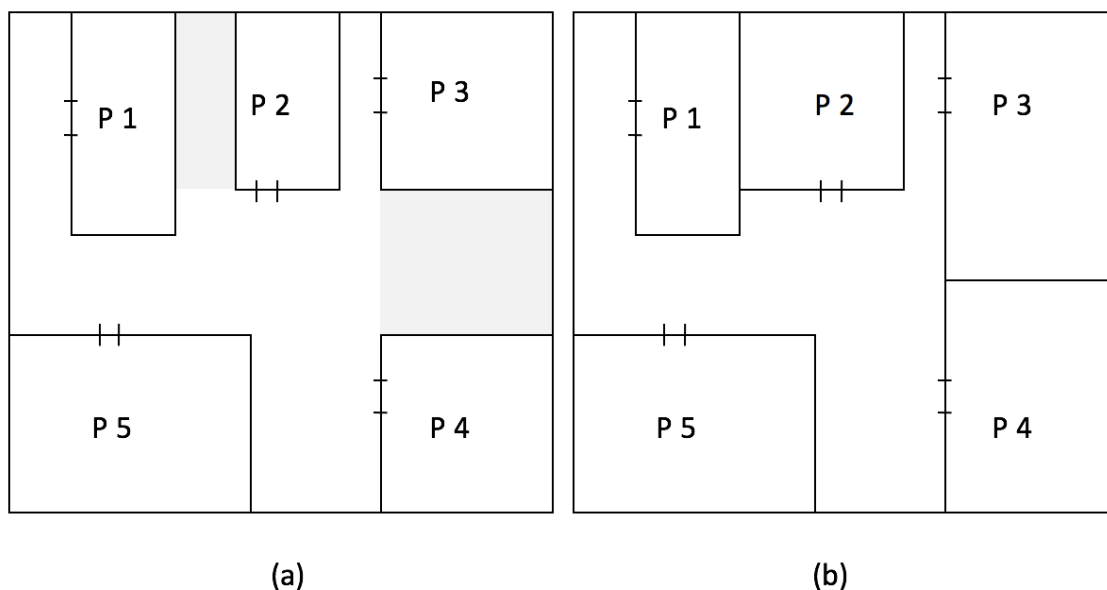


Figure 56. (a) couloirs inutiles en gris, (b) après la suppression des couloirs inutiles

Pour faciliter les explications à venir, les murs sont caractérisés par une lettre parmi $\{H, B, G, D\}$ (pour Haut, Bas, Gauche, Droite) et par le numéro de la pièce. Par exemple, $D3$ est le mur droit de la pièce 3.

La suppression des couloirs inutiles pose quatre questions principales : comment détecter la zone inutile d'un couloir, quel(s) mur(s) faut-il bouger, dans quel ordre et, avec quelle amplitude ? En réalité, répondre à la deuxième question apporte une réponse évidente à la première question, nous nous concentrons donc sur les trois dernières dans les sections qui suivent.

La Figure 56 montre une des suppressions possibles des couloirs ; on a choisi de déplacer $G2$ jusqu'à rencontrer $D1$ et de répartir l'espace vide entre $P3$ et $P4$ à parts égales sur les deux pièces (évidemment une répartition différente de l'espace entre $P3$ et $P4$ est possible).

4.1. Sélection des murs autorisés à se déplacer

Dans les étapes précédentes de ce processus, les pièces ont été placées et dimensionnées, les portes ont été ensuite insérées dans certains murs et cela a été fait en prenant surtout en compte la valeur des itinéraires à optimiser selon les capacités ou la nature du handicap du futur occupant. Le calcul de ces itinéraires a nécessité la création des points de passage en face des coins de chaque pièce (graphe de mobilité). Quand un de ces points est utilisé pour le calcul d'un itinéraire, il n'est évidemment pas judicieux de le déplacer car cela aurait un impact sur la valeur des itinéraires l'utilisant, ce qui remettrait en cause le placement des portes de l'étape précédente. En conséquence, les deux murs qui se rejoignent au niveau d'un point de passage ne sont pas autorisés à se déplacer. Par exemple, dans la Figure 56, $B1$ ne peut être repoussé vers le bas car l'itinéraire de $P1$ vers $P3$ contourne nécessairement $P1$ par le dessous. Pour cela, il utilise les points de passage en vis à vis des deux coins inférieurs de $P1$. Déplacer le mur déplacerait les points de passage et changerait les chemins calculés. De même, un mur qui contient une porte n'est pas autorisé à se déplacer du fait qu'à nouveau les itinéraires seraient compromis. Evidemment, les murs qui sont superposés aux murs extérieurs du logement eux aussi ne peuvent être déplacés.

Tous les autres murs sont susceptibles de bouger.

4.2. Tri des murs à déplacer par priorité décroissante

Rappelons qu'à ce stade, il est certain que les dimensions minimales que l'occupant a formulées sont satisfaites ; en conséquence, les extensions permises par la suppression des couloirs inutiles ne constituent que des bonus.

Dans le cadre d'un outil logiciel d'aide à la décision, on demandera à l'utilisateur de choisir les pièces qui méritent le plus d'être agrandies et de les trier par priorité décroissante. Certaines personnes préféreront avoir de l'aisance dans la cuisine, d'autres dans le salon. Les murs héritent de la priorité de leur pièce et, au sein d'une pièce, ils sont arbitrairement traités dans l'ordre droit, haut, gauche et bas.

4.3. Amplitude de l'éloignement d'un mur

Pour calculer de combien un mur peut bouger, nous listons tous les obstacles qui limitent sa translation dans une direction perpendiculaire à lui-même et nous gardons la distance au plus proche. À titre d'exemple, considérons les obstacles de $H4$ dans la Figure 57: il y a d'abord les murs parallèles à $H4$ dans les autres pièces (ici $B2$ et $B1$, respectivement à des distances $dist1$ et $dist2$) ; il y a ensuite un mur du contour extérieur, à la distance $dist3$; et enfin un segment du chemin de la pièce 2 à la pièce 3, à la distance $dist4$. Dans dernier cas, le calcul de la distance doit prendre en compte le rayon de la personne, ce qui implique en réalité la distance jusqu'au segment en pointillés. Il faut aussi noter que la notion de *distance* utilisée dans ce paragraphe n'est pas la distance au sens géométrique habituel. Pour $dist1$, $dist2$, $dist3$, la nuance est sans conséquence ; par contre, on constate que $dist4$ est strictement supérieure à la distance "classique" de $H4$ au segment en pointillés, cette dernière étant la distance de l'extrémité gauche de $H4$ à son projeté orthogonal sur le segment en pointillés.

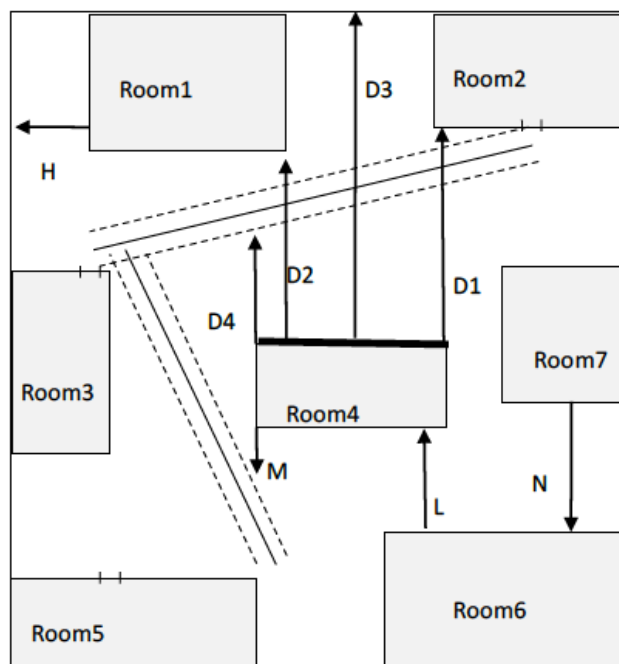


Figure 57. Les quatre obstacles pour $HM4$

Les portes sont aussi des obstacles : *B3* (Figure 57) ne peut être déplacée contre *H5* sinon la porte serait partiellement masquée. Pour éviter cela, un mur artificiel est ajouté en face de chaque porte, comme illustré dans la Figure 58. Ce mur est plus long que la porte, afin de s'assurer que si un mur vient à bouger contre un mur qui contient une porte, la distance minimale entre la porte et le mur reste respectée.

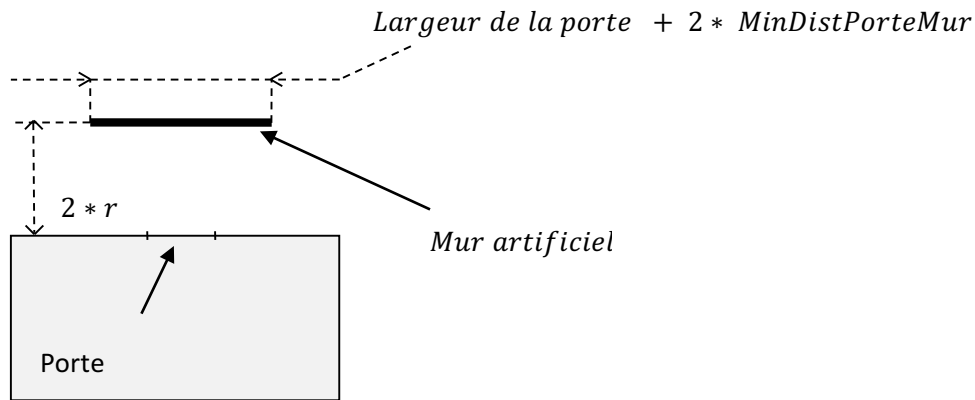


Figure 58. Mur artificiel associé à une porte

Pour un mur donné, une fois les obstacles énumérés et la distance au plus proche calculée, on a la translation maximale possible pour ce segment. Parfois, il est souhaitable de lui appliquer la totalité de ce mouvement ; c'est le cas de *H4*, qui translate de *dist4* (*D4* dans la Figure 57), et de *G1*, qui translate de *H*. Par contre, déplacer *B4* d'une distance de *M* empêche ensuite *H6* de bouger de *L*, ce qui pourrait être préférable, compte tenu du fait que le gain en surface pour la pièce 6 serait nettement supérieur à celui de la pièce 4. À nouveau, c'est seulement l'utilisateur du logiciel qui peut définir ces priorités ; nous recourons donc à nouveau au classement qu'il a fourni pour arbitrer les extensions.

Pour résumer, l'algorithme du sous-programme qui calcule l'éloignement maximal d'un mur est donc :

```

Fonction EloignementMax (m, M1, M2) : réel

// M1 est l'ensemble des quatre murs du logement, des quatre murs
// de chaque pièce et des obstacles factices ajoutés devant les
// portes.

// M2 est l'ensemble des segments de chaque itinéraire entre
// deux pièces.

// m est un mur autorisé à bouger.

Début

    D = +∞

    Pour chaque obstacle Obst de M1 et tant que D>0 faire

        D' = Distance1* de m à Obst

        Si (D'≥0) et (D'<D) alors D = D' FSi

    FPour

    Pour chaque obstacle Obst de M2 et tant que D>0

        D' = Distance2** (m, Obst)

        Si (D'≥0) et (D'<D) alors D = D' FSi

    FPour

    DistMax (m) = D

Fin

```

Algorithme 2 Fonction d'éloignement maximal d'un mur

(*) la Distance1 est calculée perpendiculairement au mur à éloigner avec la convention qu'elle doit être négative (et négligée) si l'obstacle n'entrave pas la translation du mur vers l'extérieur de la pièce. Les aspects calculatoires sont détaillés dans le Chapitre II.18 (Distance d'un segment à un obstacle).

(**) idem à (*) mais la Distance2 de *m* à *o* doit tenir compte de l'écart à maintenir avec l'élément de trajectoire *o*. C'est donc la *distance1* de *m* au plus proche des deux segments parallèles et à une distance *r* de *o*.

4.4. Algorithme de suppression des couloirs inutiles

L'algorithme de suppression commence par trier les pièces par ordre de priorité décroissante ; les pièces sont ensuite parcourues dans cet ordre et tous les murs autorisés sont translatés autant que possible. Les distances sont réévaluées après chaque déplacement d'un mur. La Figure 59 montre

comment la disposition précédente évolue si les murs d'une pièce sont déplacés dans l'ordre trigonométrique, en commençant, par défaut, par le mur de droite, si l'ordre de priorité est simplement le numéro des pièces et s'il n'y a pas d'autres chemins que les deux dessinés (ce qui n'est évidemment pas réaliste).

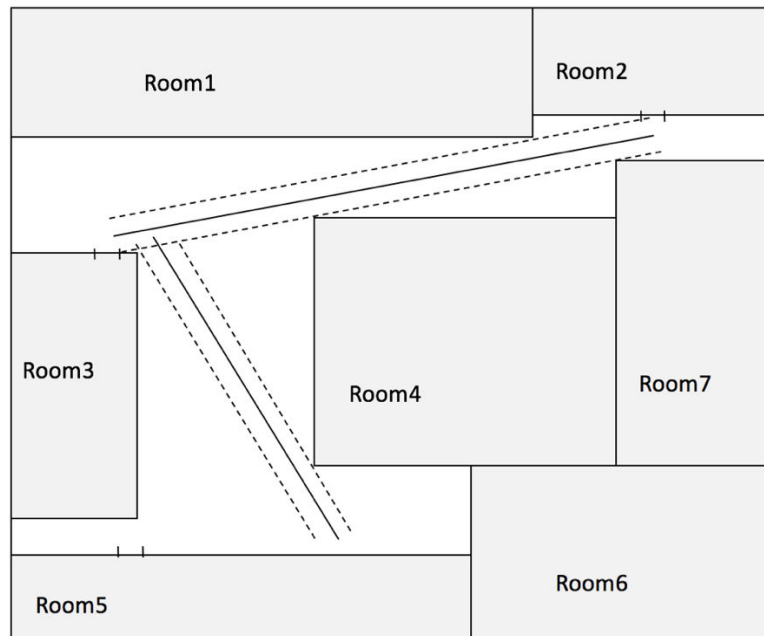


Figure 59. Une redistribution possible des couloirs inutiles

Soit $M1$ l'ensemble des quatre murs du logement, des quatre murs de chaque pièce et des obstacles factices ajoutés devant les portes.

Soit $M2$ l'ensemble des segments de chaque itinéraire entre deux pièces.

Soit M la séquence des murs autorisés à bouger, ordonnée par priorité décroissante et dans l'ordre $D - H - G - B$ pour une pièce donnée.

L'algorithme de suppression des couloirs inutiles est donc :

```

Procédure SuppressionCouloirs (Logement)

Début

  pour chaque mur  $m$  de  $M$ 

     $D = \text{EloignementMax}(m, M1, M2)$ 

    Eloigner  $m$  de  $D$ 

  FPour

Fin

```

Algorithme 3 Procédure de suppression des couloirs

Le résultat est, en général, dépendant de l'ordre de priorité des pièces mais aussi de l'ordre dans lequel les murs sont éloignés pour une pièce donnée. L'impact de ce dernier ordre est cependant moins important que ce que suggère l'exemple précédent. Dans une situation plus réaliste, l'aire des pièces étant à maximiser (ce qui aurait placé $G1$ sur le mur extérieur gauche), les espaces vides insérés entre les pièces pour servir de couloirs sont limités au diamètre de l'occupant et la plupart d'entre eux sont utilisés par les chemins ce qui réduit considérablement le nombre d'élargissements possible.

On doit remarquer aussi que, après le retrait des couloirs inutiles, toute pièce reste accessible depuis n'importe quelle autre pièce, car les chemins ne sont pas affectés par les élargissements ; Mais, par contre, un couloir peut être inaccessible : la Figure 60 illustre ce cas de figure : $D1$ et $H1$ ne sont pas autorisés à bouger mais $G1$ l'est : $R1$ s'étend à gauche jusqu'à $R4$; $B1$ descend ensuite jusque $R2$; $G2$ translate ensuite jusque $R3$, $H3$ jusque $R4$ et on aboutit à la configuration de droite.

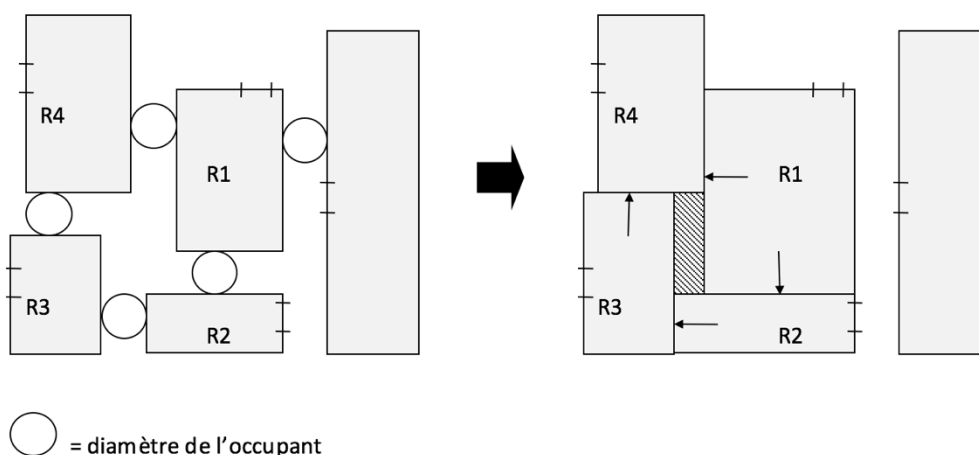


Figure 60. Un couloir isolé (zone rayée) qui apparaît après l'élargissement des pièces $R1..R4$

L'heuristique présentée plus haut ne permet pas d'éviter que ce cas de figure se produise, sauf si on refuse d'élargir une pièce autant que possible. Dans l'exemple ci-dessus, arrêter le déplacement de $G2$ plus tôt, à l'aplomb de $D4$, aurait permis à $D3$ d'atteindre $G1$ et l'ensemble des couloirs inutiles auraient été affectés aux pièces adjacentes.

5. Gestion du handicap

5.1. Prise en compte du handicap

Nous souhaitons souligner que la façon dont les contraintes sont conçues et leur variété permet de produire un aménagement adapté à une personne handicapée ou vieillissante. Cette section donne quelques indications sur la manière dont ces aspects ont été appréhendés et pris en compte par la modélisation mathématique.

Un cas de déficience couramment abordé est l'utilisation d'un fauteuil roulant. Pour une personne en fauteuil roulant, se déplacer de manière confortable dans le logement implique la satisfaction de plusieurs conditions :

1. les dimensions des pièces (incluant les toilettes) doivent être supérieures à un seuil donné ; cela se fait aisément en définissant correctement $MinDim1_i$, $MinDim2_i$ pour chaque pièce i ;

2. les couloirs doivent permettre à la personne de tourner sur elle-même (par exemple, la réglementation en France impose un couloir avec une largeur de 1.5 m) ; cela est fait en mettant le rayon du cercle approchant l'occupant à 0.75 m ;
3. 3) la personne en fauteuil roulant, en particulier une personne qui ne peut pas se pencher en avant par ce qu'elle est sanglée au fauteuil roulant doit pouvoir atteindre la poignée de la porte avant que ses pieds ne touchent le mur perpendiculaire à la paroi de la porte (Figure 61) ; Ceci est fait en donnant à *MinDistPorteMur* une valeur adéquate ;
4. 4) toute porte doit être assez large pour permettre le passage du fauteuil roulant ; pour cela, les portes candidates doivent être au moins aussi larges que le fauteuil roulant.

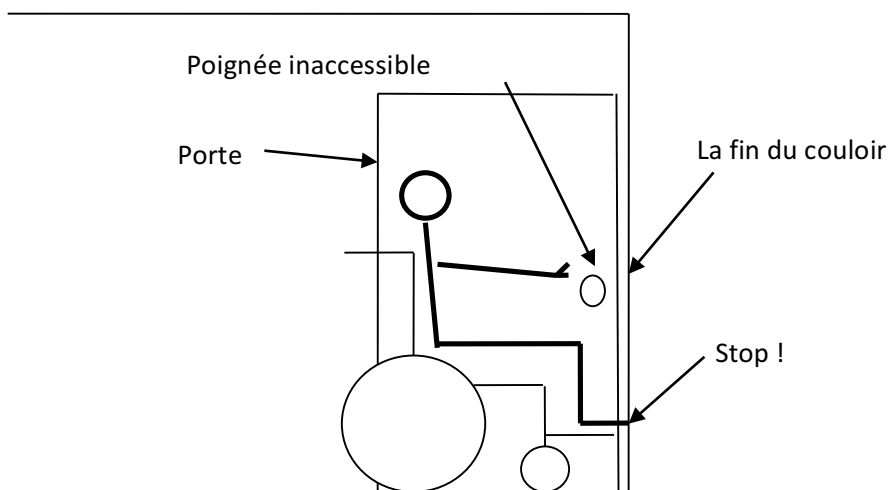


Figure 61. Une poignée de porte, inaccessible parce que *MinDistPorteMur* est trop petit

Dans des logements pour personnes sourdes et malentendantes, il est parfois préférable que le séjour ne soit pas adjacent aux murs mitoyens, car les occupants ont tendance à mettre le son de la télé, par exemple, très fort. Une manière de considérer cette contrainte est d'utiliser une suite de *SContraintes* de type *Eloignement* pour déplacer le séjour loin des murs mitoyens ; par exemple, pour éviter que le séjour ne soit contre le mur extérieur droit, on fixera $MaxX_{Séjour}$ à $Largeur_{Séjour} - k$. La valeur de k doit être choisie suffisamment grande pour éviter de générer des zones inutiles. C'est pourquoi nous fixons k au diamètre du cercle qui approche l'occupant ($2 \cdot r$), soit la largeur minimale requise pour un couloir.

Les personnes qui sont très sensibles à la lumière peuvent vouloir placer les pièces de sorte qu'elles soient les moins exposées à la lumière du jour. Pour cela, elles peuvent demander de ne pas avoir la lumière du matin en cuisine en imposant une *SContrainte* de type « Pas sur le mur « Est » et ne pas avoir la lumière de l'après-midi ou du soir dans le séjour et la chambre à coucher. Une approche symétrique peut être appliquée en cas de faible perception de la lumière.

Certains ergothérapeutes suggèrent également, en particulier pour une personne vieillissante, qu'une manière de ralentir la perte d'autonomie est de s'assurer que la personne parcourt au moins une distance donnée par jour. L'aménagement d'un logement peut donc contribuer à atteindre cet objectif : par l'identification des chemins habituels et leurs fréquences, et en soumettant ces chemins à une contrainte de maximisation, on peut agir sur la position des portes et la distance parcourue quotidiennement ; La Figure 62.a montre un aménagement sans contraintes de chemins ; la longueur du chemin ($R0 - R1 - R2 - R3 - R0$) est 288 cm. Dans la Figure 62.b une contrainte de minimisation a été appliquée au même chemin ; la longueur correspondante est de 151 cm ; dans la Figure 62.c, une maximisation a été requise et la longueur devient 601 cm. Dans le dernier cas, la

longueur du chemin est multipliée par 4, ce qui montre clairement qu'on peut augmenter de manière significative la distance parcourue quotidiennement par l'habitant en agissant sur les positions des portes.

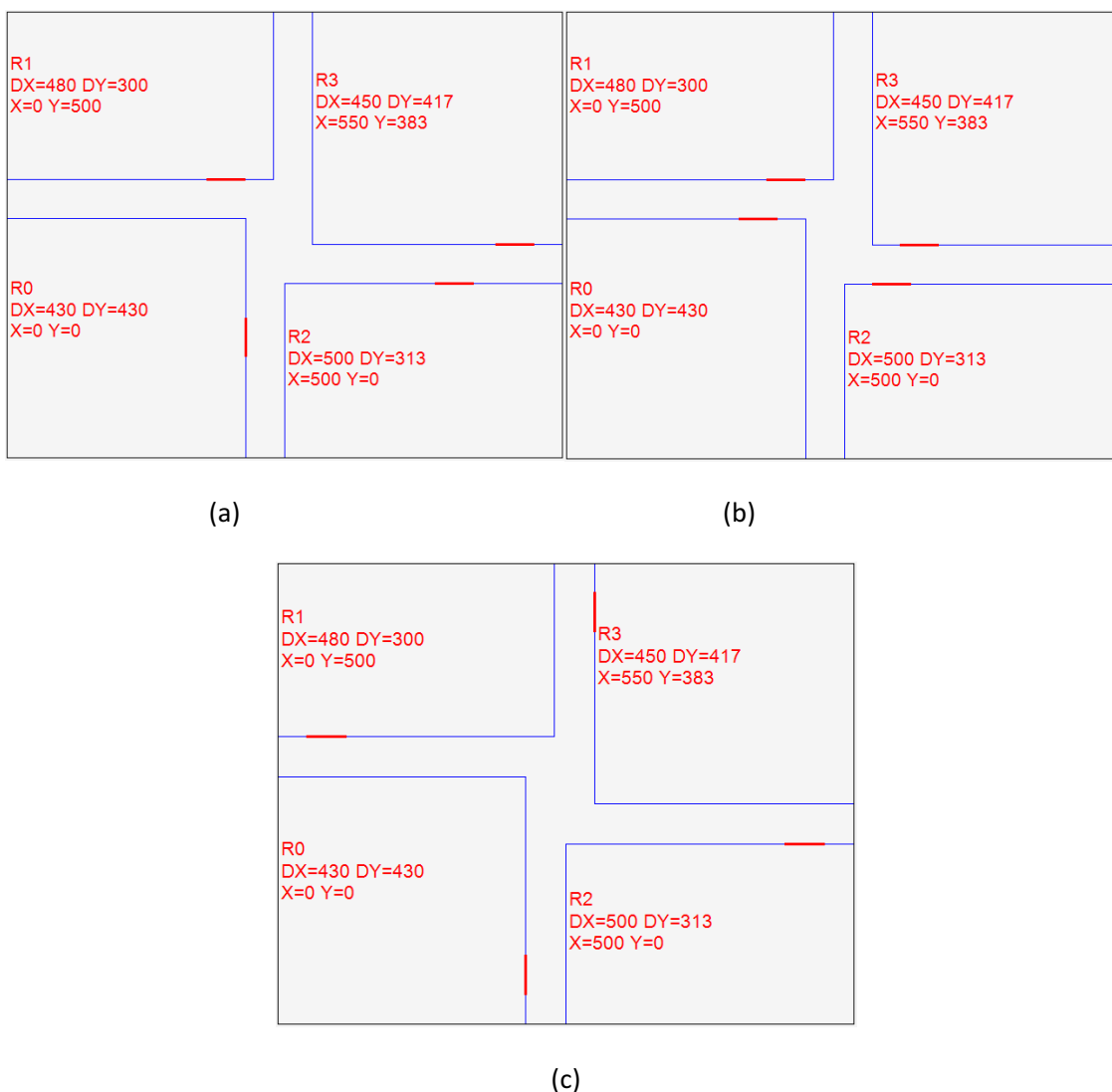


Figure 62. Impact des contraintes d'optimisation

5.2. Illustration

Afin d'illustrer la démarche globale et l'impact des besoins ou des souhaits de l'occupant sur l'agencement, nous avons élaboré trois scénarios. Le premier, scénario de référence, avec une personne ne présentant aucun handicap. Le deuxième scénario intègre des contraintes induites par les conditions de mobilité du futur occupant (personne en fauteuil roulant). Le troisième scénario prend en compte des contraintes imposant un minimum d'exercice en rallongeant les itinéraires (personne atteinte de la maladie de parkinson).

Description de l'instance et des données en entrée

On considère un logement avec 8 pièces et 2 entrées.

Type de logement	Dimension 1	Dimension 2	Surface
8 pièces	18 m	15 m	270 m ²

Tableau 4 Caractéristiques du logement

Pour chaque pièce, on note :

- D1 Min, D1 Max, D2 Min D2 Max (en cm) : les encadrements des deux dimensions ;
- X Deb Min, X Deb Max, X Fin Min, X Fin Max, Y Deb Min, Y Deb Max, Y Fin Min, Y Fin Max (en cm) : les bornes inférieures et supérieures sur chaque axe.

Contraintes de placement et de dimensionnement des pièces

Pièce	D1 Min	D1 Max	D2 Min	D2 Max	X Deb Min	X Deb Max	X Fin Min	X Fin Max	Y Deb Min	Y Deb Max	Y Fin Min	Y Fin Max
Chambre 1	200	500	200	500	0	1600	200	1800	0	1300	200	1500
Chambre 2	200	500	200	500	0	1600	200	1800	0	1300	200	1500
Chambre 3	300	500	300	500	0	1500	200	1800	0	1200	300	1500
Chambre 4	200	500	200	500	0	1600	200	1800	0	1300	200	1500
Salon	200	400	200	400	0	1600	200	1800	0	1300	200	1500
Cuisine	200	400	200	400	0	1300	200	1700	400	1000	600	1500
Salle de Bain	200	300	200	300	800	1500	200	1800	0	1300	200	1500
Toilette	200	250	200	250	1000	1600	200	1800	0	1300	200	1550

Tableau 5 Les dimensions minimales et maximales et des zones de placement des pièces

Contraintes de placement et de dimensionnement des entrées

Les entrées sont délimitées par des murs virtuels représentés par des traits pointillés. Pour chaque entrée, on note :

- Mur : le mur sur lequel l'entrée est placée ;
- Position de l'axe (en cm) : la position de l'axe de la porte d'entrée ;
- Largeur de la porte (en cm) : la largeur de la porte d'entrée ;
- D1 Min, D1 Max, D2 Min D2 Max (en cm) : les encadrements des deux dimensions.

Entrée	Mur	Position de l'axe	Largeur de la porte	D1 Min	D1 Max	D2 Min	D2 Max
entrée 1	Gauche	200	100	200	300	200	300
entrée 2	Droit	200	100	200	250	200	250

Tableau 6 Les paramètres des entrées

Scénario n°1 : Personne sans handicap (cas de référence)

Rayon de l'habitant : 35 cm

Contraintes de chemin : Aucune contrainte de chemin n'a été imposée.

Résultat de la phase 1 : Placement et dimensionnement des Pièces

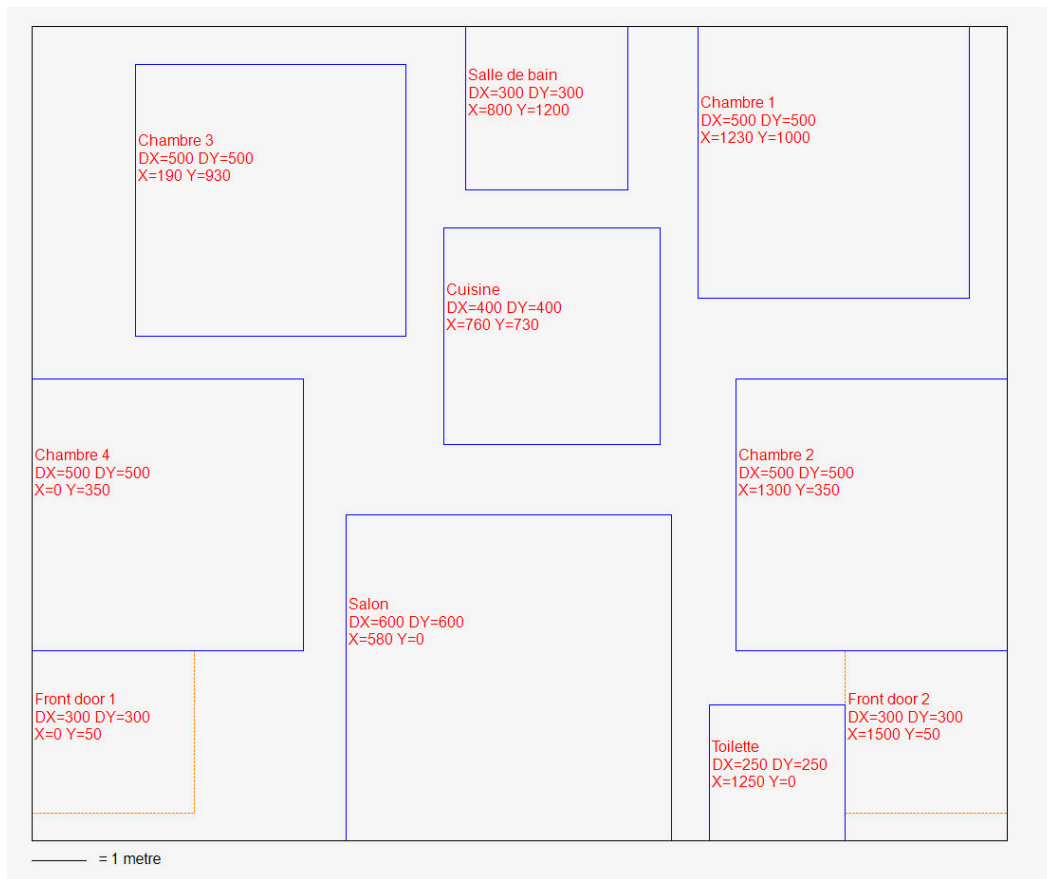


Figure 63. Solution de la première étape

Temps de calcul	Surface utile	Ratio Surface utile / Surface globale	Surface des couloirs
0.34 sec	185.25 m ²	68.6 %	84.75 m ²

Tableau 7 Résultats de la phase 1

Remarque : la chambre 3 n'est pas collée au contour du logement car elle a atteint les dimensions maximales.

Résultat de la phase 2 : Placement des portes

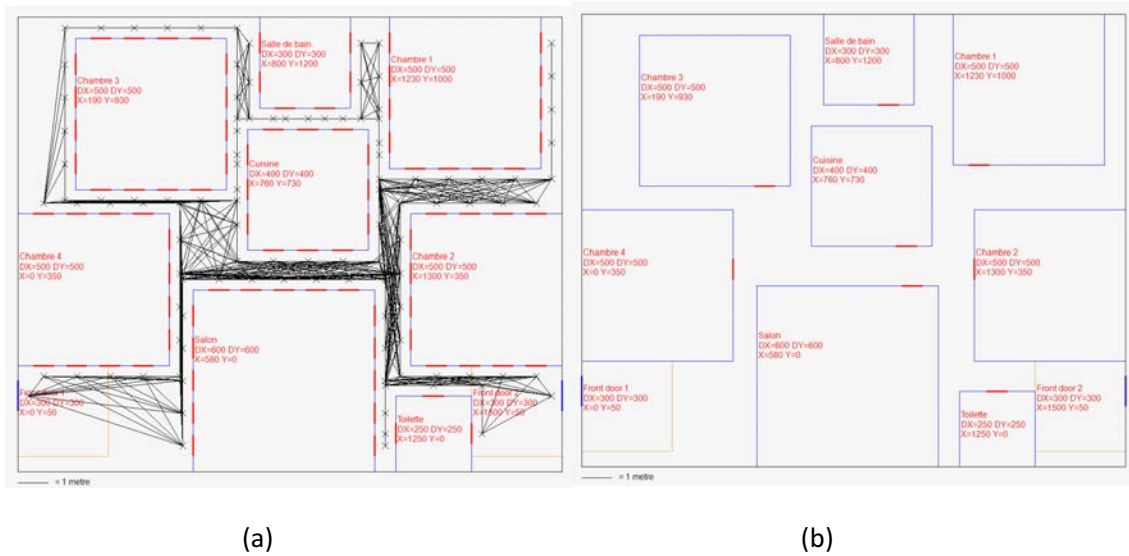


Figure 64. (a) Plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues

Temps nécessaire à la construction du graphe	Temps de résolution du modèle de chemins	Temps de calcul de la phase 2	Temps de calcul cumulé phases 1 et 2
1.27 sec	0.7 sec	1.97 sec	2.31 sec

Tableau 8 Résultats de la phase 2

Résultat de la phase 3 : Plan final après la suppression des couloirs inutiles

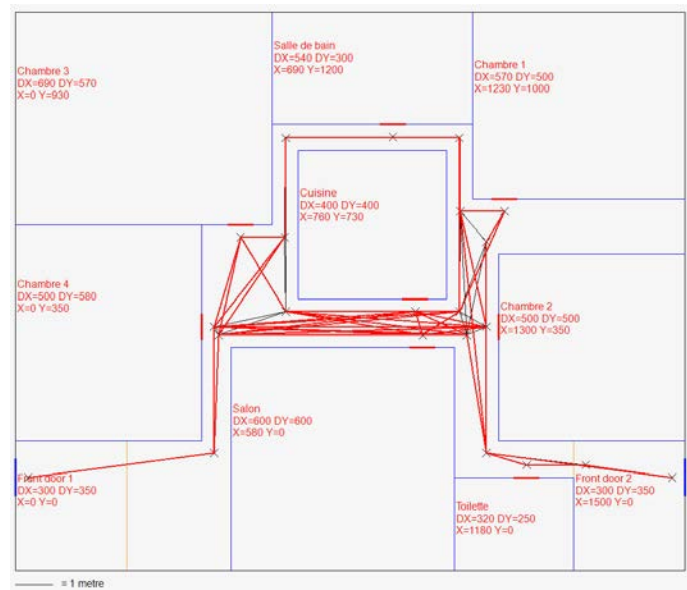


Figure 65. Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera

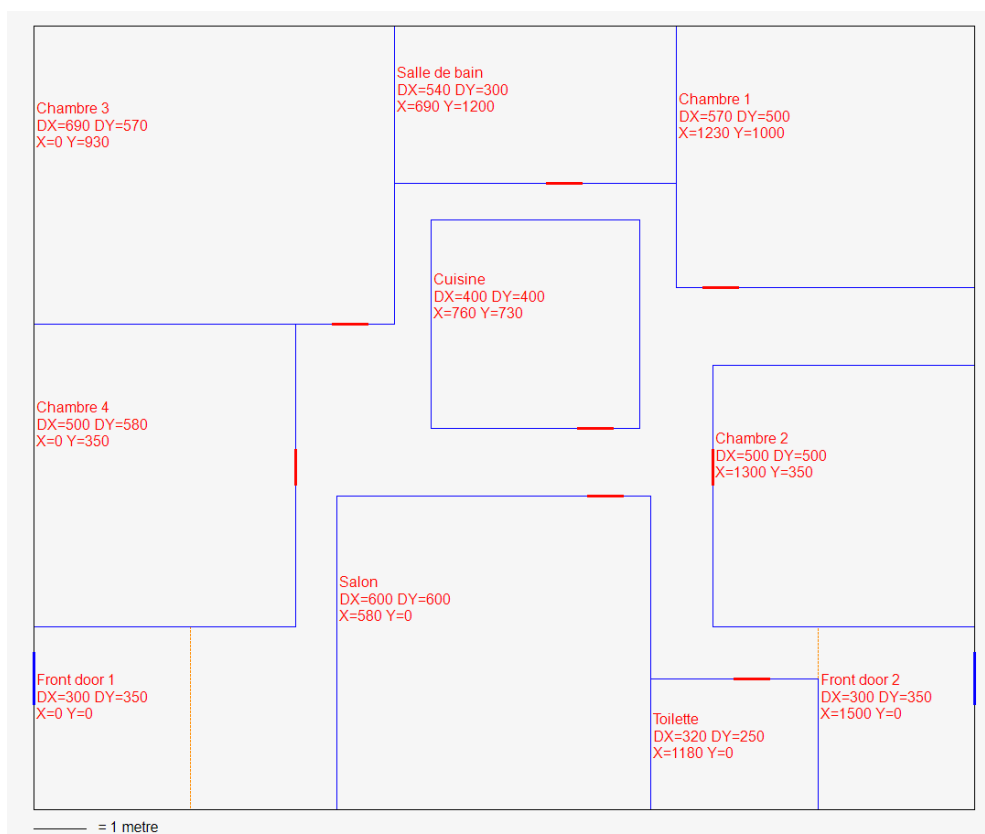


Figure 66. Solution finale

Temps de calcul de la <u>phase 3</u>	Temps de calcul cumulé <u>phases 1, 2 et 3</u>	Surface utile	Ratio Surface utile / Surface globale	Surface des couloirs
0.06 sec	2.37 sec	217.53 m ²	80.6 %	52.47 m ²

Tableau 9 Résultats des trois phases

Scénario n°2 : Personne en fauteuil roulant

Nous reprenons la même instance à une différence, le rayon de l'habitant est fixé à 75 cm au lieu de 35 cm pour tenir compte de l'encombrement du fauteuil roulant.

Rayon de l'habitant : 75 cm

Contraintes de chemin : Aucune contrainte de chemin n'a été imposée.

Résultat de la phase 1 : Placement et dimensionnement des Pièces

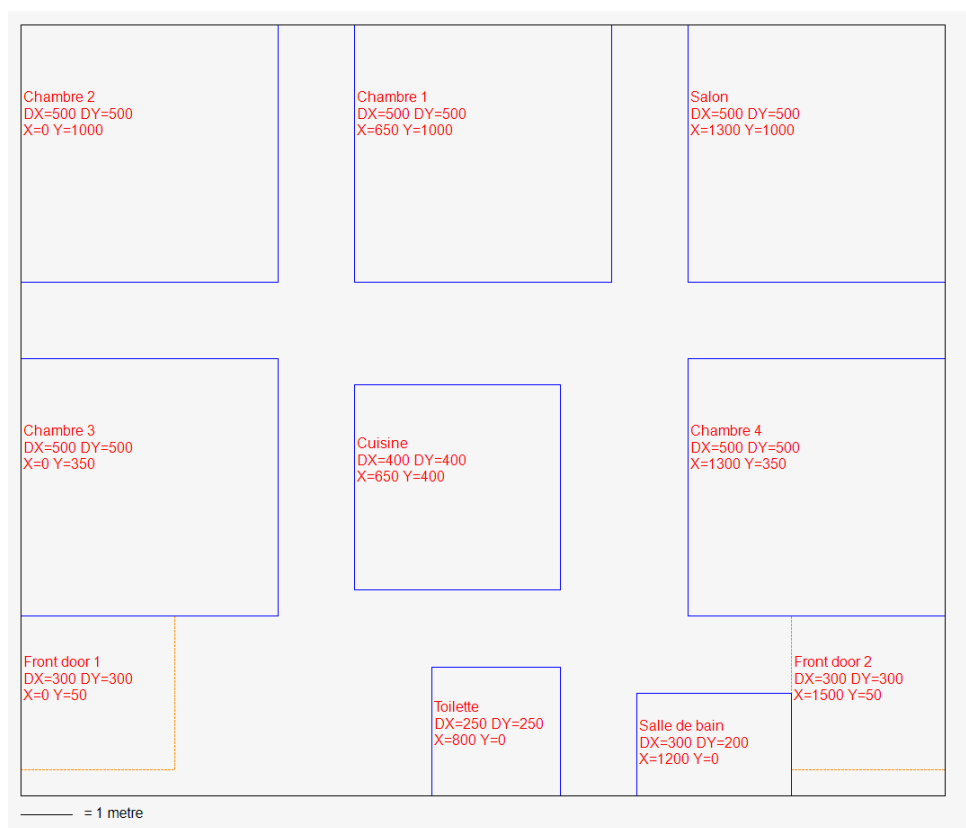


Figure 67. *Solution de la première étape*

Temps de calcul	Surface utile	Ratio	Surface des couloirs
		Surface utile / Surface globale	
38.5 sec	161.25 m ²	59.7 %	108.75 m ²

Tableau 10 Résultats de la phase 1

Résultat de la phase 2 : Placement des portes

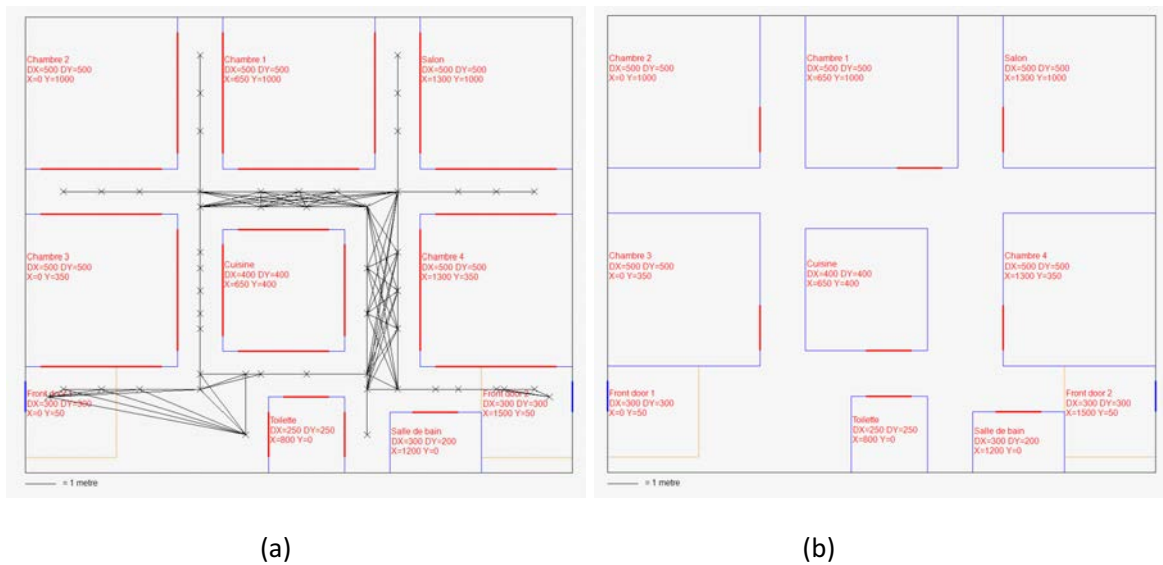


Figure 68. (a) Plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues

Temps nécessaire à la construction du graphe	Temps de résolution du modèle des chemins	Temps de calcul de la phase 2	Temps de calcul cumulé phases 1 et 2
0.78 sec	0.47 sec	1.25 sec	39.75 sec

Tableau 11 Résultats de la phase 2

Résultat de la phase 3 : Plan final après la suppression des couloirs inutiles

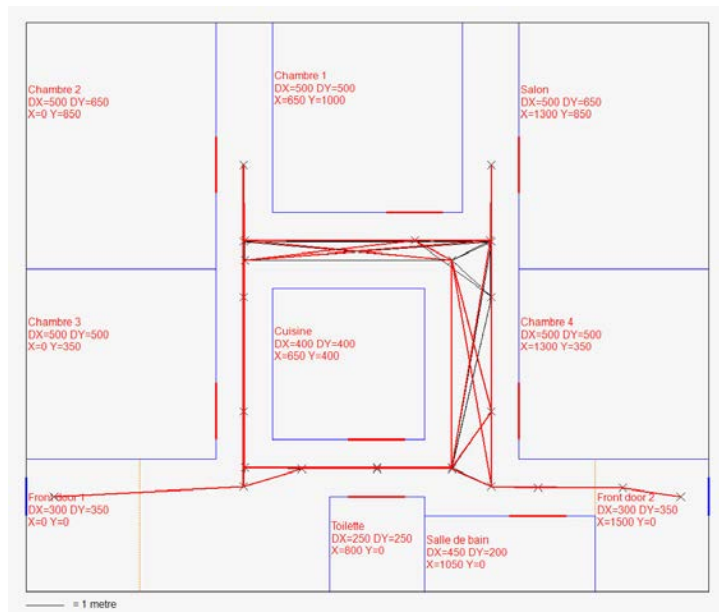


Figure 69. Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera

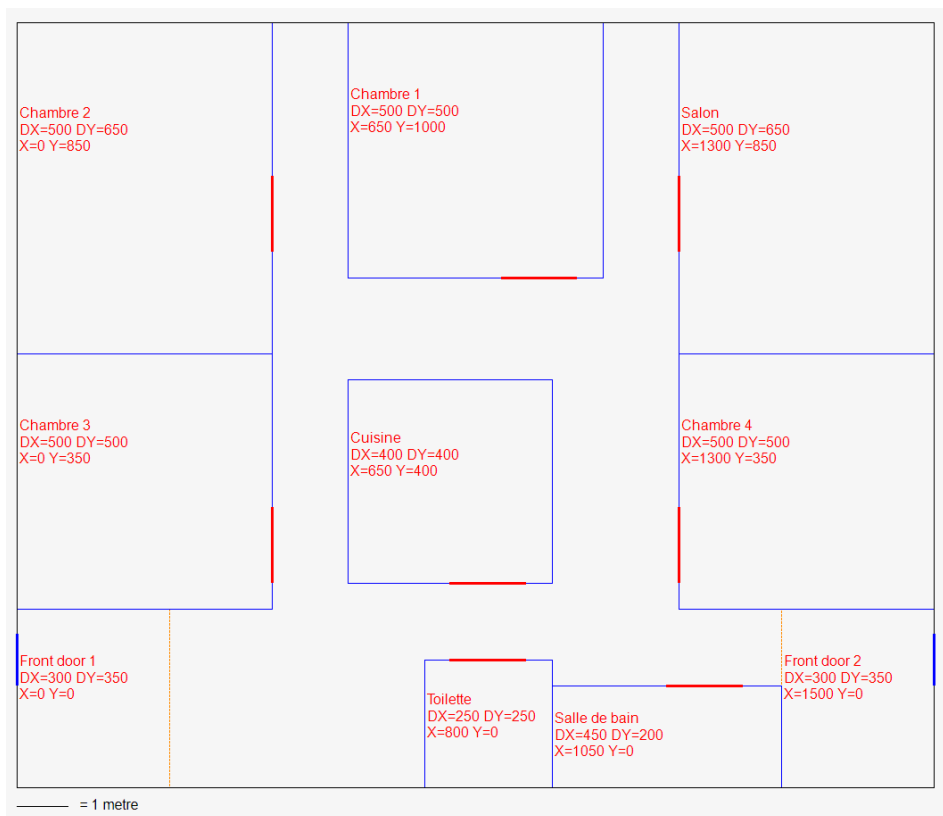


Figure 70. Solution finale

Temps de calcul de la phase 3	Temps de calcul cumulé Phases 1, 2 et 3	Surface utile	Ratio Surface utile / Surface globale	Surface des couloirs
0.15 sec	39.9 sec.	176.25 m ²	65.3 %	93.75 m ²

Tableau 12 Résultats des trois phases

On remarque que la solution générée est différente de celle générée pour une personne sans handicap. En effet, les couloirs sont plus larges pour permettre de faire des demi-tours sur place ; ce qui a pour conséquence de réduire les surfaces allouées aux pièces.

Scénario n°3 : Personne atteinte de la maladie du parkinson

Une personne atteinte de la maladie du parkinson a besoin de maintenir un niveau d'effort acceptable pour ralentir la perte musculaire. Ce besoin se traduit par le fait qu'on cherche à allonger les itinéraires les plus fréquents.

Rayon de l'habitant : 35 cm

Contraintes de chemins : Deux cycles sont concernés.

Le chemin 1 : Chambre 1-Salle de bain – Toilette –Cuisine – Salon \geq 50 m

Le chemin 2 : Salon-Cuisine -Salle de bain - Salon \geq 35 m

Résultat de la phase 1 : Placement et dimensionnement des Pièces

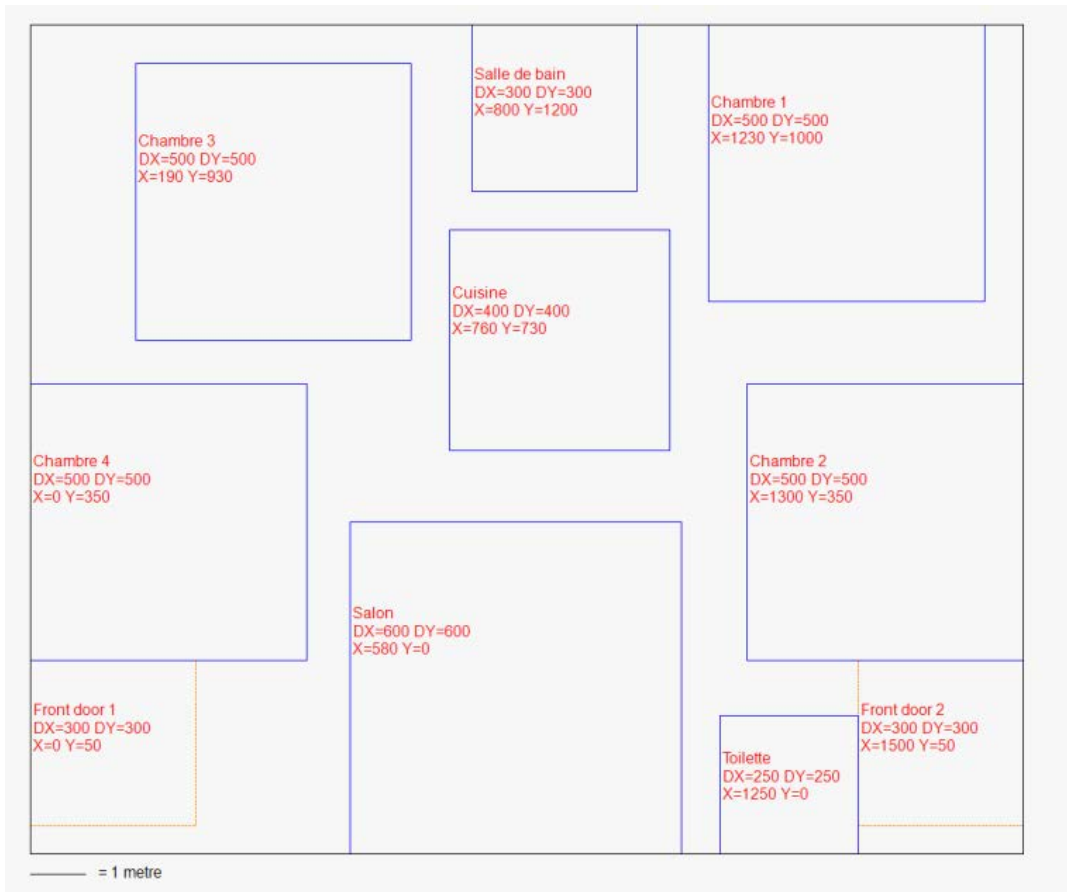


Figure 71. Solution de la première étape

Temps de calcul	Surface utile	Ratio	Surface des couloirs
		Surface utile / Surface globale	
0.34 sec	185.25 m ²	68.6 %	84.75 m ²

Tableau 13 Résultats de la phase 1

Remarque : on retrouve la même solution générée dans la première étape du cas de référence (cas n°1) car les instances de cette première partie sont les mêmes.

Résultat de la phase 2 : Placement des portes

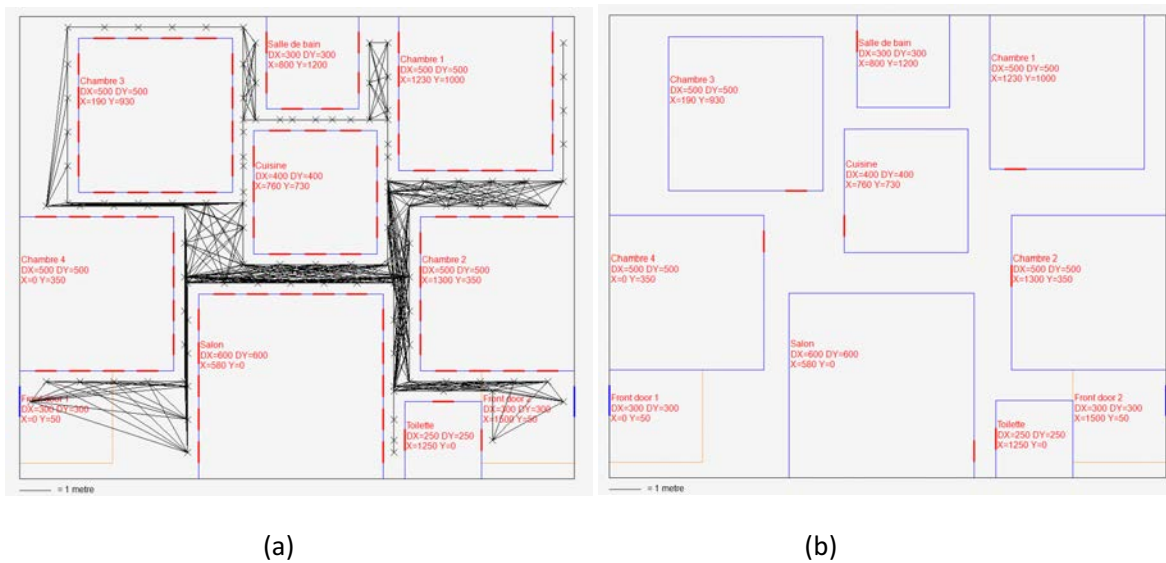


Figure 72. (a) plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues

Temps nécessaire à la construction du graphe	Temps de résolution du modèle des chemins	Temps de calcul de la phase 2	Temps de calcul cumulé phases 1 et 2
1.55 sec	1.45 sec	3.5 sec	3.85 sec

Tableau 14 Résultats de la phase 2

On remarque que la position des portes est différente du cas de référence (cas n°1). Les portes sont plus éloignées les unes par rapport aux autres.

Résultat de la phase 3 : Plan final après la suppression des couloirs inutiles

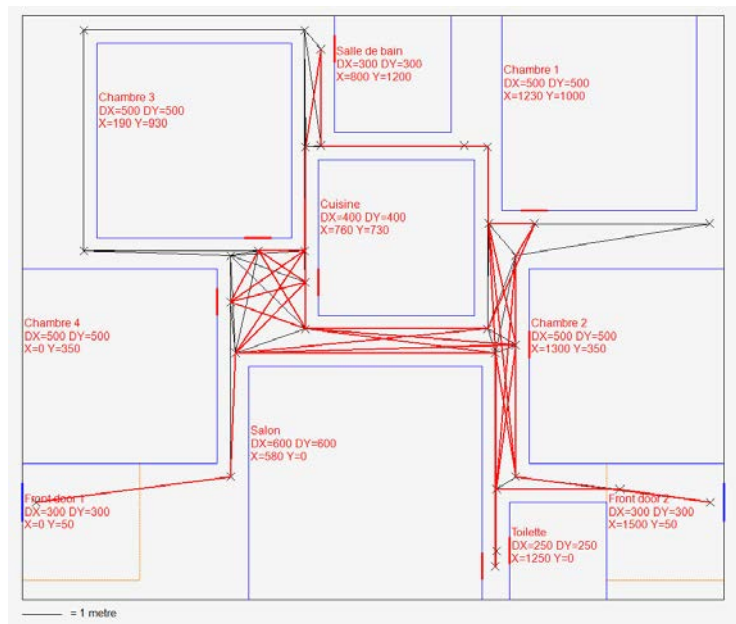


Figure 73. Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera

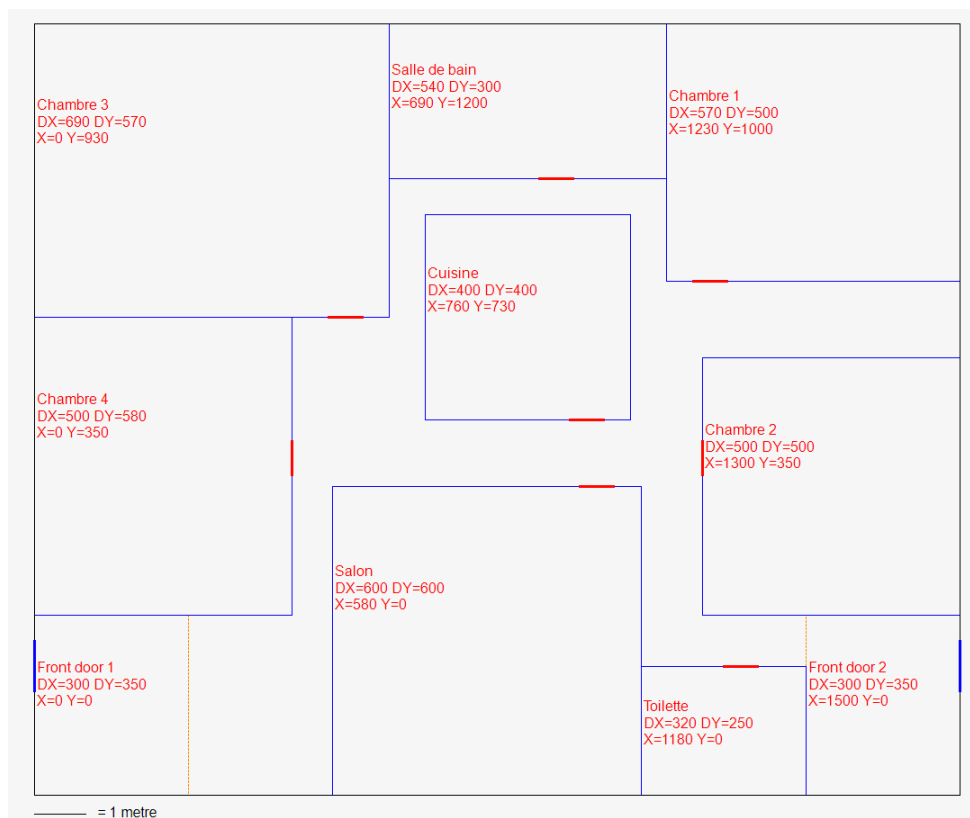


Figure 74. Solution finale

Temps de calcul de la phase 3	Temps de calcul cumulé Phases 1, 2 et 3	Surface utile	Ratio Surface utile / Surface globale	Surface des couloirs
0.08 sec	3.93 sec	213.98 m ²	79.3 %	56.02 m ²

Tableau 15 Résultats des trois phases

En comparant la solution proposée dans le cas n° 1 et la solution proposée ici, on remarque que :

- dans le plan final du cas de référence, la chambre 3 et la salle de bain ont un mur en commun, pareil pour le salon et les toilettes. Dans ce plan final, il existe un couloir qui sépare chacune de ces paires de pièces ;
- la surface utile de la solution de référence est plus grande que la surface utile de la présente solution ; il y a une différence de 3.55m². Cette différence s'explique par les deux couloirs ajoutés entre les pièces mentionnées dans le point précédent.

5.3. Tableau récapitulatif des résultats des trois scénarios

Scénario	Placement et dimensionnement des pièces			Placement des portes			Suppression des couloirs inutiles			Temps global de résolution
	CPU Phase 1	Surface utile	Ratio Surface utile / Surface globale	CPU Construction Graphe	CPU Résolution	CPU phase 1	CPU Phase 3	Surface utile	Ratio Surface utile / Surface globale	
Scénario N°1	0.34s	185.25m ²	68.6%	1.27s	0.7s	1.97s	0.06s	217.53m ²	80.6%	2.37s
Scénario N°2	38.5s	161.25m ²	59.7%	0.78s	0.47s	1.25s	0.15s	176.25m ²	65.3%	39.9s
Scénario N°3	0.34s	185.25m ²	68.6%	1.55s	1.45s	3.5s	0.08s	213.98m ²	79.3%	3.93s

Tableau 16 Tableau récapitulatif

On constate que la prise en compte du handicap a une influence sur la structure globale de la solution calculée et sur le temps de calcul nécessaire pour l'obtenir.

6. Résultats expérimentaux pour les deux modèles de placement et dimensionnement des pièces

Nous proposons dans cette section une étude comparative des deux modèles mathématiques que nous avons développé pour le placement et le dimensionnement des pièces.

Nous allons commencer par parler du générateur des instances, ensuite, nous détaillerons le protocole que nous avons suivi pour les expérimentations. Nous donnons les résultats de chaque modèle, séparément et nous terminons par faire une comparaison des deux modèles.

6.1. Présentation générale

Générateur des instances

Nous avons distingué 8 types de logements (T2-T9). Pour chaque type, nous avons ajouté une salle de bain et une cuisine. Ainsi, un logement de type de T2 est un logement qui est composé de deux pièces principales, en plus d'une cuisine et une salle de bain.

Les surfaces considérées pour chaque type sont comme suit :

Type de Logement	Surface
T2	$40 m^2 \leq Surface \leq 60 m^2$
T3	$60 m^2 \leq Surface \leq 80 m^2$
T4	$80 m^2 \leq Surface \leq 100 m^2$
T5	$100 m^2 \leq Surface \leq 120 m^2$
T6	$120 m^2 \leq Surface \leq 140 m^2$
T7	$140 m^2 \leq Surface \leq 160 m^2$
T8	$160 m^2 \leq Surface \leq 180 m^2$
T9	$180 m^2 \leq Surface \leq 200 m^2$

Tableau 17 Intervalles des surfaces par type de logement

Une instance est définie par :

- le type de logement,
- les dimensions et la surface du logement,
- pour chaque pièce :
 - les bornes inférieures et supérieures de chacune des deux dimensions,
 - les bornes inférieures et supérieures, sur chaque axe, pour délimiter les zones de positionnement.

Nous avons considéré 6 surfaces par type de logement (un pas de $5m^2$ entre deux surfaces consécutives). Les surfaces par type sont comme suit :

Type de Logement	Surfaces considérées
T2	40, 45, 50, 55, 60 m^2
T3	60, 65, 70, 75, 80 m^2
T4	80, 85, 90, 95, 100 m^2
T5	100, 105, 110, 115, 120 m^2
T6	120, 125, 130, 135, 140 m^2
T7	140, 145, 150, 155, 160 m^2
T8	160, 165, 170, 175, 180 m^2
T9	180, 185, 190, 195, 200 m^2

Tableau 18 *Tableau des surfaces*

Pour chaque type de logement et pour chaque surface, nous avons généré 10 instances. Ce qui revient, en considérant tous les types de logements et toutes les surfaces associées, 400 instances à tester.

Protocole

Pour chaque modèle, chaque instance a été testée, dans un premier temps, avec un rayon de 35 cm et, dans un second temps, avec un rayon de 75 cm. Ensuite, et pour chaque rayon, nous avons testé chaque instance une première fois en laissant le positionnement des pièces libre et une deuxième fois en imposant une zone de positionnement pour uniquement une seule pièce.

Les premiers tests ont montré que les quelques instances, que nous avons testés sur des logements de type T6 et T7, dépassaient 8h de temps d'exécution à chaque fois. Nous avons fait d'autres tests avec, respectivement, une durée limitée à 30 min et à 10 min, les gains étaient négligeables. Nous avons, donc, décidé de fixer la durée limite à 10 min.

Dans ce qui va suivre, nous allons détailler les résultats des deux modèles pour les instances avec un rayon fixé à 35 cm et sans la contrainte de positionnement. Nous ferons, à la fin de cette section, un récapitulatif avec les chiffres clés pour l'ensemble des tests.

Pour tous les tableaux des résultats, chaque ligne (un type et une surface données) est le résultat de la moyenne des 10 instances associées.

Les tests ont été réalisés sur un PC avec la configuration suivante :

- Système d'exploitation : Windows 7,
- RAM : 4Go,
- Processeur : intel core i5 520M.

Nous avons utilisé le solveur IBM ILOG CPLEX, version 12.6.

6.2. Résultats du premier modèle

La colonne « CPU » donne le temps de résolution du modèle par CPLEX. La valeur de l'objectif est la demie somme du périmètre. La colonne « ratio surface » donne le ratio de la surface utile calculée et la surface globale du logement. La colonne « erreur relative valeur », donne l'écart relatif entre la valeur en nombre entier et la valeur de la relaxation continue.

Type Logement	Surface (m2)	CPU (S)	Valeur de l'Objectif	Erreur relative valeur	Surface (m2)	Ratio Surface utile / Surface potentielle (%)
T2	40	0,21	22,90	53,95	28,59	71,48
T2	45	0,16	23,65	48,60	30,50	67,79
T2	50	0,21	24,54	71,10	33,31	66,61
T2	55	0,28	27,56	45,09	38,80	70,54
T2	60	0,23	27,63	66,80	42,17	70,29
T3	60	1	31,46	79,64	40,62	67,7
T3	65	0,47	33,13	77,51	44,60	68,62
T3	70	0,37	34,96	76,45	47,80	68,29
T3	75	5	36,54	69,73	53,00	70,67
T3	80	0,37	37,41	77,31	56,77	70,96
T4	80	4	39,85	89,05	53,73	67,16
T4	85	6	41,52	91,88	56,85	66,88
T4	90	40	43,57	89,48	61,65	68,5
T4	95	56	45,51	84,52	66,03	69,51
T4	100	9	46,15	88,04	69,93	69,93
T5	100	178	51,40	93,07	68,05	68,05
T5	105	300	52,46	84,35	70,92	67,54

T5	110	159	53,80	95,22	74,87	68,07
T5	115	139	55,91	85,56	78,75	68,48
T5	120	206	58,10	84,38	82,52	68,77
T6	120	300	59,94	97,83	80,31	66,92
T6	125	300	62,01	100,90	84,82	67,86
T6	130	300	64,26	92,32	88,40	68
T6	135	300	64,94	102,22	92,45	68,48
T6	140	300	67,02	93,32	95,83	68,45
T7	140	300	71,75	93,94	92,98	66,41
T7	145	300	72,66	92,56	97,94	67,54
T7	150	300	73,99	99,04	101,57	67,71
T7	155	300	76,44	95,30	105,22	67,88
T7	160	300	77,95	98,41	108,40	67,75
T8	160	300	83,30	97,19	106,41	66,51
T8	165	300	84,52	95,92	110,45	66,94
T8	170	300	84,85	100,45	114,57	67,39
T8	175	300	87,13	97,40	118,43	67,68
T8	180	300	88,64	93,29	121,77	67,65
T9	180	300	93,39	106,61	117,99	65,55
T9	185	300	95,75	92,58	122,70	66,32
T9	190	300	96,80	100,76	126,08	66,36
T9	195	300	97,55	100,12	129,99	66,66
T9	200	300	101,43	89,08	134,55	67,27

Tableau 19 Résultats des tests

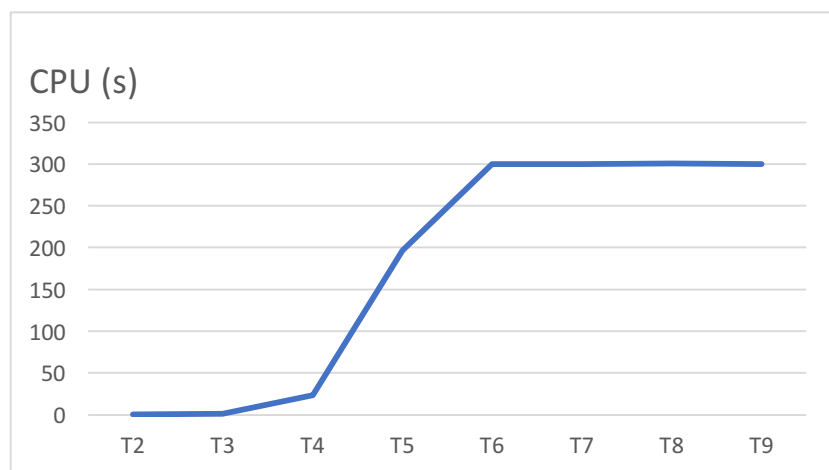


Figure 75. L'évolution du temps de calcul en fonction du type du logement

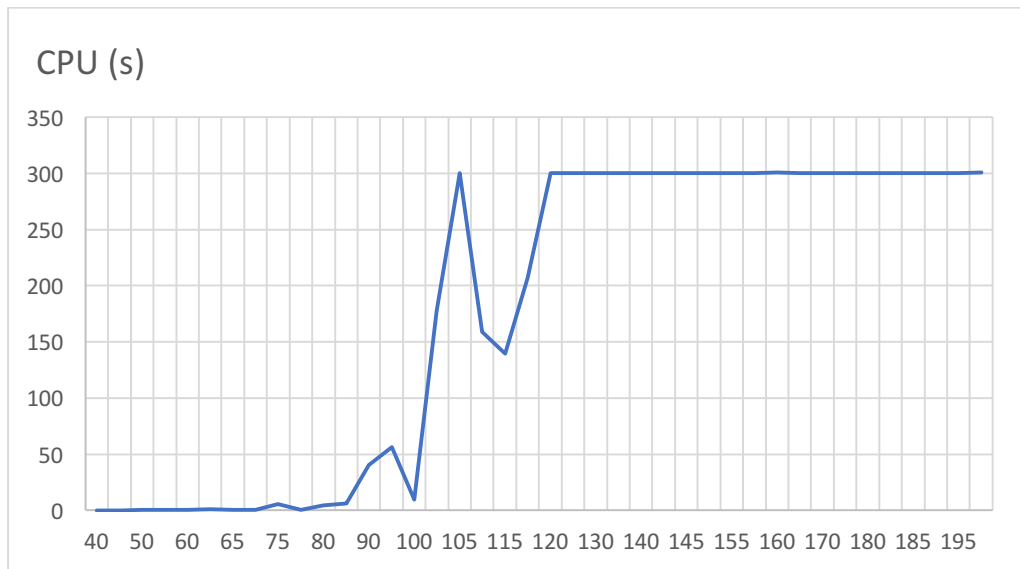


Figure 76. L'évolution du temps de calcul en fonction de la surface

Ces résultats montrent que :

- la surface utile (surfaces des pièces) calculée occupe en moyenne 67.88%. Cet écart varie très peu, peu importe l'instance ;
- Les logements de type T2, T3, T4 et T5 sont résolus en moins de 10 min, à part une seule variante de T5 (105 m²). Le reste des instances ont tous atteint le temps limite. L'écart relatif moyen entre la valeur en entier et la relaxation continue est supérieur à 90% pour les instances de type T4 ou plus. Ce qui peut expliquer le temps de résolution de ces instances.

6.3. Résultats du modèle 2

Ce modèle utilise un paramètre supplémentaire, à savoir le pas de discrétisation. Nous avons effectué des tests avec différentes valeurs 10 cm, 20 cm et 50cm. La variation entre les valeurs des solutions calculées en fonction de ce paramètre étant faible (moins de 1%), nous ne présentons ici que les résultats pour le pas de discrétisation de 50 cm. Le tableau ci-dessus détaille les résultats obtenus.

Pour rappel, le « CPU » donne le temps de résolution du modèle par CPLEX. La « valeur de l'objectif » est la surface utile des pièces. La colonne « ratio surface » donne le ratio de la surface utile calculée et la surface globale du logement. La colonne « erreur relative valeur », donne l'écart relatif entre la valeur en nombre entier et la valeur de la relaxation continue.

Type Logement	Surface (m2)	CPU (S)	Erreur relative Valeur (%)	Val Objectif (Surface en m2)	Ratio Surface utile Vs Surface potentielle (%)
T2	40	1	321,82	23,43	58,58
T2	45	1	176,36	28,06	62,37
T2	50	0,9	217,28	31,20	62,4
T2	55	2	156,84	37,09	67,43
T2	60	1	186,68	39,84	66,39

T3	60	7	294,29	39,60	66
T3	65	4	291,44	42,36	65,18
T3	70	9	282,70	47,60	68
T3	75	4	287,91	49,42	65,9
T3	80	36	260,84	54,70	68,38
T4	80	74	296,96	53,05	66,32
T4	85	116	331,19	56,53	66,51
T4	90	153	304,95	61,87	68,74
T4	95	226	331,56	66,85	70,37
T4	100	175	299,00	70,28	70,28
T5	100	300	350,90	69,68	69,68
T5	105	291	307,36	73,60	70,09
T5	110	245	378,09	76,32	69,38
T5	115	253	340,76	81,81	71,14
T5	120	300	341,62	86,11	71,76
T6	120	300	394,81	83,70	69,75
T6	125	300	423,20	87,95	70,36
T6	130	300	386,90	91,93	70,71
T6	135	300	420,26	96,00	71,11
T6	140	300	376,56	100,56	71,83
T7	140	300	409,81	99,67	71,2
T7	145	300	406,02	102,69	70,82
T7	150	300	439,49	107,64	71,76
T7	155	300	420,71	111,04	71,64
T7	160	300	426,24	115,67	72,29
T8	160	300	471,31	114,31	71,44
T8	165	300	463,37	118,01	71,52
T8	170	300	467,37	121,44	71,43
T8	175	300	475,13	127,55	72,88
T8	180	300	430,22	130,31	72,4
T9	180	300	534,64	128,38	71,32
T9	185	300	450,16	133,42	72,13
T9	190	300	485,55	138,49	72,9
T9	195	300	492,02	140,95	72,29
T9	200	300	452,86	146,57	73,29

Tableau 20 Résultats des tests

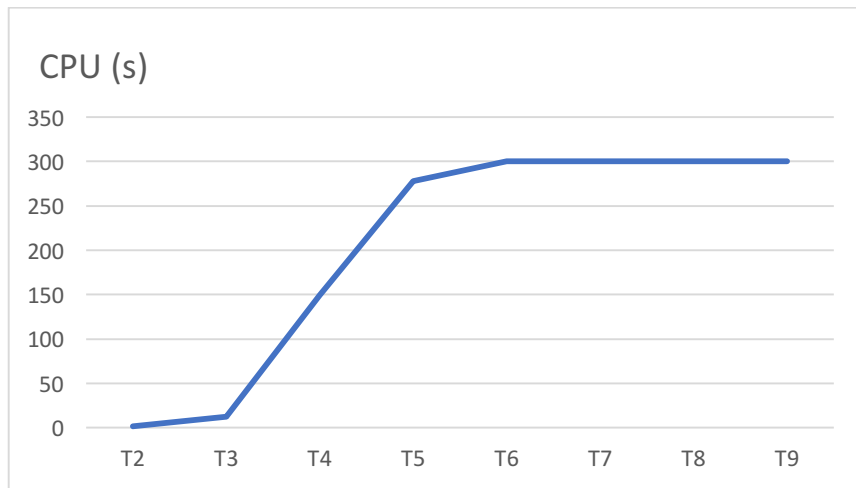


Figure 77. L'évolution du temps de calcul en fonction du type du logement

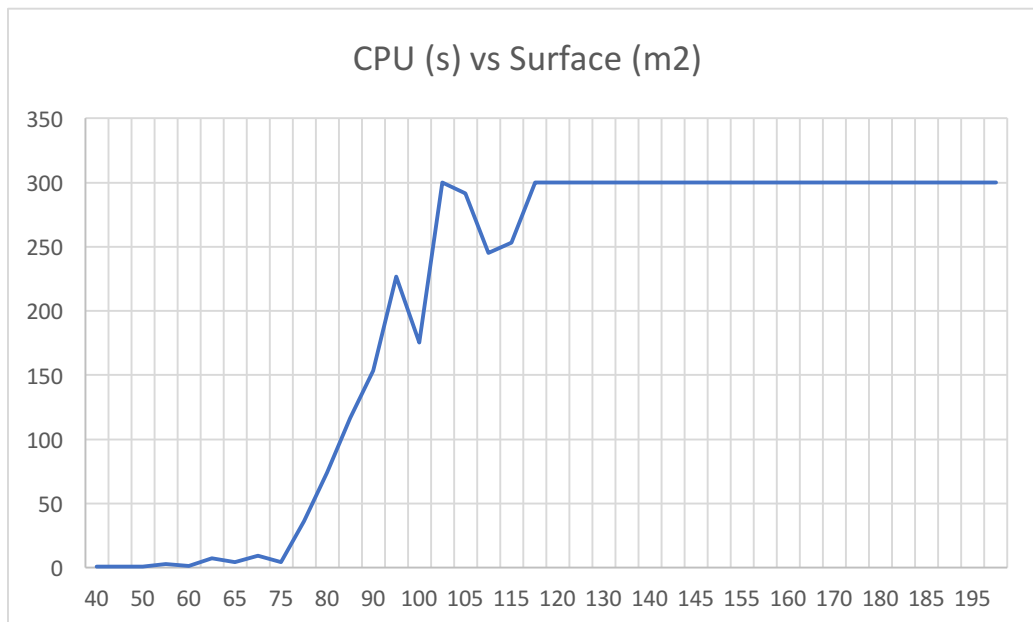


Figure 78. L'évolution du temps de calcul en fonction de la surface

Ces résultats montrent que :

- la surface utile (surfaces des pièces) calculée occupe en moyenne 70%. Cet écart varie très peu, peu importe l'instance ;
- Les logements de type T2, T3, T4 et T5 sont résolus en moins de 10 min, à part deux variantes de T5 (100 et 120 m²). Le reste des instances ont tous atteints le temps limite. L'écart relatif moyen entre la valeur en entier et la relaxation continue est supérieur à 384% pour les instances de type T4 ou plus. Ce qui peut expliquer le temps de résolution de ces instances.

6.4. Modèle 1 versus Modèle 2

Nous présentons dans cette section un récapitulatif des chiffres clefs pour l'ensemble des résultats et une comparaison entre les deux modèles.

Chaque modèle a été testé sur 4 scénarios :

35 SP : les tests sans un rayon de 35 cm et sans prise en compte de la contrainte de positionnement ;

35 AP : les tests avec un rayon de 35 cm et sans prise en compte de la contrainte de positionnement ;

75 SP : les tests sans un rayon de 75 cm et sans prise en compte de la contrainte de positionnement ;

75 AP : les tests avec un rayon de 75 cm et sans prise en compte de la contrainte de positionnement.

Pour chaque colonne, la valeurs minimale et maximale du ratio de la surface utile et surface potentielle et de l'erreur relative valeur sont, respectivement, la plus petite valeur et la plus grande valeur de l'ensemble des instances, et non de la moyenne des instances.

		Model 1				Model 2			
		35 SP	35 AP	75 SP	75 AP	35 SP	35 AP	75 SP	75 AP
Ratio Surface utile / Surface potentielle (%)	Moyenne (%)	67,9	66,2	45,5	42,3	70,3	68,9	45,3	43,6
	La valeur minimale (%)	60,5	58,8	40,7	40,6	57,1	57,6	39,2	40,6
	La valeur maximale (%)	79,0	70,4	54,1	44,1	77,0	75,0	51,2	46,3
Erreur Relative Valeur (%)	Moyenne (%)	90,2	94,27	165,0	180,0	385,0	396,0	735,0	794,0
	La valeur minimale (%)	25,4	47,4	91,2	146,0	127,0	152,0	263,0	499,0
	La valeur maximale (%)	131,4	135,5	231,7	225,3	734,3	731,4	1192,0	1210,0

Tableau 21 Tableau récapitulatif de l'ensemble des tests

Les résultats montrent que :

- le ratio surface utile et surface potentielle varie d'approximativement 2% entre les tests avec et sans la contrainte de positionnement ;
- plus le rayon est grand plus la perte en surface utile est importante. Cela s'explique par le fait que les couloirs prennent plus de places ;
- Pour les petites instances (T2 à T3), le modèle 1 a donné de meilleures valeurs pour les surfaces utiles avec un gain qui peut atteindre 18%. Pour les grandes instances (T4 et plus) le modèle 2 fourni de meilleures valeurs. Les écarts peuvent aller jusqu'à 9%. Ce gain peut s'expliquer par le fait que le modèle 2 maximise la surface des pièces alors que le modèle 1 le fait indirectement en maximisant le périmètre.

7. Bilan

La synthèse du plan d'un logement consiste à aménager un lieu à partir des dimensions du contour extérieur du logement, nombre souhaité de pièces, des intervalles autorisés pour les deux dimensions de chaque pièce, de contraintes d'inclusion de certaines zones par certaines pièces et de contraintes d'orientation sur certaines pièces en plus des besoins et souhaits de l'habitant.

L'objectif étant de placer les pièces dans un premier temps en respectant les contraintes de dimensionnement et de chevauchement. Nous imposons un espace égal à la largeur minimal d'un couloir entre chaque paire de pièces. Si les pièces sont distantes, cela est sans effet, puisqu'elles sont déjà écartées, sinon cela garantit que l'on pourra passer entre les pièces et, de proche en proche, atteindre toutes les pièces. Ensuite on place les portes tel que l'effort à fournir pour se déplacer dans le logement sera à minimiser ou à maximiser (selon le besoin de l'occupant) et pour finir, nous supprimons les couloirs inutiles ;

La prise en compte des handicaps se fait de manière implicite. Par exemple, il faut garantir un couloir d'une largeur minimale de $0.75m$ pour une personne en fauteuil roulant (Cela revient à garantir un espacement entre les pièces de minimum $0.75m$), qu'un espace dégagé de $1.50m$ de diamètre dans la salle de bain pour permettre de manœuvrer le fauteuil roulant (cela revient à garantir que la longueur et la largeur de la salle de bain soient suffisamment grandes pour avoir un espace dégagé de $1.50m$), etc.

Le placement et le dimensionnement des pièces est fondé sur un programme mathématique en variables mixtes. Les deux modèles mathématiques proposés permettent de prendre en compte le nombre de pièces, leur orientation, les dimensions minimales et maximales de chacune, les dimensions du logement ainsi que les besoins et les souhaits de l'occupant. La résolution est faite à l'aide du solveur CPLEX de manière exacte avec CPLEX dans le cadre de notre étude expérimentale.

La modélisation du problème de placement des portes est basée sur la programmation mathématique en nombre entier. La méthode est exacte, elle est basée sur le choix d'une porte par pièce parmi une liste de porte potentielle (générée à un intervalle régulier sur chaque mur éligible aux placements des portes). Le choix de l'ensemble des portes se fait selon les contraintes d'optimisation des chemins. Si l'occupant a des difficultés à marcher et qu'il a l'habitude de faire un parcours (Ex : chambre -> salle de bain -> Cuisine -> Salon), le choix des portes sera tel que la distance parcourue pour faire l'ensemble de ces itinéraires est la plus petite possible. Nous pouvons aussi faire le choix d'éloigner les portes pour obliger l'occupant à marcher plus s'il en a besoin. La résolution est faite avec CPLEX.

Quant à la suppression des couloirs inutiles, la méthode implémentée est une méthode approchée. Elle est faite en trois étapes : on commence par la sélection des murs qui sont autorisés à se déplacer, ensuite, on tri les murs sélectionnés par priorité décroissante et pour finir, on éloigne ces murs un par un, autant que possible, dans l'ordre imposé par les priorités.

8. Discussion

Le premier mérite de la méthode présentée dans ce chapitre est de fonctionner. Ceci est confirmé par le bon comportement de la maquette logicielle qui a été réalisée, dont les résultats sont commentés dans la section des expérimentations. Notons aussi que, pour la première fois dans une étude visant à la production automatisée d'un aménagement de logement avec une méthode non itérative, les contraintes tolérées sont de natures très variées : prise en compte du handicap du futur occupant, gestion des déplacements dans le logement (garantie d'accessibilité à toutes les pièces grâce à une

gestion intégrée des couloirs, optimisation d'itinéraires ...), respects des intervalles de dimensions des pièces etc.

Evidemment, un certain nombre de difficultés restent posées et constituent autant de pistes pour de futures explorations. Elles sont présentées dans ce qui suit.

Le fait de travailler en 2D fait, par exemple, que la méthode ne peut être utilisée dans des appartements mansardés, puisqu'il se peut que des hauteurs insuffisantes ou des poutres empêchent le passage aux endroits recommandés par le logiciel.

Telle qu'elle est spécifiée, la méthode ne gère pas non plus les logements sur plusieurs niveaux. Ce type d'extension *ne se limite pas* à une application de la méthode à chaque niveau. Il faut en effet décider de la répartition des pièces par étage, d'où la nécessité d'une couche méthodologique supplémentaire. L'opération semble tout à fait réalisable : il suffit a priori de prévoir des variables de décision supplémentaires indiquant si une pièce se trouve sur un étage donné et des contraintes spécifiques aux logements sur plusieurs niveaux, par exemple l'obligation d'avoir des toilettes à chaque niveau ou au moins une salle de bains pour deux chambres. Pour que les calculs d'itinéraires restent possibles, il faut également associer à chaque rampe, escalier, ascenseur des informations sur la difficulté qu'ils représentent (distance parcourue, effort fourni dans un sens et dans l'autre, nombre de marches consécutives ...) et sur les endroits auxquels ils débouchent dans les étages.

La décomposition d'un problème complexe en plusieurs sous problèmes a pour but de réduire sa complexité pour faciliter sa résolution. Cette décomposition entraîne de fait une relaxation de certaines contraintes qui ont un impact sur plusieurs sous problèmes à la fois. C'est le cas pour nous, la satisfaction de certaines contraintes dépendant à la fois du placement des pièces, du placement des portes et de la suppression des couloirs. Une piste de réflexion pour remédier à ce problème est l'étude de la faisabilité d'intégrer les deux premières phases dans un seul modèle.

Les pièces sont rectangulaires, une forme plus complexe rendrait la modélisation et la résolution plus difficiles pour le peu à gagner en qualité, d'autant plus qu'une grosse majorité des pièces dans les appartements réels sont rectangulaires. De plus, rien n'empêche l'architecte de personnaliser la forme des pièces a posteriori : s'il se contente d'arrondir un mur, par exemple pour faire apparaître un bow-window, l'influence sur la satisfaction des contraintes sera très faible. Dans le souci de voir ce qu'apportent des formes plus libres, l'étude proposée dans le chapitre suivant, qui porte sur l'optimisation d'un aménagement *existant*, autorise que les pièces aient des formes polygonales. La section des expérimentations chiffre l'amélioration apportée par cette liberté accrue.

Une généralisation possible est de permettre au modèle de placement des portes de choisir le nombre de portes par pièce. Ceci nécessiterait d'adapter les contraintes en lien avec la conservation des flots comme la contrainte (3) du modèle en question. Le nombre de couloirs (zones inutiles) à supprimer dans la 3^{ème} phase serait réduit mais la méthode elle-même resterait inchangée.

Chapitre IV --- Amélioration d'un aménagement

1. Introduction

Ce chapitre présente une autre manière d'aboutir à l'aménagement d'un logement ; la définition d'un aménagement reste celle du chapitre précédent, à savoir un choix de la position, de la forme et des dimensions des pièces, des couloirs et des portes.

Cette deuxième approche a été suggérée par ce qui se passe depuis une dizaine d'année et de manière de plus en plus systématique dans le domaine de l'ingénierie mécanique : dans ce secteur, en tenant compte d'un certain nombre de contraintes, les ingénieurs conçoivent des pièces mécaniques. Mais ces pièces sont *ensuite* soumises à un outil d'optimisation qui, sans modifier les surfaces de contact avec les autres organes du mécanisme et sans dégrader les performances mécaniques des pièces, change leur forme jusqu'à parfois les alléger de 60 %, ce qui représente d'énormes économies en matière première : on confie donc à un outil informatique le soin d'améliorer la proposition d'un opérateur humain. Notre objectif est de faire de même avec les aménagements de logements : nous souhaitons partir 1) d'un aménagement conçu par un professionnel et 2) de toutes les contraintes qu'il s'est efforcé de satisfaire et exploiter l'énorme capacité de calcul des ordinateurs pour produire un aménagement encore meilleur.

Pour terminer le parallèle avec l'ingénierie assistée par ordinateur, il faut noter qu'une intervention humaine reste encore nécessaire après l'optimisation des pièces mécaniques par le logiciel : en effet, à cause de la technologie utilisée, les pièces optimisées ont souvent un aspect crénelé, qui n'est ni esthétique ni facile à fabriquer. C'est pourquoi un opérateur humain a souvent la charge d'observer le design élaboré par le logiciel, d'en comprendre la philosophie et de dessiner la version finale de la pièce. La Figure 79 illustre ce processus avec une poulie de lave-linge : à gauche, on voit la pièce telle qu'un ingénieur l'a conçue ; la pièce optimisée par le logiciel est au centre ; cette deuxième version suggère que quatre rayons suffisent et que ces derniers devraient se scinder avant d'atteindre la périphérie de la poulie. Ayant compris cela, un concepteur humain a produit la version de droite.

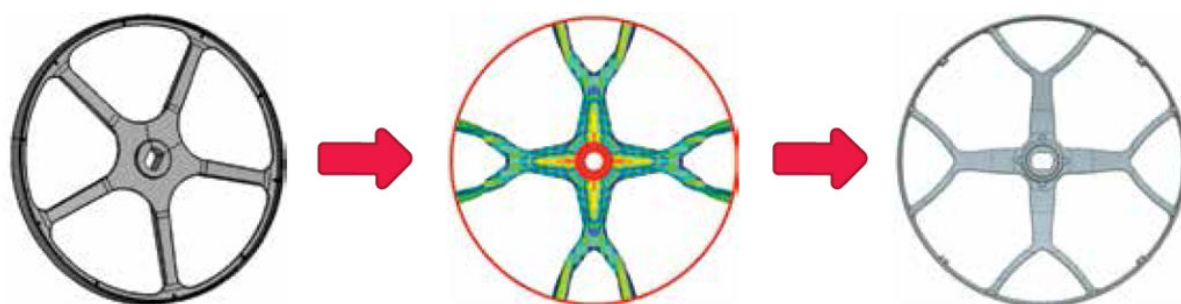


Figure 79. Les trois étapes du processus d'optimisation en IAO

De la même manière, dans le processus d'optimisation d'un agencement de logement, nous ne visons pas la perfection géométrique du plan produit mais plutôt à la production d'un agencement riche de

suggestions, conforme à toutes les exigences exprimées mais que l'utilisateur pourra éventuellement retoucher.

Le déroulement de cette deuxième méthode d'amélioration d'un aménagement est illustré par la Figure 80. On commence par l'acquisition du plan à améliorer, des contraintes réglementaires et des besoins et souhaits du futur occupant. Les notions de besoins et souhaits ont déjà été présentées dans le chapitre précédent mais, cette deuxième méthode ne s'appuyant pas sur la programmation linéaire en nombres entiers, il n'est plus nécessaire de se limiter à des besoins/souhaits que l'on peut traduire en des contraintes linéaires. En conséquence, la panoplie des besoins/souhaits est beaucoup plus étendue. D'un point de vue technique, les contraintes réglementaires et les besoins/souhaits du futur occupant sont exprimés en instanciant ce que nous appelons des critères (d'évaluation). Cette instanciation est pour l'instant manuelle, c'est-à-dire que c'est l'utilisateur du logiciel qui indique, au moyen d'une interface graphique, quel critère ou quelle combinaison de critères il faut instancier pour satisfaire telle réglementation, tel souhait ou tel besoin.

Une fois les acquisitions réalisées, la méthode itère sur deux opérations, l'évaluation de l'agencement et la modification de l'agencement. Cette itération se poursuit jusqu'à ce que l'une des conditions suivantes soit satisfaite : un aménagement parfait (atteignant la note maximum) a été trouvé, il y a stagnation (plus d'améliorations) ou la durée maximale allouée au temps de calcul a été atteinte.

L'évaluation d'un agencement est faite relativement à un individu donné, celui dont les besoins et souhaits ont été traduits en critères. Elle est obtenue en notant d'abord tous les critères puis en agrégeant les notes obtenues en une note globale.

La note d'un critère est généralement calculée en utilisant ce que nous appelons des *opérateurs* (de bas niveau). Par exemple, un des *critères* impose qu'une pièce donnée ait une aire supérieure à une valeur donnée. Ce critère repose sur l'*opérateur* de calcul de l'aire d'un polygone, détaillé dans le chapitre sur l'algorithmique géométrique. La nuance entre critères et opérateurs vient du fait que les premiers sont exprimés dans un langage de haut niveau, quasiment naturel, alors que les autres sont des traitements de moindre niveau sémantique, dont la réalisation informatique, sous la forme d'un sous-programme, est aisée (pour un informaticien).

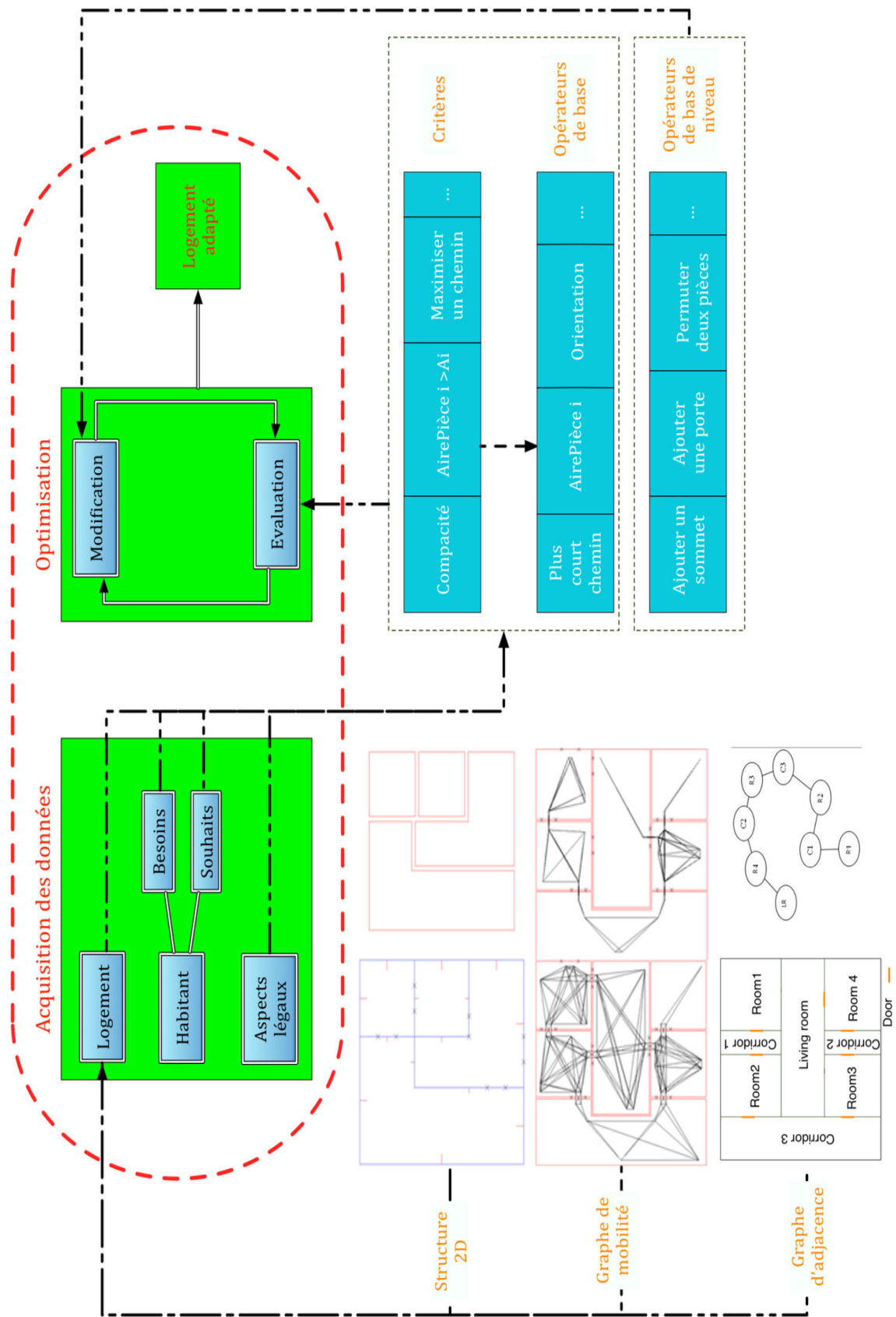


Figure 80. Architecture logicielle de l'optimiseur

Cette nouvelle approche présente de nombreuses différences avec la méthode synthétique du chapitre précédent. Elles sont détaillées dans la suite de ce chapitre et énumérées ci-dessous :

- avant toute chose, comme on le comprend à la lecture des lignes qui précèdent, la méthode est une heuristique, fondée sur un ensemble d'algorithmes de type « recherche locale » et « méta-heuristique ». Le canevas général présenté ci-dessus (Figure 80) peut être mis en œuvre de différentes façons et nous en présenterons quatre ;
- les pièces n'ont plus l'obligation d'être des rectangles, ce sont maintenant des polygones, éventuellement non convexes. Des pièces en L ou en U deviennent donc possibles. En conséquence, une réelle structure de données devient nécessaire pour modéliser le logement, ainsi que de nombreux algorithmes géométriques ; ces derniers ont été présentés dans le Chapitre II;
- le nombre de portes d'une pièce n'est plus réduit à un, il est déterminé par la méthode ;
- les couloirs sont considérés comme des pièces et, ainsi, deviennent des entités explicites. Dans la première méthode, les couloirs n'étaient pas modélisés et ne pouvaient être manipulés. A présent, le pourtour d'un couloir étant connu, le couloir peut faire l'objet des mêmes requêtes qu'une pièce (calcul de l'aire, échange avec une autre pièce, déplacement d'un sommet ...) ;
- le nombre de besoins/souhaits pris en compte, comme indiqué plus haut, est sensiblement plus élevé ;
- à chaque évaluation d'un agencement, la totalité des critères traduisant les besoins/souhaits est évaluée. Le problème est donc traité dans sa globalité à chaque itération et non décomposé en trois étapes successives comme dans la méthode employée dans la synthèse d'un aménagement ;
- la méthode est extensible : bien que considérablement étendue, la panoplie des critères actuellement disponibles reste évidemment limitée. Nous verrons que, dans une certaine mesure, il est possible de l'agrandir sans intervention d'un informaticien. Et quand cette intervention est nécessaire, nous verrons qu'elle est facilitée par l'architecture logicielle qui a été adoptée.

Notons que les exemples et normes utilisés dans ce chapitre s'appuient sur la réglementation française et reste parfaitement ajustable à d'autres réglementations.

Ce chapitre est construit comme suit. On présente d'abord la structure de données qui modélise un agencement et qui assure les services de base sur lesquels s'appuient les algorithmes géométriques de plus haut niveau présentés dans le Chapitre II. On détaille ensuite le processus d'évaluation de l'adaptation d'un aménagement à un occupant. Viennent ensuite une énumération des outils qui permettent de modifier un agencement, une présentation des quatre heuristiques qui réalisent le canevas général de la méthode d'amélioration décrite plus haut, une synthèse et une discussion.

Pour éviter un peu de lourdeur dans la rédaction, nous emploierons indifféremment les termes d'aménagement, de plan, de solution et de logement pour désigner la manière dont les pièces, portes et couloirs sont placés et dimensionnés.

2. Modélisation d'un aménagement

La modélisation d'un aménagement doit permettre la réalisation des services exigés du logement pour l'adapter au mieux aux critères de l'habitant, tout en respectant la réglementation en termes de construction. Ces services sont les requêtes auxquelles le logement doit répondre pour pouvoir le modifier et l'évaluer.

Pour cela, deux aspects sont à prendre en considération : l'aspect topologique et l'aspect mobilité dans le logement. Le premier aspect est représenté par une structure de données en 2D, quant au deuxième aspect, il consiste en deux graphes : un graphe qui recense tous les déplacements possibles dans le logement, en prenant en compte la corpulence ou l'encombrement (s'il est appareillé) de l'occupant et une version simplifiée de ce graphe, appelée graphe d'adjacence [10].

2.1. Préambule

Un logement est décrit comme un ensemble de plateaux, chaque plateau (Figure 81) est divisé en plusieurs polygones représentant les pièces. Chaque segment d'un polygone est un mur, sur lequel des ouvertures (portes ou fenêtres) peuvent être placées ; dans nos illustrations, seules les portes sont dessinées.

Dans ce qui suit, les pièces sont planes avec une forme polygonale quelconque. Les couloirs sont considérés comme des pièces (Exemple : Couloir_0_plateau_0 (6) dans la Figure 81). Plusieurs ouvertures par mur sont possibles. Deux murs qui se succèdent peuvent être alignés (Ex : Les murs 16 et 21) et les murs sont subdivisés sur leurs intersections avec les autres murs (Ex : Les murs 10 et 11).

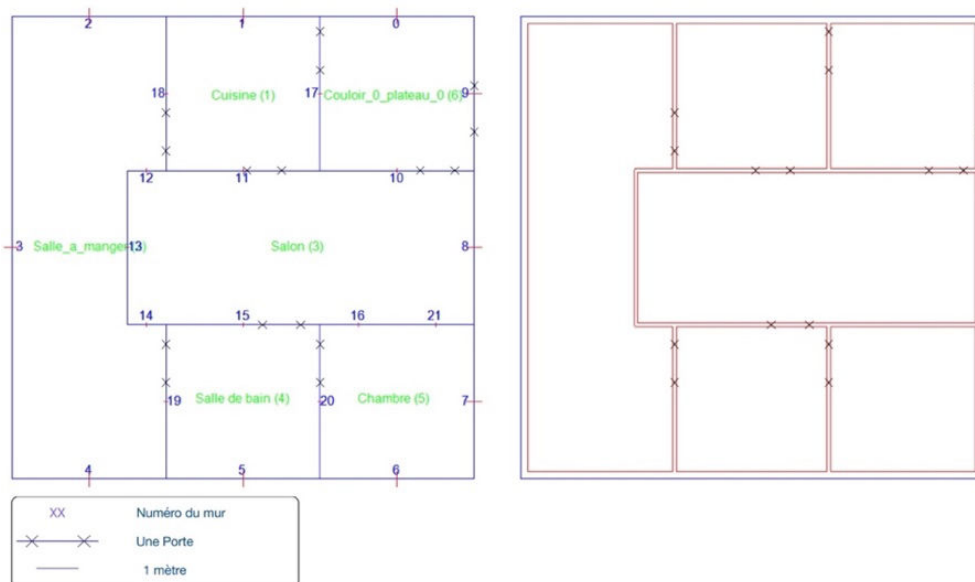


Figure 81. (à gauche) Description en 2D filaire du logement, (à droite) description en 2D avec la prise en compte des épaisseurs et de la position de la matière des murs

2.2. Description de la structure de données

Même si notre principal objectif est la description d'un logement, la structure de données décrite ci-dessous, a été conçue pour la description de tout type de bâtis (logement, ERP, immeuble etc.). Certains paramètres peuvent ne jamais être utilisés pour décrire un logement.

2.2.1. Ouverture

Une ouverture peut être une fenêtre ou une porte. Dans les deux cas, elle est décrite par la position de l'axe de rotation sur le mur, la position où termine le vantail, le sens de l'ouverture (à gauche, à droite, les deux sens ou coulissante), la hauteur de la poignée par rapport au sol. En plus, si c'est une fenêtre, un paramètre supplémentaire est ajouté pour représenter l'altitude du point bas de la fenêtre.

Les fenêtres sont donc modélisées dans l'agencement initial du logement, celui qui doit être amélioré, mais elles ne sont pas prises en compte totalement dans la version courante du processus d'amélioration. Nous verrons dans la suite de ce chapitre que le placement des fenêtres dépend des murs, un mur mitoyen ne peut supporter une fenêtre par exemple. Les murs qui peuvent contenir des fenêtres sont gérés implicitement lors du placement des pièces, une pièce orientée exprès du côté le plus ensoleillé est une manière implicite de prendre en compte les fenêtres dans l'approche proposée. Le placement exact des fenêtres sur un mur, pour une meilleure luminosité, est donc laissé à l'architecte.

2.2.2. Mur

Un mur est entre autres décrit par les coordonnées de ses deux extrémités. C'est donc un segment et c'est pourquoi nous l'appelons souvent un mur filaire. D'autres informations lui sont attachées : les deux pièces auxquelles il appartient (il y a toujours exactement deux pièces, voir ci-dessous la définition du plateau), ses ouvertures, des booléens disant s'il est porteur, si c'est un mur extérieur qui peut contenir une fenêtre, son épaisseur et la position de sa matière (à gauche, à droite, de part et d'autre). Sur la Figure 81, les épaisseurs sont matérialisées par des petits segments perpendiculaires aux murs, placés en leur milieu.

Ces deux derniers attributs (épaisseur et la position de la matière) sont nécessaires pour le calcul des surfaces des pièces. La différence entre les aires obtenues avec et sans prise en compte de ces attributs ne peut être négligée. Par exemple, dans la Figure 81 l'aire de la salle à manger est de $44 m^2$ si on ne considère ni l'épaisseur ni la position de la matière, et de $39.60 m^2$ en considérant une épaisseur de $20 cm$ et une distribution de la matière de part et d'autre pour tous les murs, soit une différence de $4.4 m^2$.

2.2.3. Pièce

Une pièce est représentée par un polygone. Elle est décrite par son nom, les murs qui la composent et un attribut qui permet de savoir si la traverser pour aller à une autre pièce est considéré comme gênant ou pas (par exemple, une chambre ne tolère pas le passage, certaines personnes acceptent que le salon soit une pièce de passage et un couloir est automatiquement considéré comme "tolérant le passage").

Notons que d'un point de vue structure de données, un couloir est considéré comme une pièce. Deux sortes de couloirs existent : les couloirs explicites, fournis dans le plan à améliorer, et les couloirs implicites, qui sont détectés par le logiciel. Ces derniers sont limités par tous les murs qui n'appartiennent qu'à une seule pièce dans le plan (initial ou amélioré) du logement. Cette détection est moins simple qu'il n'y paraît au premier abord et a été détaillée dans le Chapitre II.

2.2.4. Plateau

Un plateau est un ensemble connexe de pièces situées à la même altitude ; c'est donc une zone dans laquelle on peut se déplacer sans que le relief pose un problème. Il est composé d'un contour extérieur (considéré comme une pièce factice dans la structure de données), d'un ensemble de pièces et d'une altitude. Nous avons choisi de mémoriser une altitude plutôt qu'un numéro d'étage parce qu'il arrive qu'un étage comprenne des différences de niveaux (parfois pour des raisons esthétiques, parfois parce que deux plateaux à des niveaux légèrement différents ont été raccordés), ce qui va à l'encontre de notre définition du plateau. La contrainte de connexité du plateau implique aussi d'utiliser plusieurs plateaux pour un étage donné, quand un bâtiment contient plusieurs tours. Par exemple, pour le bâtiment de la Figure 82, le dernier niveau est décrit par trois plateaux distincts.

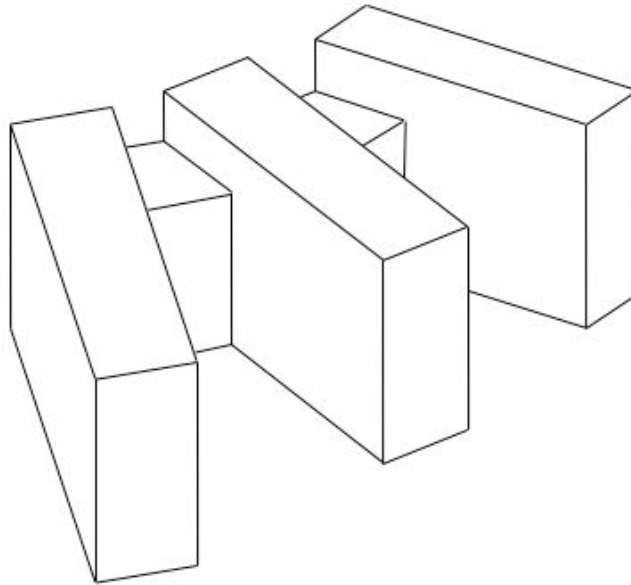


Figure 82. Une vue simplifiée de l'hôpital de Thionville (Moselle)

Un mur est associé à un seul plateau et appartient à exactement deux pièces (car on considère que le contour extérieur et les couloirs sont des pièces) comme illustré dans la Figure 83 :

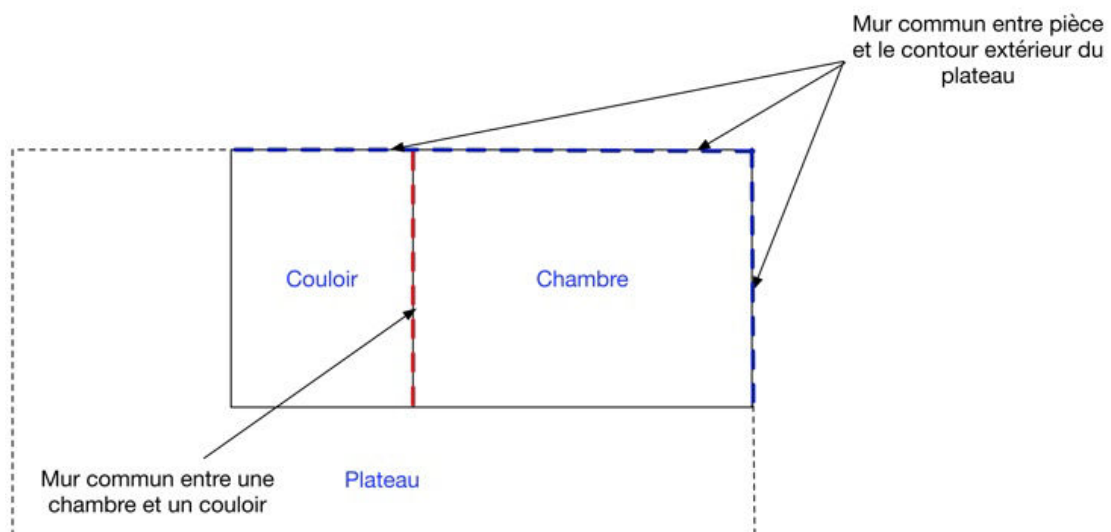


Figure 83. Exemple de murs communs sur un plateau

2.2.5. Logement

Un logement est décrit à l'aide d'un ensemble de plateaux et d'un vecteur qui donne la direction du nord.

2.3. Graphe de mobilité

Le graphe de mobilité a pour objectif de recenser tous les déplacements *élémentaires* possibles de l'occupant lorsqu'il se déplace dans le logement d'une pièce à une autre. Un déplacement *élémentaire*

est un segment de droite maximal d'un itinéraire dans le logement. La Figure 84 montre tous ces segments élémentaires dans un logement.

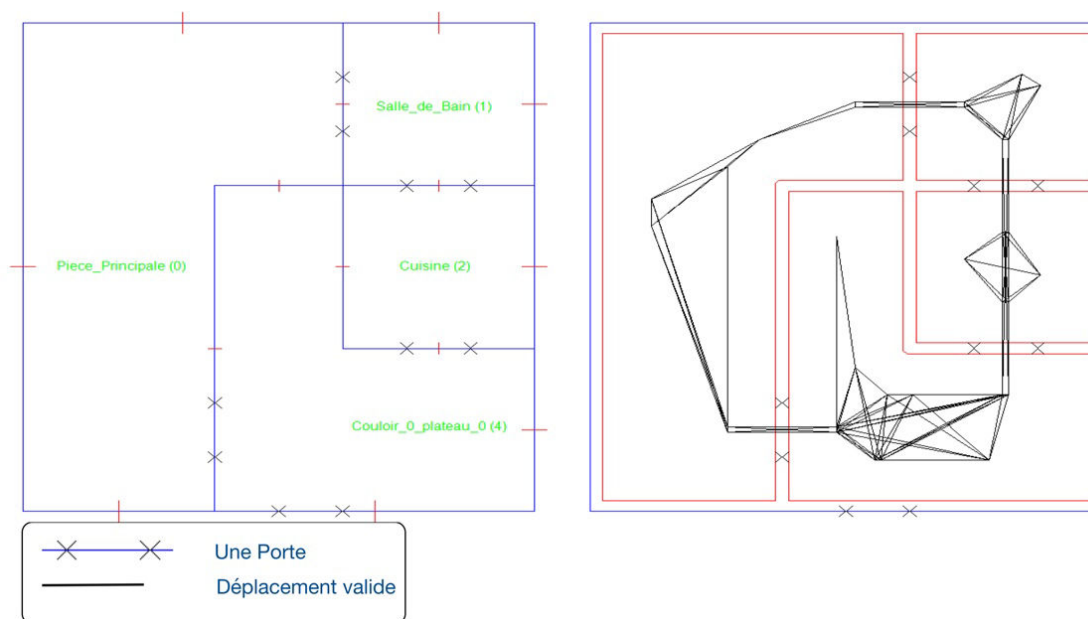


Figure 84. Exemple d'un graphe de mobilité pour une personne en fauteuil roulant

Une précision doit être donnée sur la phrase "tous les déplacements élémentaires possibles" : Les points de départ et d'arrivée potentiels étant en nombre infini, le nombre d'itinéraires l'est aussi. Il n'est évidemment pas question de calculer tous les segments qui constituent cette infinité d'itinéraires : une sélection parmi tous les itinéraires est effectuée et elle est présentée et justifiée plus bas. De même, il est abusif d'affirmer que les itinéraires sont composés de segments de droites et, donc, qu'ils sont des lignes brisées : dans l'hypothèse où les pièces sont des polygones (ce qui est le cas), les itinéraires optimaux sont en réalité composés de segments *et* d'arcs de cercles. Le choix de ne considérer que des segments de droite (et de négliger les arcs) est justifié plus bas, dans la présentation détaillée du processus de construction du graphe de mobilité.

Notons que ce graphe est calculé à partir des pourtours réels des pièces (voir Figure 84), dont l'algorithme de calcul a été présenté en détail dans le Chapitre II. Pour mémoire, le pourtour d'une pièce est déduit des murs "filaires" de la pièce (les murs "filaires" sont les murs idéalisés conservés dans la structure de données 2D). Chaque mur filaire engendre un segment dans le pourtour de la pièce et la position de ce segment dépend de la manière dont la matière est placée sur le mur filaire (Figure 85) : si la matière du mur est entièrement du côté du mur filaire où ne se trouve pas la pièce (cas 1), le segment du pourtour est superposé au mur filaire, sinon le segment est strictement parallèle au mur filaire, à une distance égale soit à l'épaisseur (cas 2) soit à la demi-épaisseur du mur filaire (cas 3).

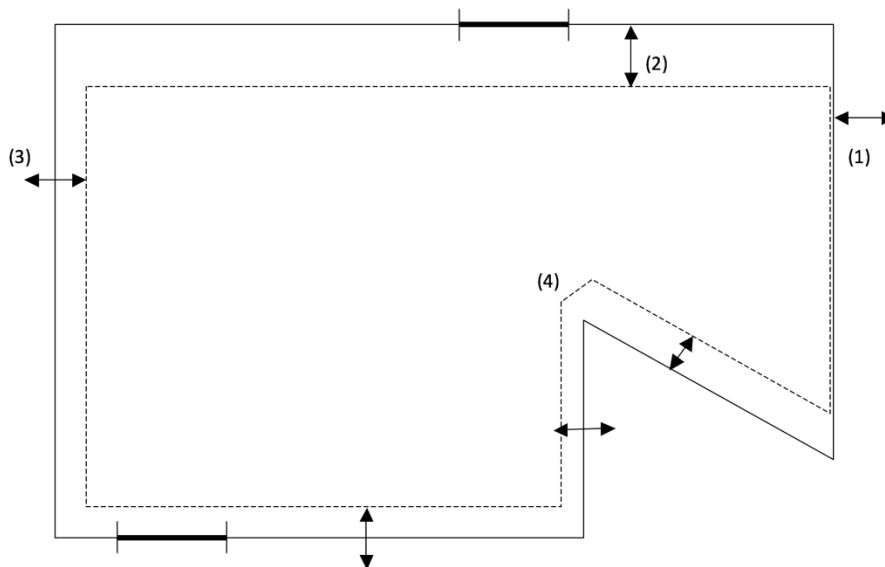


Figure 85. *En trait plein, les murs filaires ; les doubles flèches donnent la position de la matière sur les murs filaires ; en pointillés, le pourtour de la pièce*

Quand deux murs filaires consécutifs forment un angle trop concave (cas (4)), l'angle est cassé pour éviter que le pourtour ne forme une pointe qui se prolongerait loin dans la pièce. Cette précaution est rendue nécessaire par le fait que, pendant la modification du logement, les pièces peuvent temporairement prendre des formes aussi irrégulières que celle de la Figure 85.

2.3.1. Dépendance du graphe à l'habitant

Un habitant ne peut être approché par un point car l'espace qu'il occupe est non nul et l'idéaliser sous la forme d'un point reviendrait à calculer les mêmes itinéraires pour toutes les personnes. Par exemple, un couloir peut être trop étroit pour une personne en fauteuil roulant, ce qui peut lui interdire d'emprunter le plus court chemin et, donc, allonger son chemin (si un chemin existe effectivement).

Pour tenir compte de l'encombrement de l'habitant dans le calcul de ses déplacements, nous avons fait le choix de l'approcher par un cercle. Ainsi, et comme cela est préconisé par la loi¹⁵, pour une personne en fauteuil roulant, le diamètre du cercle qui l'approche lors de ses déplacements est de 1.5 m pour lui permettre de pivoter sur elle-même (Figure 86). Pour une personne obèse ou pour une personne de corpulence normale le diamètre du cercle est plus petit (respectivement 80 cm et 60 cm).

Le choix de représenter l'habitant avec un cercle et non avec la projection au sol de sa silhouette est justifié par trois arguments : avant toute chose, comme on le disait ci-dessus, c'est recommandé réglementairement pour les personnes en fauteuil ; ensuite, cela simplifie considérablement le calcul

¹⁵ Loi n°2005-102 du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées

de l'emprise de l'occupant sur le sol ; enfin, rapportée à la longueur des itinéraires, la majoration sur ces longueurs est très faible (quelques centimètres).

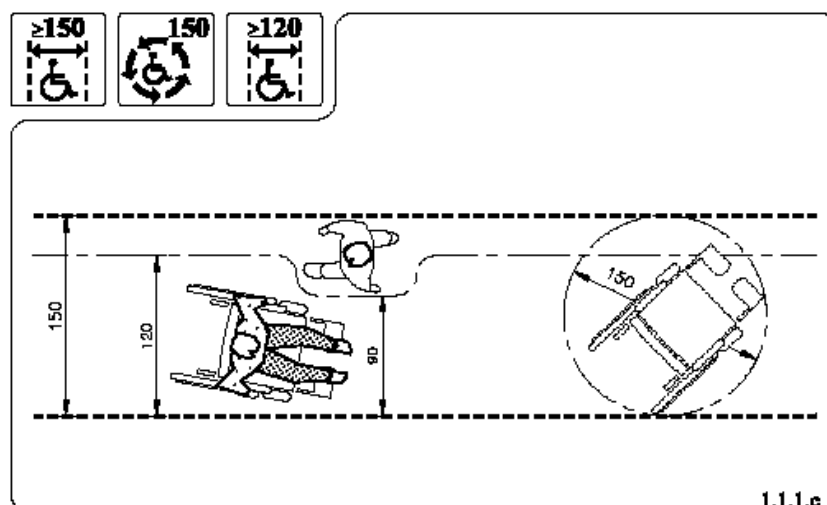


Figure 86. L'encombrement pour une personne en fauteuil roulant [36]

Un deuxième cercle d'un rayon plus petit est utilisé pour le franchissement des portes car l'habitant n'a pas besoin de pivoter sur lui-même pendant qu'il traverse la porte.

Dans ce qui suit, et pour différencier les deux rayons, nous appelons le premier *rayon de l'habitant* et le deuxième *rayon pour franchissement des portes*.

2.3.2. Points caractéristiques

La Figure 87 montre un plus court chemin d'un point A de la pièce P1 à un point B de la pièce P2. On constate que le chemin est constitué de segments et d'arcs de cercles dont le rayon est celui de l'habitant. Sur la partie droite de la même figure, on voit une approximation polygonale de ce plus court chemin. Cette approximation est une ligne brisée passant par des points appelés, dans ce qui suit, « points caractéristiques ». Nous avons choisi d'utiliser la version *approchée* car elle n'est que très légèrement plus longue que la version exacte et évite la mise en place d'une conséquente série d'algorithmes (ceci est justifié plus bas).

En conséquence, tout itinéraire dans le logement est exprimé comme une combinaison de déplacements rectilignes reliant des points caractéristiques ; ces derniers sont de trois sortes : ceux qui permettent le franchissement d'une porte (4 au maximum), ceux qui permettent de contourner un sommet concave (comme C et D, qui se trouvent en vis à vis d'un sommet concave quand il est vu de l'extérieur de la pièce) et les points choisis comme points de départ dans une pièce (A et B dans la Figure 87).

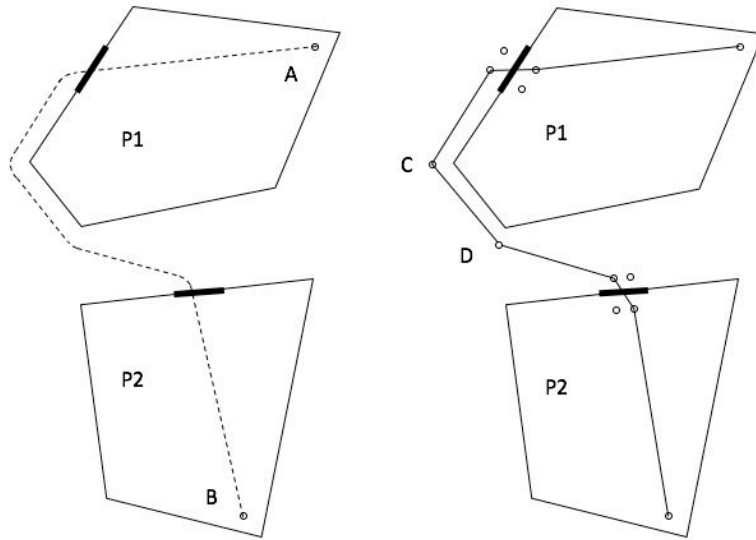


Figure 87. Un plus court chemin exact de A à B (à gauche) et le plus court chemin effectivement calculé, passant par plusieurs points caractéristiques (à droite)

2.3.2.1. Points caractéristiques d'une porte

Pour parfaitement minimiser les longueurs d'itinéraires au moment du franchissement des portes, il faudrait autoriser des trajectoires contenant des arcs de cercles. Ceci est illustré par la Figure 88: le plus court chemin de A à B contient une section droite [A, J], pendant laquelle l'occupant longe le mur à une distance R (le rayon de l'occupant). Le franchissement de la porte commence alors et, donc, le rayon passe brutalement à R' (le rayon de franchissement). Pour gérer cette discontinuité, plusieurs approches sont envisageables mais celle qui produit le chemin le plus court consiste à faire suivre au petit cercle un segment issu de J et tangent à un cercle de rayon R' centré sur le premier coin du mur à contourner. Vient ensuite une portion d'arc de cercle, une brève section rectiligne puis un nouvel arc et deux sections droites.

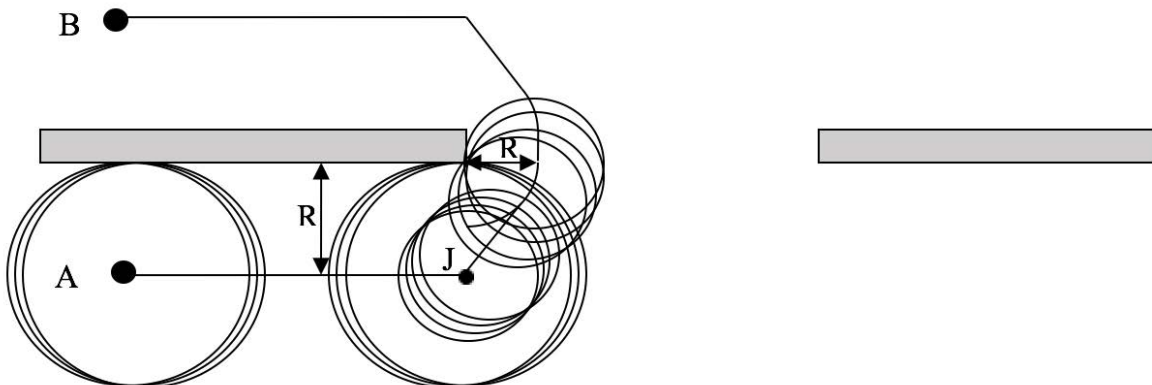


Figure 88. Franchissement de porte optimal pour aller d'A à B

Un itinéraire plus simple, constitué uniquement de segments de droites, consiste à passer par les points caractéristiques $PC3$ et $PC1$ (Figure 89) : $[A, PC3, PC1, B]$. La pénalité sur la longueur de l'itinéraire est faible. Elle est donnée par la formule suivante :

$$2 * (R + R') - 2 * [\sqrt{R^2 - R'^2} + R' * \text{asin}(R' / R)]$$

Autrement dit, si on considère le seul franchissement (section $[J .. K]$) et si on néglige l'épaisseur du mur, on obtient pour :

- une personne en fauteuil ($R = 0.75, R' = 0.5$) : des longueurs de 1,85 m ou 2,5 m, soit un allongement de 0,65 m ;
- un homme de stature normale sans handicap ($R = R' = 0,3$), des longueurs de 0,94 m ou 1,2 m, soit un allongement de 0,26 m.

La simplification consistant à passer par les points caractéristiques proposés allonge donc peu l'itinéraire mais libère du développement de plusieurs algorithmes géométriques :

- calcul d'un segment tangent à un arc et issu d'un point ;
- calcul de la distance d'un arc de cercle à un segment (une portion de trajectoire, ici l'arc, doit garantir que la personne qui suivrait cette trajectoire ne se heurterait pas aux murs de la pièce. Pour cela, il faut que la distance de l'arc à tous les murs de la pièce soit supérieure à R) ;
- calcul plus compliqué de la longueur d'une portion de trajectoire: si une portion de trajectoire est un segment, sa longueur est la distance entre ses extrémités; si c'est un arc, il faut calculer l'ouverture angulaire de l'arc pour obtenir sa longueur.

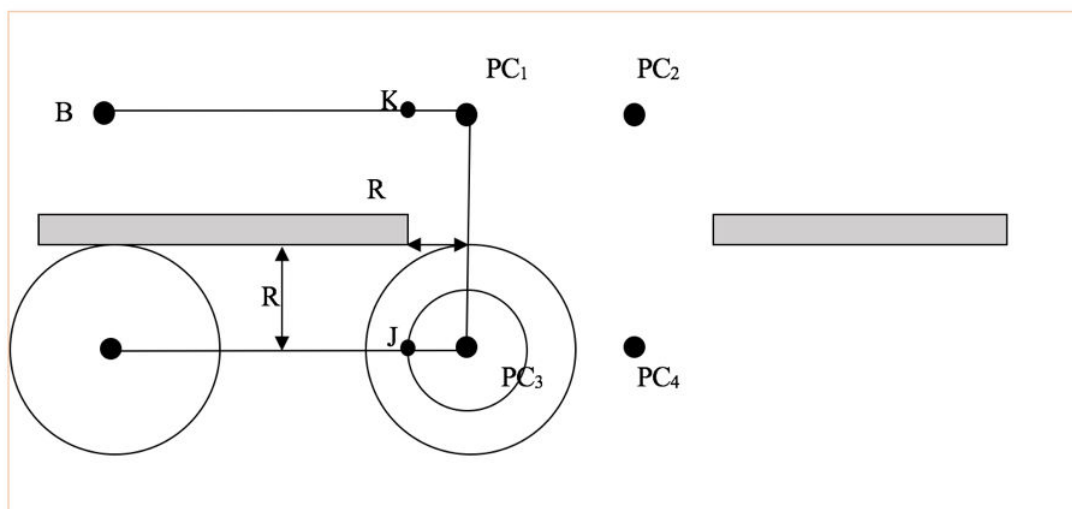


Figure 89. Franchissement de porte sous-optimal pour aller de A à B

Nous avons donc fait le choix d'imposer que les itinéraires qui passent par une porte empruntent les points $PC1 .. PC4$. Ces quatre points peuvent n'être que deux si la largeur de la porte est égale à exactement deux fois le rayon de franchissement de la porte R' et peuvent même ne pas exister du tout si la largeur est inférieure à $2.R$ (la porte ne peut pas être franchie). Si un des points caractéristiques est à une distance inférieure à R d'un mur autre que le mur de la porte alors le point n'est pas construit car l'espace disponible à cet endroit n'est pas suffisant pour garantir la circulation de l'occupant.

2.3.2.2. Points caractéristiques d'une pièce

Deux types de points sont associés à une pièce : les points qui correspondent à des sommets concaves, et les points de départ et d'arrivée des itinéraires depuis et vers la pièce.

2.3.2.2.1. Sommets concaves

Une pièce est non convexe (nous dirons concave) si au moins un de ses sommets est concave. Un sommet est concave quand ses deux murs forment un angle de plus de 180° , cet angle étant mesuré à l'intérieur de la pièce (le point C dans la Figure 90). Un tel sommet est synonyme d'une jonction qui brise un mouvement rectiligne comme illustré dans la figure suivante : le déplacement du point A vers le point B n'est pas un mouvement rectiligne, il est la somme de deux mouvements rectilignes et d'un mouvement circulaire.

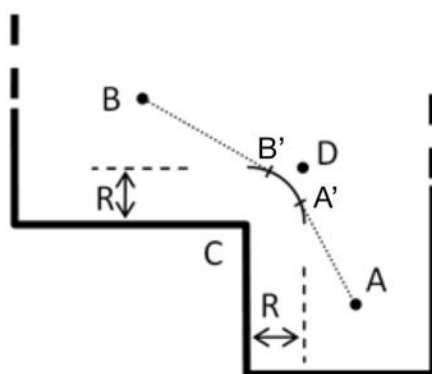


Figure 90. Exemple d'un point concave

Quand un sommet est concave, il faudrait donc lui associer des arcs de cercle pour le contourner, ces arcs dépendant des positions relatives du point qui précède et du point qui suit, dans la trajectoire. Dans le cas de la Figure 90, ces deux points étant B et A , l'arc va de B' à A' mais avec d'autres points que B et A , l'arc aurait d'autres extrémités (mais il serait toujours porté par le même cercle). Compte tenu des complications que cela amène dans la mise en œuvre, compte tenu aussi du fait qu'il s'agit d'un problème très annexe par rapport au sujet de cette thèse et compte tenu du très faible gain que cela amène sur la longueur de l'itinéraire, nous nous sommes contenté de placer un point caractéristique sur la bissectrice de l'angle formé par les deux murs, à intersection des parallèles aux deux murs à une distance R (voir le point D dans la figure ci-dessus).

Il n'y a pas à craindre que ce point soit placé à une distance excessive sur la bissectrice car, dans la construction du pourtour, les angles très concaves ont été biseautés.

Si la pièce est convexe (tous les coins sont convexes), aucun point caractéristique de cette sorte n'est utile car tout point dans la pièce (pas trop proche des bords) peut être atteint directement depuis n'importe quelle porte.

2.3.2.2.2. Points de départ de la pièce

Quand un itinéraire entre deux pièces est demandé, il faut décider de quel point on part dans la première pièce et sur quel point de la deuxième pièce on arrive. Ces points de départ et d'arrivée ne peuvent être des portes car plus la pièce est grande plus il y aurait une sous-estimation de la longueur

de l'itinéraire. De plus, l'itinéraire entre deux pièces partageant une porte serait de longueur nulle, ce qui ne reflète pas la réalité.

La solution la plus intuitive pour pallier à ce problème consiste à choisir le centre de la pièce comme point de départ et d'arrivée. Néanmoins, le centre d'une pièce d'une forme quelconque n'est pas si simple à définir, en particulier quand la pièce est de forme concave ; par exemple, le centre de masse d'une pièce en L peut être à l'extérieur de la pièce (Figure 91).

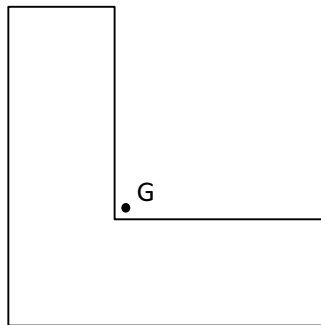


Figure 91. Le centre de masse G de la pièce en L est en dehors de la pièce

De plus, et comme le montre la figure suivante (Avec $D \approx 2 * d$), en partant d'un point "central", on prend le parti de négliger environ une demi-largeur de la pièce dans la longueur de l'itinéraire de cette pièce vers une autre. Dans le souci de ne pas sous-estimer ces longueurs nous n'avons pas retenu cette solution.

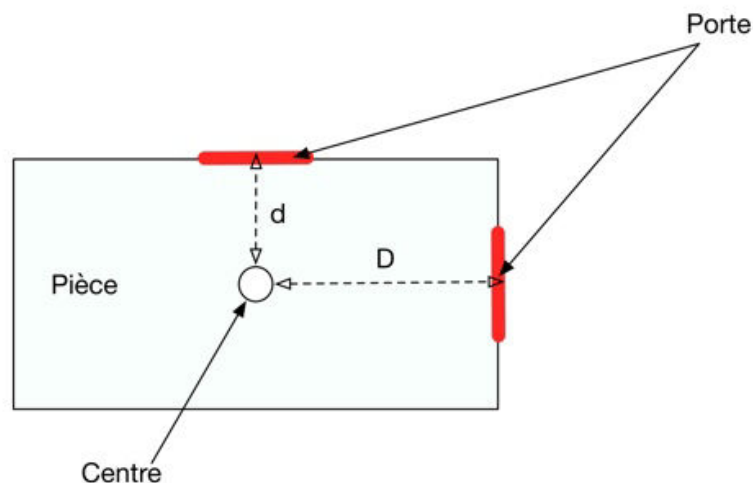


Figure 92. Exemple d'un point de départ situé au centre d'une pièce avec deux portes

Notre principal objectif dans le choix des points de départ est de ne pas sous-estimer la distance parcourue dans la pièce lorsque l'occupant la quitte pour une autre pièce. Ceci a orienté notre choix vers le point le plus éloigné de la ou des sorties. Ceci étant, quand il y a plus d'une porte, s'éloigner de l'une peut rapprocher de l'autre. Nous avons donc attribué à une pièce autant de points de départ qu'elle a de portes.

Pour une porte donnée, le point qui lui est associé est le point le plus éloigné parmi tous les points de la pièce qui prendront cette porte comme une porte de sortie, autrement dit le plus éloigné des points

plus proche de cette porte que de toute autre porte. La Figure 93 illustre de manière *approximative* cette partition dans le cas d'une pièce à 2 portes : la zone foncée est l'ensemble des points plus proches de la porte du haut que de la porte de droite ; le point de départ pour la porte du haut est Dep_1 , le plus éloigné de ces points.

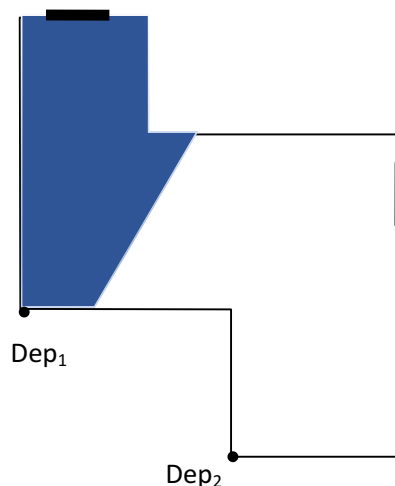


Figure 93. Les points de départ d'une pièce à deux portes

Le terme « approximatif » est mis en évidence dans la phrase qui précède car, pour calculer les points de départ, nous avons, dans le but de réduire les temps de développement, utilisé une recherche approchée. Avant de la présenter, notons que les points situés dans une bande de largeur R le long du pourtour de la pièce ne peuvent être atteints par le centre du cercle qui approche l'occupant et, donc, ne peuvent être des points de départ ; ils doivent être exclus de la recherche ; les points Dep_1 et Dep_2 de la figure précédente sont donc *faux*. Cette bande interdite est limitée intérieurement par un polygone « parallèle » au pourtour de la pièce : l'*offset*. Le Chapitre II donne des éléments sur la méthode de calcul de cet *offset*. Il reste à choisir, dans la zone du plan incluse dans l'*offset*, les points qui seront les points de départ. Nous avons conjecturé que ces points se situent nécessairement sur l'*offset* et c'est pour les trouver que nous utilisons une méthode approchée : des points O_i sont placés à intervalle régulier sur l'*offset*, à 50 cm les uns des autres dans le prototype logiciel (Figure 94).

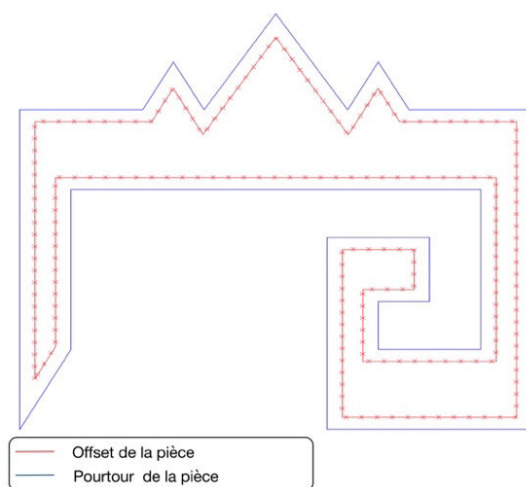


Figure 94. Offset d'une pièce

Pour chaque point O_i , on calcule sa distance à chaque porte et on en déduit de quelle porte il est le plus proche. Ce calcul de distance, donc de plus court chemin, est possible car, à ce stade, les points caractéristiques des portes et des sommets concaves sont déjà calculés. Un graphe des déplacements possibles *dans la pièce* est donc déjà disponible (Figure 95.a). Les nœuds de ce graphe sont les points caractéristiques - des portes et des points concaves - et il y a un arc entre deux nœuds si et seulement si le segment qui relie les points correspondants est à une distance supérieure à R de tous les murs de la pièce, ce qui implique aussi qu'il ne les coupe pas. Cette contrainte qu'un « segment de déplacement doit être à une distance supérieure à R de tous les murs » est nécessaire pour que l'occupant qui suivrait ce segment ne se heurte pas aux murs.

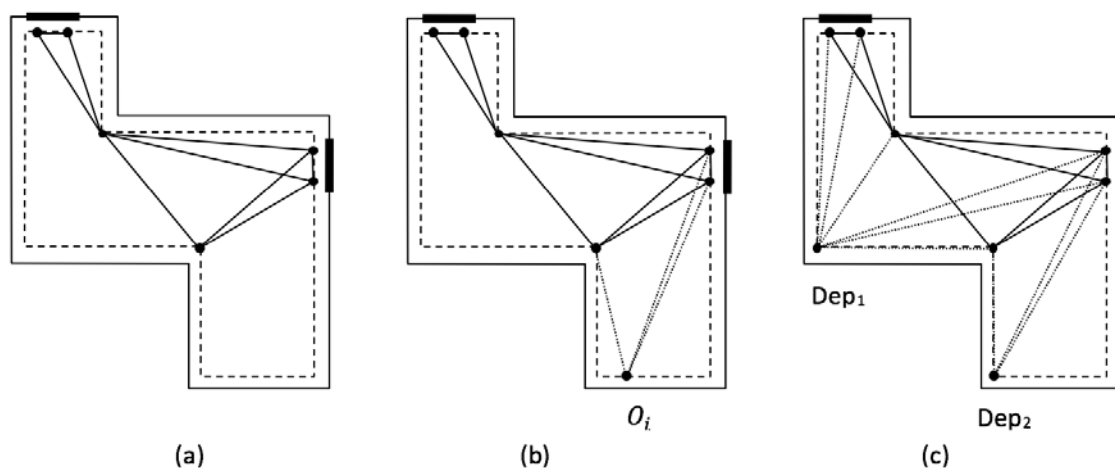


Figure 95. (a) Le graphe des déplacements possibles entre points caractéristiques d'une pièce et (b) les ajouts à ce graphe pour le calcul de la porte la plus proche d'un point O_i de l'offset

Il suffit de compléter ce graphe en y ajoutant temporairement un sommet pour le point O_i et des arcs de O_i vers les points caractéristiques qu'il est possible d'atteindre avec un mouvement rectiligne (il y en a trois dans l'exemple de la Figure 14.b) ; il est alors possible de calculer le plus court chemin de O_i vers chaque porte pour savoir à quelle porte il faut l'associer (celle dont il est le plus proche). O_i et les arcs incidents peuvent alors être supprimés du graphe. Ensuite, si O_i était plus éloigné de sa porte que l'actuel point associé à cette porte, il remplace ce point. De proche en proche, on obtient ainsi, pour chaque porte, le plus éloigné des points les plus proches de cette porte que de toute autre porte. On ajoute alors ces points de départ au graphe des déplacements possibles dans la pièce, ainsi que les arcs qui les connectent aux autres sommets (Figure 14.c). Du fait de l'échantillonnage, on sait que ces points de départ sont proches des points de départ théoriques, à défaut de leur être exactement égaux, mais les erreurs induites sur les longueurs des itinéraires entre pièces sont suffisamment faibles pour que cela ne soit pas préoccupant.

Les exemples qui précèdent montrent des points de départ situés sur des *sommets* de l'offset. Ce n'est cependant pas systématique. Dans la Figure 96, on constate que les points de départ sont cette fois strictement *dans un segment* de l'offset.

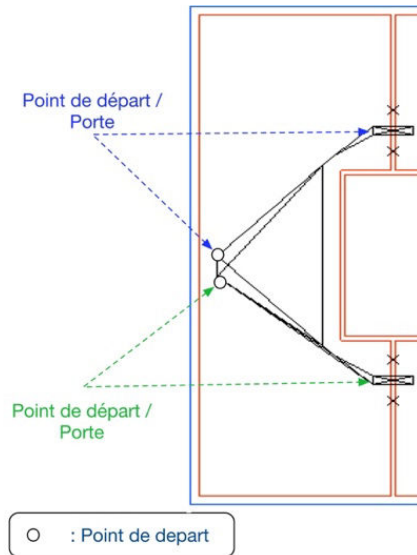


Figure 96. Des points de départ qui ne sont pas des sommets de l'offset (l'offset n'est pas représenté)

2.3.3. Graphe de mobilité complet de l'aménagement

Le graphe de mobilité du logement complet est l'assemblage des graphes de mobilité de toutes les pièces. L'assemblage est fait en ajoutant des arêtes entre les quatre points caractéristiques de chaque porte, ce qui représente potentiellement quatre arêtes par porte. La figure ci-dessous montre ce graphe pour un logement de six pièces.

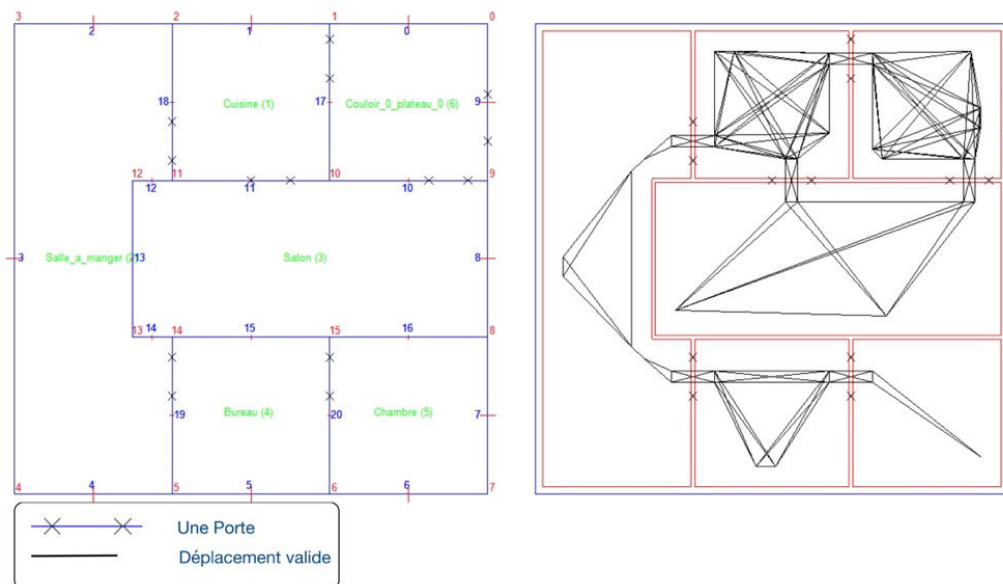


Figure 97. Graphe de mobilité d'un appartement de six pièces, pour une personne d'un rayon de 0.5 m le rayon de franchissement étant de 0.35 m

2.4. Graphe d'adjacence

Le graphe d'adjacence informe sur les possibilités de passer d'une pièce à une autre en franchissant une porte. Il ne s'agit donc pas de l'adjacence au sens où elle est utilisée dans le reste de ce manuscrit, à savoir le partage d'au moins un mur entre deux pièces : il faut en plus qu'il y ait au moins une porte sur l'un des murs partagés. La Figure 98 (droite) donne un exemple de graphe d'adjacence.

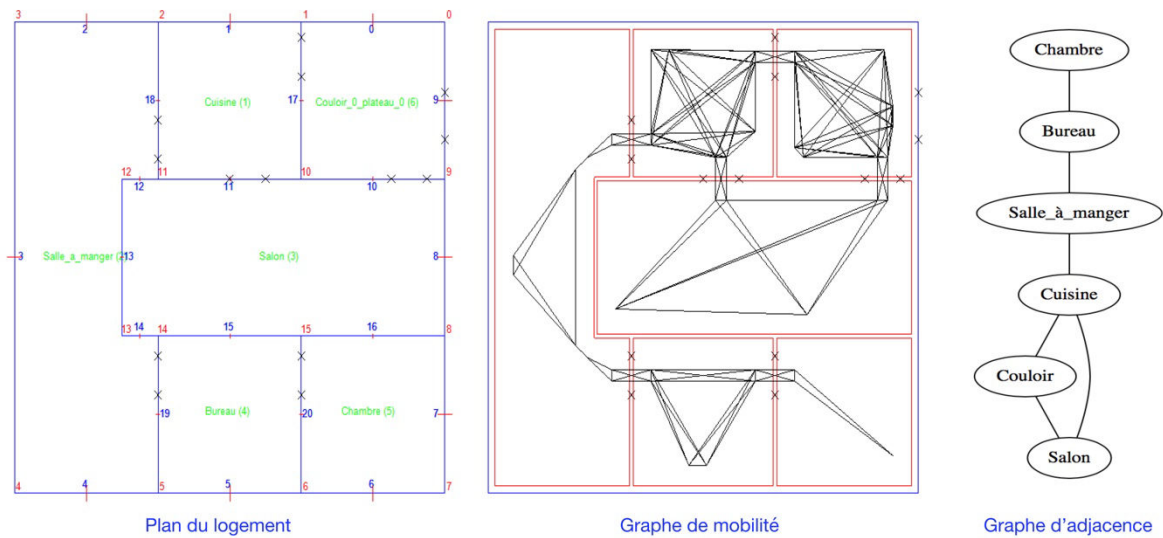


Figure 98. Exemple d'un graphe d'adjacence

Dans ce graphe, chaque nœud correspond à une pièce et il y a une arête entre deux nœuds si les deux pièces correspondantes partagent au moins une porte. Il est obtenu très simplement avec l'algorithme suivant :


```

Fonction ConstructionGrapheAdjacence (L) : Graphe G

// L est la structure de données du logement.

Début

Créer un nœud par pièce

    Pour chaque Pièce P1 de L

        Pour chaque mur M de la pièce P1

            Si le mur M contient au moins une porte et la pièce P2
de l'autre côté de la porte n'est pas la pièce factice qui entoure
le logement alors

                Ajouter l'arc (P1, P2) au graphe G s'il n'existe pas
déjà.

            FSi

        FPour

    FPour

Fin

```

Algorithme 4 Procédure de construction du graphe d'adjacence

3. Evaluation d'un aménagement

La conception d'un outil informatique capable de générer de nombreux aménagements et de sélectionner le(s) meilleur(s) nécessite d'une part une modélisation des considérations légales et d'autre part une modélisation des besoins et souhaits de l'occupant. Cette modélisation doit permettre d'exprimer toutes ces exigences en un langage exploitable d'un point de vue informatique, afin de pouvoir évaluer automatiquement le degré d'adaptation d'un aménagement à une personne donnée.

Dans notre méthode, cela consiste à traduire toutes les exigences en ce que nous appelons des *critères*. Ces critères peuvent être instanciés avec différents paramètres autant de fois que cela est nécessaire.

On distingue deux types de critères : les premiers traduisent une réglementation, une contrainte géométrique incontournable ou une contrainte forte imposée par l'état de santé du futur occupant (ce que nous appelons un besoin) ; ils doivent absolument être satisfaits. La note qui leur est attribuée est 0 ou 1. Les autres critères obtiennent une note graduée allant de 0 (critère non satisfait) à 1 (critère entièrement satisfait).

Il n'y a nullement une bijection entre les critères et les exigences évoquées plus haut (contraintes réglementaires, besoins et souhaits du futur occupant). Certains critères servent par exemple à la fois à la traduction d'une règle légale, d'un besoin et d'un souhait. Par exemple, le critère qui impose que « l'aire d'une pièce donnée doit être supérieure à une valeur donnée » sert à la fois à traduire la

contrainte *légal*e qu'une chambre doit faire au moins 9 m² pour mériter ce nom, le *besoin* d'avoir des toilettes d'au moins 4 m² pour que la circulation d'un fauteuil soit possible et le *souhait* du futur occupant d'avoir un salon d'au moins 25 m². De plus, un besoin pour certains peut-être un souhait pour d'autres et inversement. Par exemple, pour une personne âgée, il est *souhaitable* qu'elle fasse des efforts afin de maintenir une bonne autonomie, en revanche, pour une personne atteinte d'une dystrophie musculaire, cela est un *besoin* vital pour ralentir l'évolution de sa maladie. Le critère « assurer une certaine activité physique quotidienne » peut donc être tantôt associé à un besoin tantôt à un souhait.

L'évaluation des critères repose sur des opérateurs de plus bas niveau (comme l'aire d'une pièce ou le plus court chemin entre deux points du logement) [10]. Ces opérateurs utilisent les trois structures de données évoquées auparavant :

- la description géométrique 2D du logement à évaluer ;
- un graphe de mobilité, qui inventorie les déplacements rectilignes possibles à l'intérieur des pièces et à travers les portes ;
- un graphe d'adjacence, qui recense les connexions entre les pièces (par une porte, un escalier, un ascenseur ...) ; le graphe d'adjacence est un sous-graphe du graphe de mobilité.

En conséquence, la présentation qui suit de la méthode d'évaluation d'un logement est articulée comme suit : on présente d'abord les opérateurs de bas niveau utilisés dans l'évaluation des critères. On présente ensuite tous les critères actuellement disponibles et la manière dont ils sont évalués. On conclut en expliquant comment la note du logement est obtenue.

3.1. Les opérateurs de bas niveau pour la vérification des critères de validité

Les opérateurs qui relèvent de la géométrie ont déjà été évoqués en détail dans le Chapitre II. Ils ne sont pas rappelés ici. Il s'agit de :

- appartenance d'un point à un polygone ;
- aire d'un polygone ;
- angle intérieur au niveau d'un sommet d'un polygone ;
- inclusion d'un polygone dans un autre polygone ;
- orientation d'une pièce face à telle direction ;
- calcul du rapport d'aspect d'un polygone.

Nous n'évoquerons pas non plus les opérateurs qui relèvent d'une simple consultation de la structure de données. Il est question de vérifier :

- l'existence d'une ouverture donnée sur un mur ;
- l'existence d'une pièce sur un plateau d'un logement ;
- si un mur est extérieur (et susceptible, donc, de contenir une fenêtre).

Les autres opérateurs utiles sont les suivants.

3.1.1. Longueur d'un segment

La longueur d'un segment AB est calculée à l'aide de la formule suivante :

$$AB = \sqrt{(X_b - X_A)^2 + (Y_b - Y_A)^2}$$

3.1.2. Intersection de deux segments

Deux segments sont sécants quand leurs deux droites sont sécantes en un point appartenant à ces deux segments. D'un point de vue calculatoire, il suffit de résoudre le système d'équations constitué des équations des deux droites pour calculer les coordonnées du point d'intersection et de vérifier ensuite que le point appartient aux deux segments en comparant ses coordonnées à celles des deux extrémités de chaque segment.

3.1.3. Connexité d'un graphe non orienté

Un graphe non orienté G est connexe si et seulement si, pour chaque paire de sommets (u, v) de G , il existe une suite d'arrêtes permettant d'atteindre v à partir de u . Soit n le nombre de sommets et m le nombre d'arrêtes. Cette propriété peut être vérifiée à l'aide de plusieurs algorithmes comme : l'algorithme de *Warshall* - à la base pour la construction de la fermeture transitive d'un graphe -, sa complexité est de $O(n^3)$ [83]; l'algorithme de *Prim* - pour le calcul d'un arbre couvrant minimal -, sa complexité est de $O(n^2)$ [67] ; ou bien des algorithmes basés sur le parcours de graphe comme l'algorithme de parcours en largeur ou l'algorithme de parcours en profondeur dont la complexité est $O(n + m)$ [18].

Les algorithmes les plus utilisés pour ce traitement sont les deux algorithmes de la famille des algorithmes de parcours d'un graphe. Dans les deux cas, on parcourt le graphe à partir d'un sommet quelconque et, quand le parcours est terminé, on vérifie que le nombre de sommets atteints est égal au nombre total de sommets.

Dans notre cas, nous avons choisi l'algorithme de parcours en profondeur.

3.1.4. Points d'articulation d'un graphe

Un point d'articulation d'un graphe non orienté est un sommet dont la suppression (ainsi que celle de ses arêtes) déconnecte le graphe. La détection des points d'articulation étant moins simple qu'un parcours en profondeur, on en rappelle les principaux éléments.

On détecte les points d'articulation en parcourant le graphe en profondeur à partir d'un quelconque sommet et en attribuant à chaque sommet non encore rencontré un numéro croissant. De plus, au niveau de chaque sommet x , on maintient un entier Min_x qui est restitué quand le parcours de x est terminé. Min_x est initialisé au numéro du sommet et est finalement le minimum des numéros des voisins déjà marqués de x et des résultats obtenus dans les parcours en profondeur à partir des voisins non marqués. De manière imagée, on peut dire que Min_x est le numéro du plus "petit" sommet touché pendant le parcours en profondeur à partir de x . Ainsi, quand x a un voisin non marqué v , on fait un parcours récursif à partir de v . Appelons Min_v le résultat de ce parcours. Si Min_v est inférieur au numéro de x , c'est qu'il existe une connexion entre l'un des sommets situés sous v (v compris) et un ancêtre de x ; x n'est donc pas un point de passage obligé pour aller de ses ancêtres à ses descendants de la branche de v . Sinon, Min_v est nécessairement égal au numéro de x , ce qui signifie que la connexion évoquée ci-dessus n'existe pas et que x est un point d'articulation. Le cas de la racine du parcours (le tout premier sommet) est une exception. En résumé, on a :

x est un point d'articulation si et seulement si :

- il est le sommet de départ (son numéro est 1) et il a plus d'un fils dans sa descendance
ou
- il n'est pas le sommet de départ et l'un au moins des parcours effectués à partir de ses voisins non marqués a retourné le numéro de x .

3.1.5. Le plus court chemin entre deux pièces

Comme son nom l'indique, cet opérateur a pour objectif de retourner le plus court chemin entre deux pièces. La procédure utilisée ici est inspirée de la procédure de calcul du plus court chemin que l'on retrouve dans le domaine de la robotique [20]. Le traitement consiste à retourner le plus long parmi tous les plus courts chemins entre chaque paire de points de départ des deux pièces. Le plus long des plus courts chemins est privilégié pour éviter de sous-estimer la distance entre les pièces. Un plus court chemin entre deux points caractéristiques est calculé en identifiant les sommets correspondants dans le graphe de mobilité et en utilisant l'algorithme de Dijkstra [18].

Dans tout ce qui suit, quand nous évoquons le plus court chemin entre deux pièces, il faut donc bien garder à l'esprit qu'il s'agit du plus long de plusieurs plus courts chemins.

Evidemment, les plus courts chemins sont calculés une fois pour toutes, au début de l'évaluation d'un aménagement.

3.2. Les critères d'évaluation

Pour structurer la présentation des critères, nous les avons répartis en trois ensembles : les critères réglementaires et géométriques, dont l'évaluation est nécessairement binaire, les critères topologiques, en rapport avec la disposition et les dimensions des pièces, portes et couloirs, et les critères sur les itinéraires. Les critères des deux dernières catégories peuvent être évalués de manière binaire ou graduée, selon qu'ils traduisent un besoin ou un souhait.

Quand l'évaluation d'un critère repose sur un opérateur, nous mentionnons l'opérateur.

3.2.1. Critères réglementaires et géométriques

3.2.1.1. *Non chevauchement des pièces*

Un logement valide géométriquement est un logement dont les pièces ne se chevauchent pas. Nous verrons que les modifications appliquées au logement ne permettent pas que les pièces se coupent ; en conséquence, deux pièces sont soit disjointes, soit incluses l'une dans l'autre, c'est pourquoi l'évaluation de ce critère est faite à l'aide de l'opérateur d'inclusion d'un polygone dans un autre. Ce critère est toujours évalué de manière binaire.

3.2.1.2. *Aire minimale de la pièce principale*

Un logement doit avoir au moins une pièce principale (séjour, chambre) ayant une surface minimale habitable de $9m^2$ (article 4 du décret n°2002-12° du 30 janvier 2002).

Pour évaluer ce critère, il suffit de vérifier qu'au moins une pièce principale atteint l'aire minimale à l'aide de l'opérateur « *aire d'un polygone* ». Ce critère est aussi binaire.

L'opérateur sous-jacent est le calcul de l'aire d'un polygone.

3.2.1.3. *Existence d'une pièce*

Aux yeux de la loi, un logement, pour qu'il soit décent, doit être doté d'une cuisine, d'une salle de bain, d'une pièce principale, ... ce qui signifie que la présence de certaines pièces est obligatoire ; d'où un critère *binaire* qui vérifie l'existence d'une pièce et que l'on instancie autant de fois qu'il y a de pièces obligatoires.

Il n'y a pas à proprement parler d'opérateur sous-jacent pour ce critère, il s'agit simplement d'accès à la structure de données du logement.

3.2.1.4. Pièce non borgne

L'article 2.6 du décret n°2009-52 du 15 janvier 2009 stipule que les pièces principales (séjour, chambre) doivent bénéficier d'un éclairage naturel suffisant et d'un ouvrant donnant à l'air libre ou sur un volume vitré donnant à l'air libre. Cela suppose que les pièces principales ne doivent pas être borgnes et elles doivent donner sur l'extérieur.

Nous proposons donc un critère *binnaire* qui vérifie à l'aide de l'opérateur « *Mur extérieur* » qu'au moins un des murs d'une pièce donnée donne sur l'extérieur et, donc, qu'il peut contenir une fenêtre. Le placement exact de la fenêtre sur le mur est laissé à la charge de l'architecte car la position n'a aucun impact sur la note de ce critère. Ce critère est instancié sur chaque pièce considérée comme principale par la loi.

3.2.1.5. Largeur minimale des portes

La largeur minimale d'une porte dans un établissement recevant du public (ERP) doit varier entre 0.8 et 1.4m selon que qu'elle permet d'accéder aux locaux ou aux sanitaires et selon la capacité d'accueil de l'établissement (Article 10.2, Arrêté du 1^{er} Aout 2006). Pour un logement, les portes sont standardisées selon la norme (NF P 01 005), les largeurs peuvent être : 0.63, 0.73, 0.83 ou 0.93m. Nous avons donc un critère qui vérifie que toutes les portes ont une largeur valide.

La longueur d'une porte est obtenue par une requête envoyée à la structure de données. Soit L la longueur du mur qui contient la porte et deb et fin sont les indices de début et de fin de la porte ($0 \leq deb, fin \leq 1$) alors la largeur de la porte est $|fin - deb| * L$.

3.2.1.6. Aménagement accessible à une personne de corpulence dite normale

Pour être déclaré légal, un aménagement se doit d'être partout accessible à une personne de corpulence dite normale. S'il est impossible d'aller d'un point A à un point B car tous les itinéraires sont invalides du fait qu'une porte ou un couloir sont trop étroits alors l'aménagement est jugé inaccessible. Un aménagement est accessible, s'il existe au moins un itinéraire valide pour aller de n'importe quel point à n'importe quel autre point.

Il y a donc un critère binaire, dit de connexité, dont le rôle est de faire cette vérification.

Ce critère est indispensable car le logiciel va étudier de très nombreuses configurations, dont certaines sont parfaitement irréalistes (Figure 99).

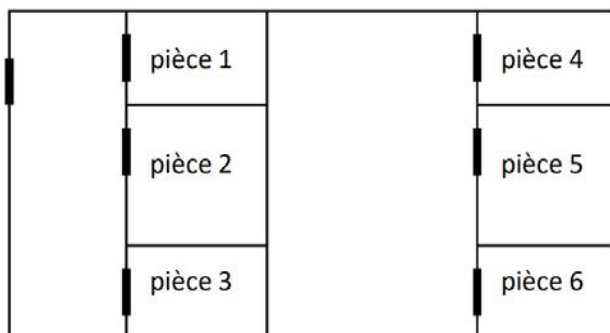


Figure 99. Les pièces 4,5,6 ne sont pas accessibles depuis la porte d'entrée

Ce critère est évalué à l'aide de l'opérateur « *connexité d'un graphe non orienté* » qui vérifie si le graphe de mobilité associé à une personne de corpulence dite normale (un graphe de mobilité généré avec un rayon de l'habitant estimé à 0.3 m) est connexe ou non. Si le graphe est connexe alors l'aménagement est accessible et la note est 1. Sinon elle vaut 0.

3.2.1.7. Zone d'intérêt

Le placement de la cuisine doit prendre en considération les endroits où se trouvent les arrivées et les évacuations des eaux, celui du salon doit prendre en compte l'emplacement d'un éventuel conduit de cheminée. Ces exemples illustrent la nécessité d'un critère qui vérifie qu'une certaine zone, de position imposée, est incluse dans une pièce.

Ces zones sont appelées *zones d'intérêt*. Chaque zone est approchée par un rectangle qui englobe la surface réelle de la zone d'intérêt. L'évaluation du critère consiste donc à vérifier que le rectangle est entièrement inclus dans le polygone de la pièce souhaitée.

L'inclusion est vérifiée en appelant quatre fois l'opérateur de bas niveau « *point appartenant à un polygone* » pour tester si chacun des sommets du rectangle est inclus dans le polygone de la pièce souhaitée. Si un des sommets n'est pas à l'intérieur du polygone, la zone d'intérêt est jugée à l'extérieur de la pièce et le critère est évalué à 0. Sinon, bien qu'on n'ait pas la garantie que la zone d'intérêt soit alors entièrement incluse dans la pièce, la note est 1.

3.2.2. Les critères topologiques

3.2.2.1. Existence d'une pièce

Ce critère *binnaire*, qui vérifie qu'une pièce précise (chambre à coucher, cuisine, salle de bain etc) est présente sur un niveau donné, a déjà été présenté dans la rubrique des critères légaux. Nous l'évoquons à nouveau pour signaler qu'il peut aussi exprimer des besoins ou des souhaits. Par exemple, pour une personne en fauteuil roulant, il est judicieux de garantir que toutes les pièces indispensables soient présentes sur un même niveau et de préférence au rez-de chaussée ; on instancie alors ce critère une fois pour chaque pièce indispensable.

3.2.2.2. Aire d'une pièce

Nous avons présenté plus haut une version binaire d'un critère assurant qu'une pièce a au moins une certaine surface ; il s'agissait là d'une disposition *réglementaire*, d'où son aspect tout ou rien.

Voici à présent d'autres critères sur les aires, cette fois gradués : minimisation, maximisation (avec ou sans seuil) et encadrement. On peut par exemple formuler le besoin que la salle de bain ait une aire minimum de 10 m² ou le souhait qu'une chambre à coucher ait une aire entre 12 m² et 15 m², pour permettre une meilleure circulation à l'intérieur de chacune de ces pièces. On peut aussi souhaiter que le salon ait une aire de 25 m² au maximum. L'opérateur sous-jacent, le calcul de l'aire d'un polygone, a été détaillé dans le Chapitre II.

Quand le critère fait intervenir un seuil, il est évalué avec les courbes linéaires par morceaux suivantes :

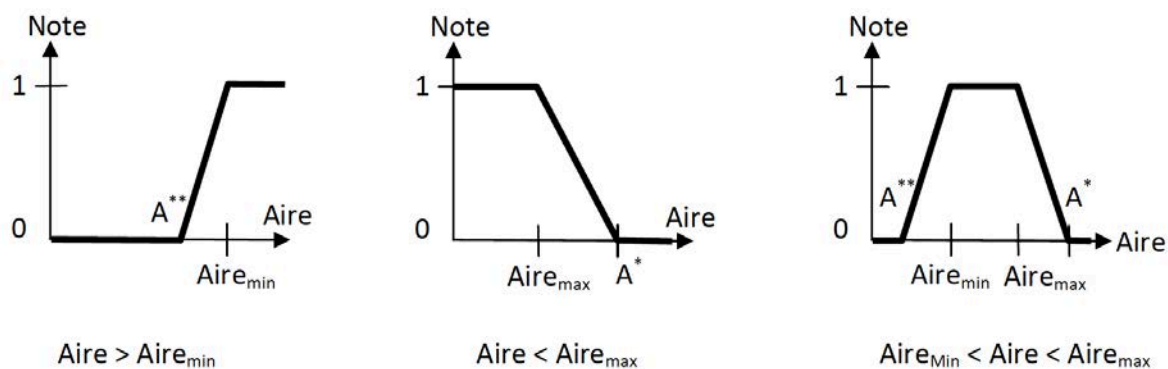


Figure 100. Les trois courbes linéaires par morceaux utilisées pour évaluer le critère avec un seuil

Dans le premier et le troisième cas, A^{**} est la surface à partir de laquelle la pièce mérite de recevoir une note non nulle. Il s'agit d'une valeur très subjective : si on veut un salon d'au moins 20 m^2 , un salon de 19 m^2 ne mérite probablement pas une note nulle, c'est pourquoi nous n'avons pas envisagé une fonction en escalier. Se pose alors la question de savoir en deçà de quelle surface la note doit être nulle. Nous avons arbitrairement initialisé A^{**} à 80 % de $Aire_{Min}$. Un raisonnement semblable s'applique à A^* , à concurrence de l'aire du logement.

Quand le critère ne fait pas intervenir un seuil (« je veux le salon le plus grand possible » et non pas « je veux un salon de plus de 20 m^2 »), les fonctions d'évaluation restent linéaires mais sont un peu simplifiées (Figure 101).

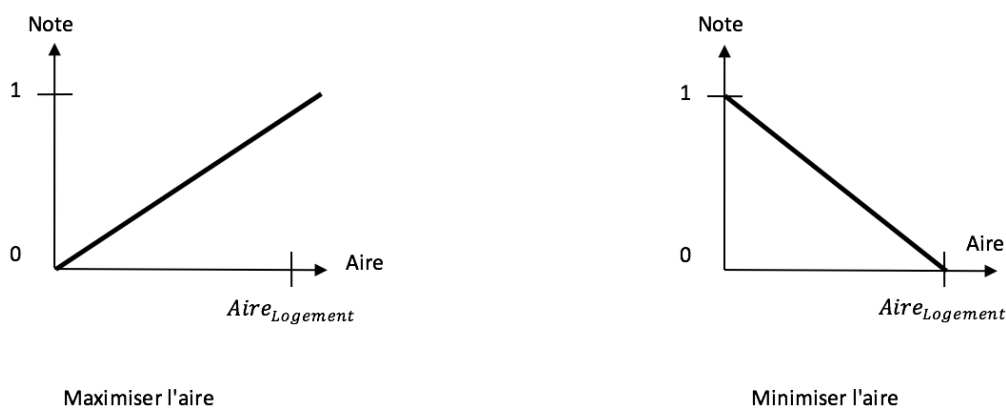


Figure 101. Les courbes des fonctions d'évaluation simplifiées

Quand un critère de maximisation de l'aire est utilisé, la surface de la pièce tend à croître au détriment de celles des autres pièces, ces dernières voyant leurs dimensions devenir déraisonnablement petites. Pour contrebalancer ce comportement, deux attitudes sont possibles : la première est d'imposer des critères de minoration à toutes les autres pièces, pour empêcher leurs surfaces de passer sous certaines valeurs ; le renseignement de ces valeurs minimales ne constitue pas un effort supplémentaire pour l'utilisateur car il les fournit généralement de manière spontanée, quand elles ne sont pas imposées par la législation (voir 3.2.1.2). La deuxième attitude possible est de calculer l'aire raisonnable maximum (ARM_a) de la pièce contrainte et de lui attribuer la note maximale dès l'instant que cette valeur est atteinte. La courbe d'évaluation devient alors celle de la Figure 102 (gauche). ARM_a est obtenue en soustrayant à l'aire du logement les aires minimum de l'ensemble des autres

pièces. La deuxième approche a le mérite de ne pas multiplier le nombre de critères, ce pour quoi nous la préférons.

Un raisonnement semblable s'applique aux minimisations, avec cette fois une aire raisonnable Minimum (ARM_i) ; la courbe d'évaluation devient celle de la Figure 102 (droite).

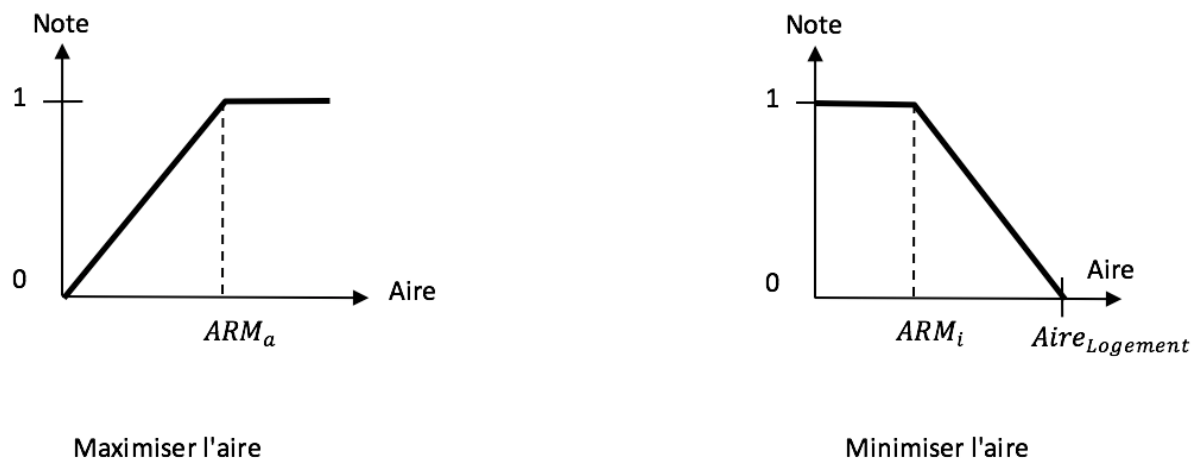


Figure 102. Les deux courbes d'évaluation avec un seuil raisonnable maximum

3.2.2.3. Orientation d'une pièce dans une direction donnée

Il ne s'agit pas du critère de même nom évoqué dans la synthèse d'un aménagement : dans ce chapitre, l'orientation traduisait le fait qu'une pièce rectangulaire était, ou pas, tournée de 90°. Cette fois, il s'agit de stipuler que la pièce doit faire face à une direction donnée. La direction est spécifiée par un angle (0° pour l'Est, 45° pour le Nord-Est etc).

Un critère symétrique spécifie qu'une pièce *ne doit pas* faire face à une direction donnée.

Ce sont des critères qu'on retrouve souvent lors de la définition des besoins et souhaits de l'occupant. Par exemple, pour une personne dont le sommeil est fragile, il est souhaitable que la chambre à coucher ne soit pas du côté de la rue passante ou face à l'est ; symétriquement, pour faire des économies d'énergie dans les pays à climat froid ou tempéré, il faut que les pièces les plus occupées du logement soient du côté le plus ensoleillé.

L'opérateur support de ce critère a été présenté dans le Chapitre II.7.

3.2.2.4. Pièce non borgne

Il y a des pièces dans un logement qui ne peuvent pas être borgnes au sens de la loi (article 2.6 du décret n°2009-52 du 15 janvier 2009) et nous avons vu qu'un critère binaire permet de vérifier le respect de cette loi. Certaines autres pièces ne font l'objet d'aucune contrainte légale sur ce point mais, souvent, les gens souhaitent qu'une salle de bains ait une fenêtre pour qu'il soit plus facile de l'aérer. De même, les utilisateurs réguliers d'un bureau préfèrent généralement bénéficier de la lumière du jour pour travailler et ont besoin de l'aérer, deux raisons de souhaiter que le bureau ne soit pas borgne. Pour traduire ces *souhaits*, on utilise le même critère que celui qui est rappelé au début du paragraphe.

3.2.2.5. Porte commune à deux pièces

Une demande récurrente dans les logements récents est qu'une chambre à coucher dispose d'un accès direct à une salle de bains ce qui revient à vérifier que les deux pièces ont une porte commune.

Pour un occupant lourdement handicapé, il est souhaitable, voire même nécessaire, que sa chambre partage une porte avec la chambre de l'auxiliaire de vie afin de garantir des interventions rapides.

Ce sont deux raisons de prévoir un critère qui garantisse l'existence d'une porte sur un mur partagé par deux pièces. Il s'agit d'un critère binaire : la porte existe ou pas ; en conséquence, sa note est 0 (porte absente) ou 1. Un 0 n'implique pas nécessairement que la note du logement soit nulle aussi, cela dépend du caractère impératif ou facultatif de l'exigence traduite par le critère.

3.2.2.6. Compacité / Dispersion des pièces

Ce critère a pour objectif d'agir sur l'ensemble des pièces pour les éloigner ou les rapprocher les unes des autres (au sens de la distance qu'il faut parcourir pour aller de l'une à l'autre). Rapprocher les pièces permet de réduire la quantité d'efforts à fournir par l'occupant, les éloigner, en revanche, contraint à davantage d'activité physique. Dans les deux cas, on agit sur la somme des longueurs des itinéraires entre tous les couples de pièces ; dans un cas, on la minimise, dans l'autre, on la maximise.

La mesure de la compacité d'un logement s'appuie sur le diamètre du logement et sur le plus long des plus courts chemins entre deux pièces. Le diamètre a été défini dans le Chapitre II : c'est la plus longue distance entre deux sommets du pourtour du logement.

Nous considérons qu'un itinéraire entre deux pièces est relativement court s'il est inférieur au demi-diamètre. En conséquence, si le plus long des chemins est inférieur au demi-diamètre, on a l'assurance que tous les chemins sont relativement courts et que le logement est compact : sa note au critère de compacité est alors de 1. Au-delà, la note diminue et atteint 0 quand la longueur du plus long plus court chemin est égale au diamètre. L'évaluation de la compacité est donc réalisée avec la courbe suivante :

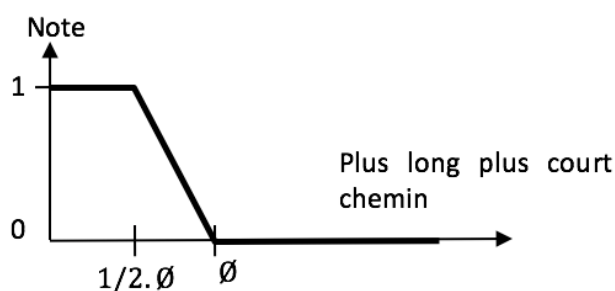


Figure 103. La courbe de la fonction d'évaluation de la compacité

3.2.2.7. La maîtrise du coût de construction

Un accident de la vie, peu d'économies, peu de subventions ou juste le souhait de ne pas dépasser un budget fixé impliquent que le coût des travaux doit être maîtrisé, d'où un critère qui le vérifie.

Pour estimer le coût d'un aménagement, nous avons repris le même principe proposé par B.Medjdoub et B.Yannou [56], repris aussi par A.Puret [68], et qui consiste à ramener le coût d'un aménagement à la longueur de ses murs.

Le coût d'un aménagement est la somme des longueurs des murs intérieurs (on exclut donc les murs de la périphérie du plateau). La longueur Lm_i du mur i est calculée à l'aide de l'opérateur « calculer la longueur d'un segment ». Ainsi, le coût $Coût$ d'un aménagement est exprimé comme suit

$$Coût = c * \sum_{i=1}^M Lm_i$$

avec M : des murs intérieurs et c : le coût de la construction d'un mètre de mur.

La note renvoyée par ce critère est 0 si le coût dépasse le budget Max fixé par l'occupant. La note est 1 si le coût ne dépasse pas le budget souhaité par l'occupant. L'évaluation du coût est réalisé avec la courbe suivante :

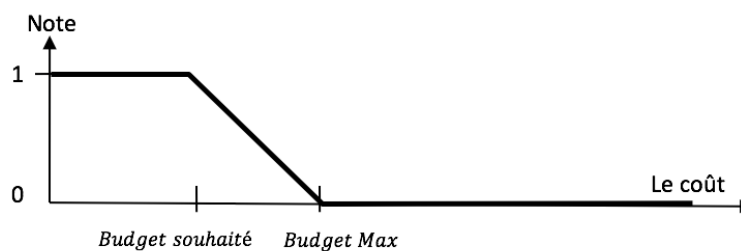


Figure 104. La courbe de la fonction d'évaluation du coût de l'aménagement

3.2.2.8. Forme raisonnable d'une pièce

Les pièces doivent avoir des formes raisonnables pour que l'on puisse considérer l'aménagement comme un aménagement réaliste.

Le critère qui vérifie cette exigence pour une pièce donnée repose sur le rapport d'aspect, qui est présenté dans le chapitre des algorithmes géométriques. Ce rapport, pour un carré, est de 1. Mais le rapport peut aussi être inférieur à 1 : si la forme de la pièce s'approche d'un cercle, le rapport d'aspect tend vers $\frac{1}{2} \cdot \sqrt{\pi}$ (le rapport d'aspect d'un cercle) soit environ 0.89. Depuis la preuve du théorème d'isopérimétrie, à la fin du XIX^{ème} siècle [80], on sait que cette valeur est un minimum. Nous considérerons que des formes voisines du cercle et du carré sont très satisfaisantes et, donc, que des rapports d'aspects d'une valeur inférieure ou égale à 1 doivent être évalués avec la meilleure note possible (1).

Un rectangle cinq fois plus large que haut a un rapport d'aspect de $3/\sqrt{5}$, ce que nous considérons comme une limite supérieure tolérable.

En conséquence, la note obtenue par une pièce pour ce critère est définie par la courbe suivante :

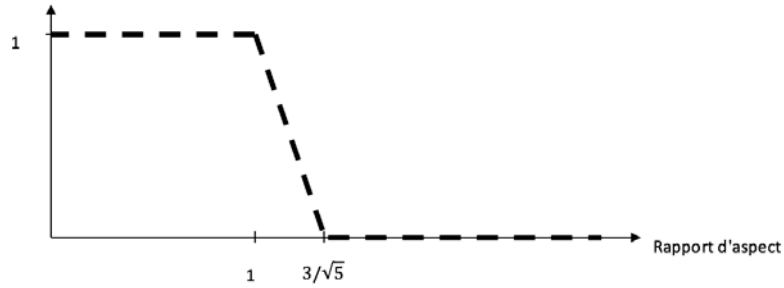


Figure 105. La courbe de la fonction d'évaluation de la forme raisonnable

3.2.2.9. La maîtrise des angles des pièces

Les modifications répétées des pièces pendant le processus d'optimisation tendent à leur donner des formes très irrégulières. Pour maîtriser ces déformations, nous proposons deux critères : « Forme raisonnable », qui fait l'objet du paragraphe précédent, et « Maîtrise des angles des pièces ». Ce critère reçoit, en entrée, une pièce et une séquence d'intervalles angulaires. Son résultat est la proportion de sommets dont l'angle intérieur à la pièce est inclus dans l'un des intervalles. Par exemple, si l'utilisateur du logiciel souhaite n'avoir que des angles droits, la séquence d'intervalles est $[90^\circ, 90^\circ]$, $[180^\circ, 180^\circ]$, $[270^\circ, 270^\circ]$. S'il ne veut que des pièces rectangulaires, la séquence se réduit à $[90^\circ, 90^\circ]$, $[180^\circ, 180^\circ]$. Dans les deux cas, il faut autoriser des angles plats puisque les murs, au sens de notre structure de données, sont subdivisés sur leurs intersections avec d'autres murs (Figure 106).

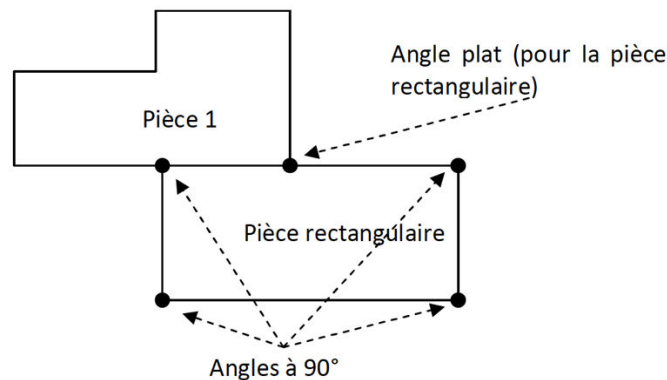


Figure 106. Une figure rectangulaire peut avoir plus de quatre murs

3.2.3. Les critères sur les itinéraires

3.2.3.1. Mobilité

Des personnes cardiaques, asthmatiques, atteintes de la maladie de Parkinson, ou d'une dystrophie musculaire... ont toutes un point commun : elles doivent gérer leurs efforts. Dans certains cas, il faut raccourcir les itinéraires pour limiter l'effort, autrement il y aura une augmentation des risques liés à la maladie. Pour une personne atteinte d'une dystrophie musculaire, l'idéal est d'allonger ses itinéraires pour la forcer à fournir le plus d'efforts possibles et ainsi ralentir sa perte musculaire. Nous proposons donc divers critères de maximisation/minimisation d'itinéraires, avec ou sans seuil ; nous proposons aussi des critères agrégés, pour assurer que la distance parcourue en une journée dans le logement est au moins ou au plus égale à une certaine valeur.

À titre d'exemple, nous présentons le critère de minimisation de la distance d'une pièce à une autre. Rappelons d'abord qu'il ne s'agit pas d'un critère symétrique : s'il y a des rampes ou des escaliers dans l'itinéraire de la première pièce à la deuxième, la distance est la même dans les deux sens mais l'effort est potentiellement *très différent*. Par exemple, une personne gravement asthmatique ne peut pas monter plus de quelques marches d'affilée, ce qui peut l'autoriser à faire l'aller mais lui interdire le retour par le même itinéraire. Bien que la notion d'effort ne soit pas intégrée dans la version courante, nous avons dès à présent pris en compte le fait que les itinéraires aller et retour entre deux pièces ne sont pas les mêmes à une inversion près.

Le critère de minimisation de la distance entre deux pièces accepte trois paramètres : les deux pièces concernées et, de manière facultative, la Distance Tolérable Maximum (DTM_a), au-delà de laquelle la note du critère est nulle. La courbe d'évaluation de ce critère est la suivante :

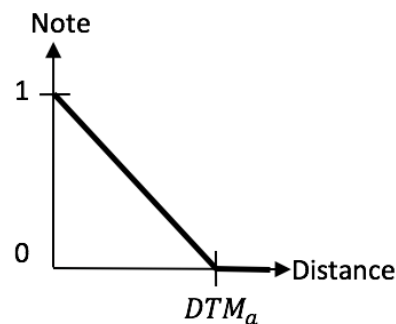


Figure 107. La courbe d'évaluation pour la minimisation de la distance d'une pièce à une autre

Si la Distance Tolérable Maximum n'est pas fournie, elle est fixée au double du diamètre du logement, le diamètre est considéré ici comme étant la plus grande distance, à vol d'oiseau, entre deux points de la périphérie du logement. Ce choix se justifie comme suit : il peut arriver que la distance marchée entre deux pièces soit supérieure au diamètre du logement, comme, par exemple, entre les pièces 1 et 2 dans la Figure 108, mais c'est rare :

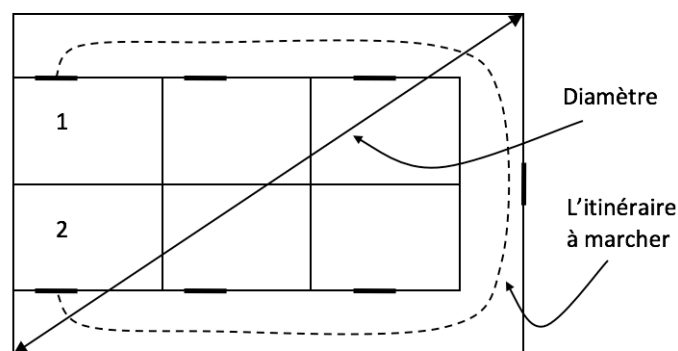


Figure 108. Exemple d'une distance marchée supérieure au diamètre du logement

A fortiori, il est très improbable que, dans un cas réaliste, l'itinéraire entre deux pièces soit plus de deux fois plus long que le diamètre. Le double du diamètre nous est donc apparu comme une borne supérieure satisfaisante dans le cas d'une minimisation.

3.2.3.2. Logement accessible à l'occupant

L'évaluation de ce critère est quasi similaire à l'évaluation du critère d'accessibilité d'un aménagement pour une personne de corpulence dite normale (évoqué dans les critères de validité réglementaire et géométrique d'un aménagement). Il y a cependant deux différences :

- le graphe de mobilité donné en paramètre est celui qui est associé à l'occupant et non celui qui correspond à une personne de corpulence dite normale ;
- la note renvoyée par ce critère est une note graduée et non binaire. Cette note est exprimée comme le rapport entre le nombre de pièces atteignables et le nombre total de pièces de l'aménagement. Elle est de 1 quand toutes les pièces sont atteignables et de 0 quand aucune ne l'est.

Notons que si certaines pièces inutiles à l'occupant sont inatteignables la notion d'accessibilité n'est pas remise en question. Il suffit d'évaluer ce critère avec un sous-graphe induit du graphe de mobilité et qui ne contient que les sommets associés aux pièces utiles pour donner une note plus juste à ce critère.

3.2.3.3. Enfilades

Une disposition avec des pièces en enfilade est souvent indésirable. Pour mesurer l'importance de ce critère, il suffit d'imaginer que pour aller d'une pièce à une autre, il faille par exemple traverser une chambre à coucher.

Le critère gradué « Enfilade » est évalué en deux phases : on calcule d'abord combien de pièces ne tolérant normalement pas le passage doivent obligatoirement être traversées pour aller d'une pièce à une autre ; ces pièces annoncent des enfilades et, par abus de langage, nous les appelons des enfilades. Appelons NE ce nombre et NP le nombre total de pièce. La note du critère enfilade est alors obtenue avec la formule suivante :

$$Note = (NP - NE)/NP$$

Autrement dit, s'il n'y a aucune pièce en enfilade, $NE = 0$ et la note est 1 ; au pire des cas, $NE = NP$ et la note est alors nulle.

Le calcul de NE est simple une fois que le constat suivant a été fait : pour qu'une pièce soit une enfilade, il faut avant toute chose qu'elle soit une pièce ne tolérant pas le passage. Il faut aussi que la suppression des portes dans cette pièce (ce qui revient à la supprimer du graphe d'adjacence) rende impossible l'accès à certaines pièces (ce qui revient à dire que le graphe d'adjacence n'est plus connexe après suppression de cette pièce). En d'autres termes, les pièces en enfilade sont les pièces qui ne tolèrent pas le passage et qui sont aussi des points d'articulation du graphe d'adjacence.

3.2.4. Modélisation informatique des critères

D'un point de vue informatique, chaque critère est une classe, toutes les classes héritent d'une même classe qui définit le comportement commun à toutes les filles, à savoir une fonction d'évaluation, dont les entrées sont le logement à évaluer et dont le résultat est une note entre 0 et 1.

3.3. Evaluation de l'adaptation d'un aménagement à un occupant

L'évaluation du logement est faite en deux temps : les critères traduisant des besoins impérieux (contraintes légales et géométriques, besoins du futur occupant) sont évalués d'abord et si l'un d'eux

n'est pas satisfait, l'évaluation est interrompue et retourne la note 0. Sinon, la note du logement est la moyenne des notes obtenues aux autres critères, qu'ils soient binaires ou gradués.

Deux précisions doivent être apportées : nous avons donné à l'utilisateur la possibilité de pondérer les critères, pour le cas où certains, sans être des couperets, auraient malgré tout une importance considérable. La moyenne évoquée ci-dessus est donc une moyenne pondérée. De plus, pour éviter l'effet compensatoire, on a instauré la possibilité d'ajouter pour chaque critère une note minimale ; si cette dernière n'est pas atteinte alors la note est réduite à 0. Ce seuil est fixé par l'occupant lors du diagnostic de ses besoins et souhaits.

4. Modification d'un aménagement

Le module de modification d'un logement offre un ensemble d'opérateurs permettant de modifier un aménagement pour créer des variantes. Il doit garantir l'existence d'au moins une série d'opérations pour atteindre tout aménagement à partir de n'importe quel autre. Cette condition est nécessaire car, sinon, des applications successives des modifications ne permettent d'explorer qu'une partie de l'espace des solutions et, donc, on court le risque de ne jamais rencontrer une solution optimale.

Notons, pour éviter toute confusion, que les opérateurs évoqués ici, les opérateurs de modification, ne sont pas les mêmes que ceux qui sont utilisés au moment de l'évaluation des critères, que nous avons appelés « opérateurs de base pour l'évaluation ».

Dans cette section, nous présentons en détail l'ensemble de ces opérateurs de modification puis nous donnons quelques éléments sur la partie de l'espace des solutions que ces opérateurs permettent de parcourir.

4.1. Les opérateurs de modifications de bas niveau

Les opérateurs agissent sur les pièces et les portes pour créer des variantes d'un aménagement. Ils permettent de modifier la forme et la position des pièces, et d'ajouter, supprimer et modifier les portes.

Aucun opérateur n'a été évoqué pour gérer les fenêtres. La raison en est que la position exacte des fenêtres n'a aucun impact sur la manière d'évaluer un aménagement. Dans notre évaluation, nous avons besoin seulement de savoir si un mur peut supporter une fenêtre pour être sûr que l'architecte puisse la positionner de manière exacte ensuite. Ce qui est vérifiable en consultant uniquement la structure de données.

Pour les besoins de cette section, nous rappelons qu'un sommet est un point d'intersection entre deux ou plusieurs murs (segments), il peut donc appartenir à plusieurs pièces (polygones). Un mur d'un point de vue architectural peut être représenté par plusieurs murs alignés dans notre structure de données puisqu'il est subdivisé sur ses intersections avec les autres murs.

4.1.1. Les opérateurs sur les pièces

Les opérateurs sur les pièces sont des traitements qui impactent l'emplacement et la forme des pièces. Les opérateurs en question sont :

- Créer un sommet ;
- Fusionner deux sommets ;
- Déplacer un sommet ;
- Supprimer un sommet ;

- Permuter deux pièces.

Comme on peut le constater cette liste ne contient aucun opérateur sur les murs. La raison en est que toute opération sur un mur est une combinaison d'opérateurs de cette liste. Par exemple, translater un mur équivaut à déplacer ses deux extrémités.

De plus, les opérateurs sont appliqués à la structure filaire du logement. Par conséquent, à chaque fois qu'un de ces opérateurs modifie la forme d'une pièce (c'est le cas des quatre derniers), un traitement est lancé pour recalculer l'ensemble des pourtours des pièces modifiées, ce qui vaut aussi pour les couloirs (voir le Chapitre II.6).

4.1.1.1. Créer un sommet

La création d'un sommet est l'insertion d'un nouveau sommet dans un segment à une proportion donnée de la longueur du segment (Figure 109). Si le segment à modifier est partagé par deux polygones, le nouveau sommet appartient aux deux.

Notons que cet opérateur ne modifie pas la forme du polygone, il sert uniquement de base pour les autres opérateurs.

L'opération échoue quand le sommet est situé dans l'intervalle réservé à une porte ou est trop proche d'une des extrémités.

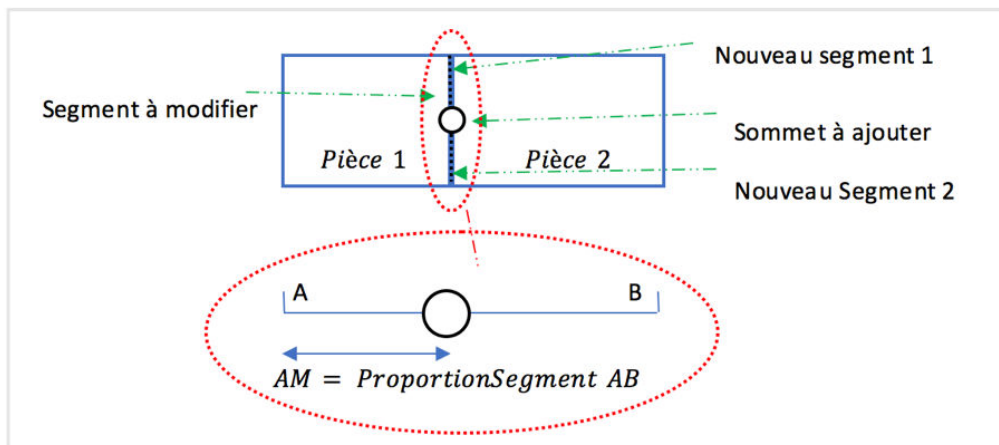


Figure 109. Exemple de l'ajout d'un sommet sur un mur partagé

4.1.1.2. Fusionner deux sommets / Scinder un sommet

La fusion de deux sommets prend deux sommets en paramètres, un sommet à supprimer *SSup* (Figure 110: disque blanc) et un sommet à conserver *SCons* (disque noir). Elle consiste à remplacer *SSup* par *SCons* dans tous les polygones qui utilisent *SSup*. Ces derniers sont donc déformés si *SSup* n'est pas exactement superposé à *SCons*.

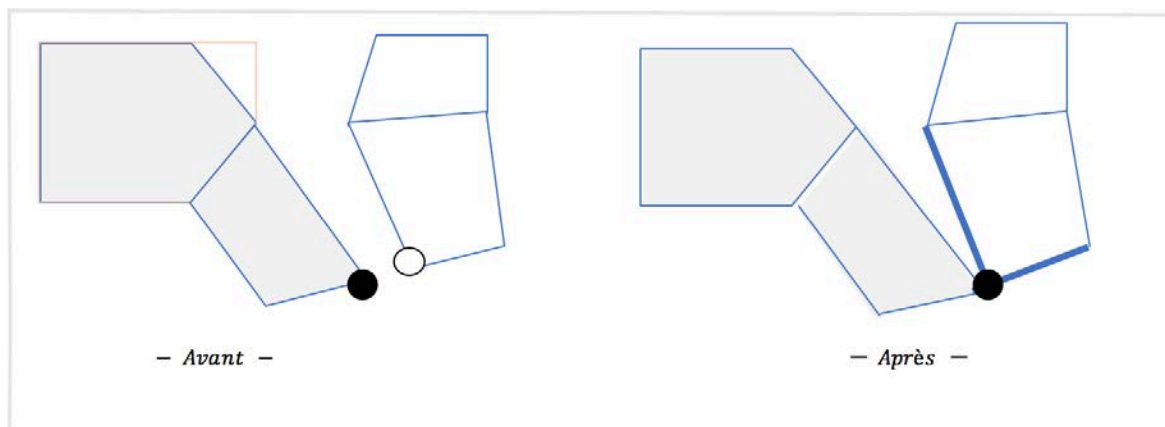


Figure 110. Exemple de la fusion de deux sommets

Ce traitement peut être amené à supprimer un ou plusieurs segments de la structure de données quand la fusion conduit des segments à se superposer. La Figure 111 illustre cette configuration : le sommet à supprimer est représenté par un disque blanc, le sommet à garder par un disque noir, le segment partagé est en pointillés.

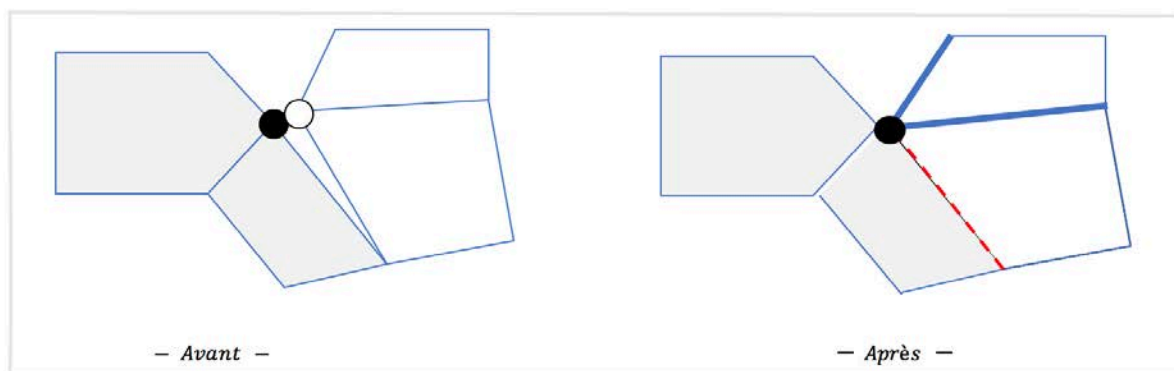


Figure 111. Exemple de la fusion de deux sommets avec un segment en doublon

La raison principale pour laquelle nous avons défini cet opérateur est qu'aucune combinaison d'opérations de bas niveau ne permet la création ou la suppression d'une pièce. À priori, cela pourrait être vu comme un bénéfice, étant donné que l'optimisation d'un aménagement est supposée se faire à nombre constant de pièces. Pourtant, il est nécessaire, pour atteindre une réelle diversité dans les aménagements construits, que des *couloirs* apparaissent ou disparaissent. L'opérateur de fusion est donc nécessaire.

La Figure 112 et la Figure 113 illustrent cette possibilité de créer ou supprimer une pièce avec cet opérateur. Dans la première, on fusionne A et B. Ceci a pour conséquence de faire disparaître la pièce interne en forme de losange.

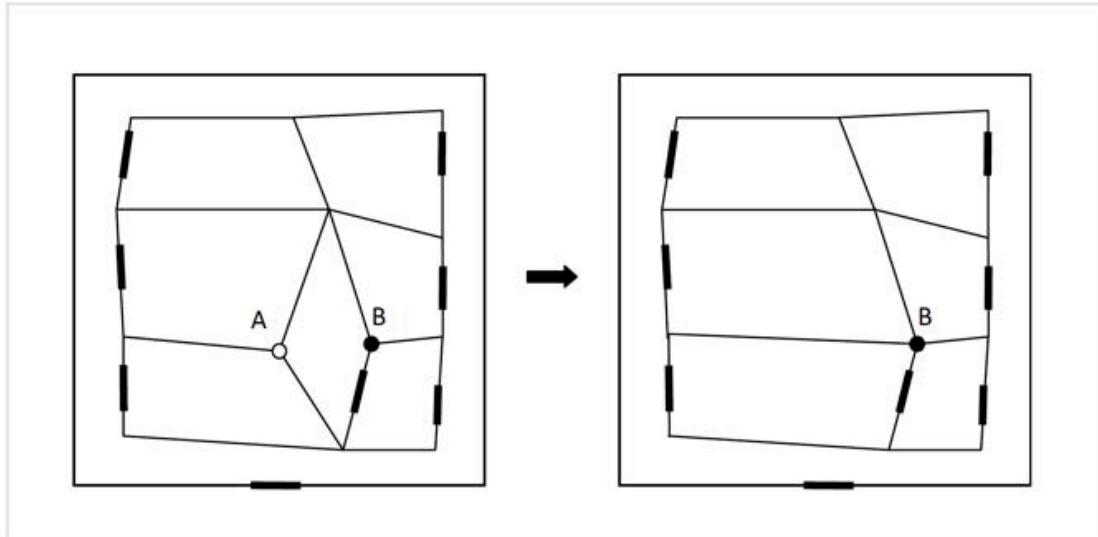


Figure 112. *Suppression d'une pièce avec la fusion de sommets*

Dans le cas de la Figure 112, les deux sommets impliqués sont dans un même polygone et leur fusion a cette fois pour effet de scinder la pièce à laquelle ils appartiennent, ce qui provoque l'apparition d'une pièce supplémentaire.

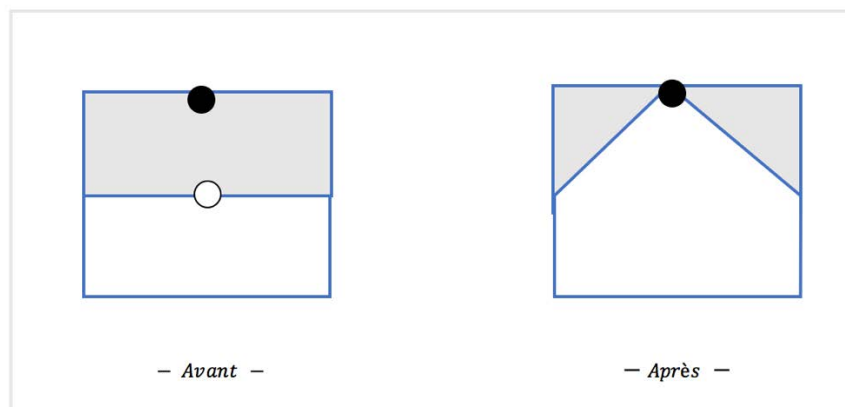


Figure 113. *Exemple de scission d'une pièce*

Pour que la fusion de deux sommets soit tolérée, plusieurs conditions doivent être satisfaites :

- quand les deux sommets sont dans une même pièce (ce qui va créer une pièce supplémentaire), la pièce doit être un couloir. Nous tolérons la création et la suppression de couloirs mais pas de pièces car leur nombre est défini au départ par l'occupant et doit rester constant ;
- le sommet à déplacer ne soit pas être un sommet du contour extérieur, pour éviter de modifier la forme du plateau ;
- La fusion ne doit pas engendrer des situations de recouvrements comme illustré dans la Figure 114.

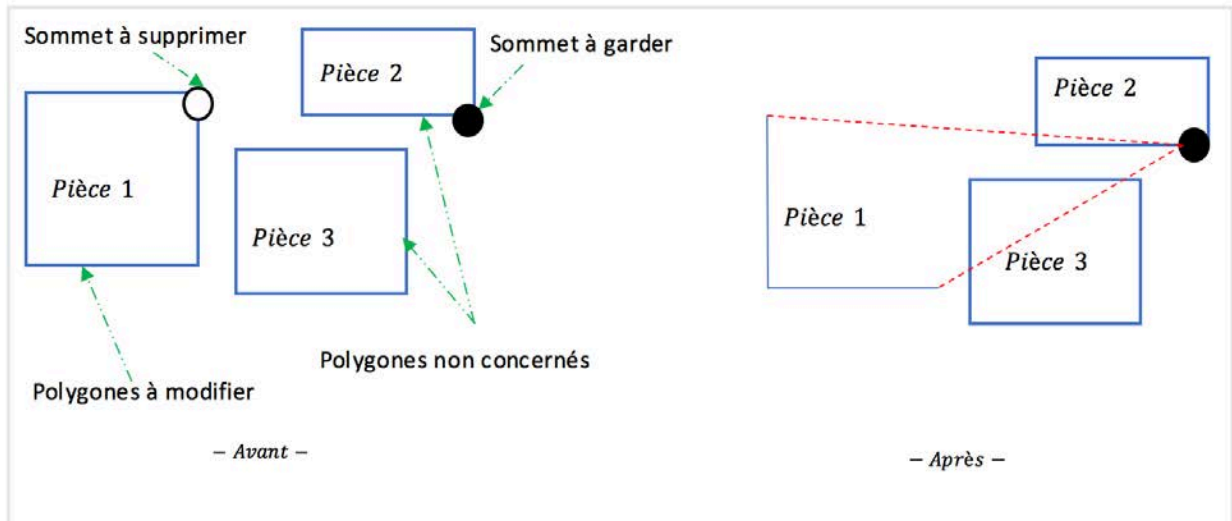


Figure 114. Recouvrement suite à la fusion de deux sommets

L'opérateur de scission d'un sommet S est le symétrique de l'opérateur de fusion : il crée un deuxième sommet S' superposé au premier ; dans la Figure 115 S' est matérialisé par un disque blanc et il est écarté de S pour faciliter la compréhension. En entrée, en plus de S , l'opérateur reçoit la liste des polygones qui conservent S dans leur pourtour (grisés dans la figure). Dans les autres polygones qui utilisaient S , S est remplacé par S' .

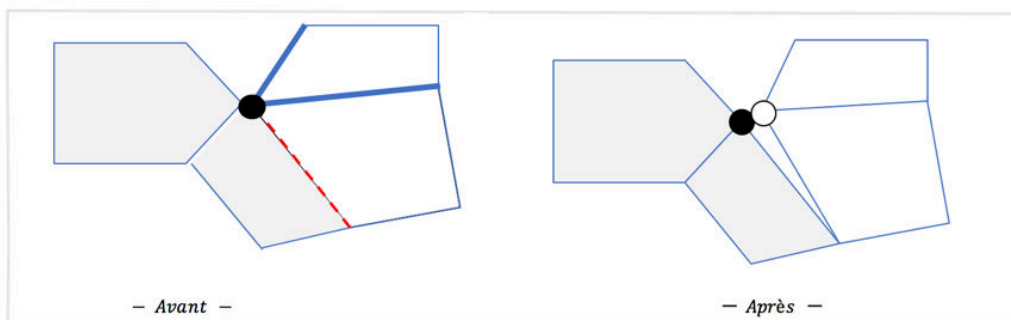


Figure 115. Scission d'un sommet

Comme le montre la Figure 115, cet opérateur peut aussi être amené à dupliquer certains murs : les segments issus de S subissent les opérations suivantes :

- ceux qui sont partagés par un polygone grisé et un polygone non grisé (en pointillés) sont dupliqués ; c'est le cas une fois dans l'exemple, deux fois au maximum ; les segments originaux restent dans les polygones grisés, les autres voient leur origine (S) remplacée par S' et remplacent les segments originaux dans les polygones blancs ;
- ceux qui appartiennent seulement à un polygone blanc (en traits épais) subissent juste un remplacement de (S) par (S') ;
- les autres sont inchangés.

Ce type de traitement ne se conçoit que si un polygone au moins est concerné, et si les polygones d'une catégorie (concernés ou pas) sont adjacents les uns aux autres. Si ces conditions ne sont pas satisfaites, le traitement est annulé.

4.1.1.3. Déplacer un sommet

Cet opérateur permet de changer la position d'un sommet. L'opération a pour conséquence la modification de la longueur et de la position des segments associés au sommet à déplacer (Figure 116).

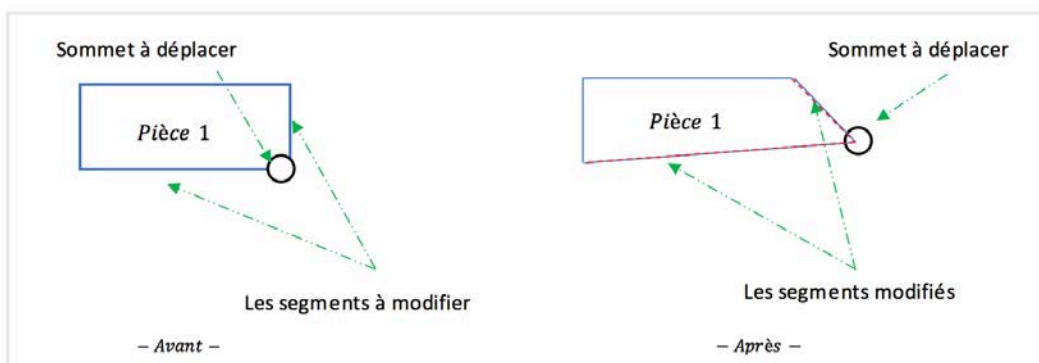


Figure 116. Exemple de déplacement d'un sommet

On refuse le déplacement d'un sommet si :

- c'est un sommet du contour extérieur ;
- sa nouvelle position est celle d'un autre sommet ;
- le déplacement provoque un chevauchement ou un croisement entre des murs.

4.1.1.4. Supprimer un sommet

Supprimer un sommet est une opération qui consiste à retirer le sommet de tous les polygones auxquels il est associé. Cette opération remplace chaque paire de segments associé au sommet supprimé par un seul segment allant de l'origine du segment précédent à l'extrémité du segment suivant (Figure 117).

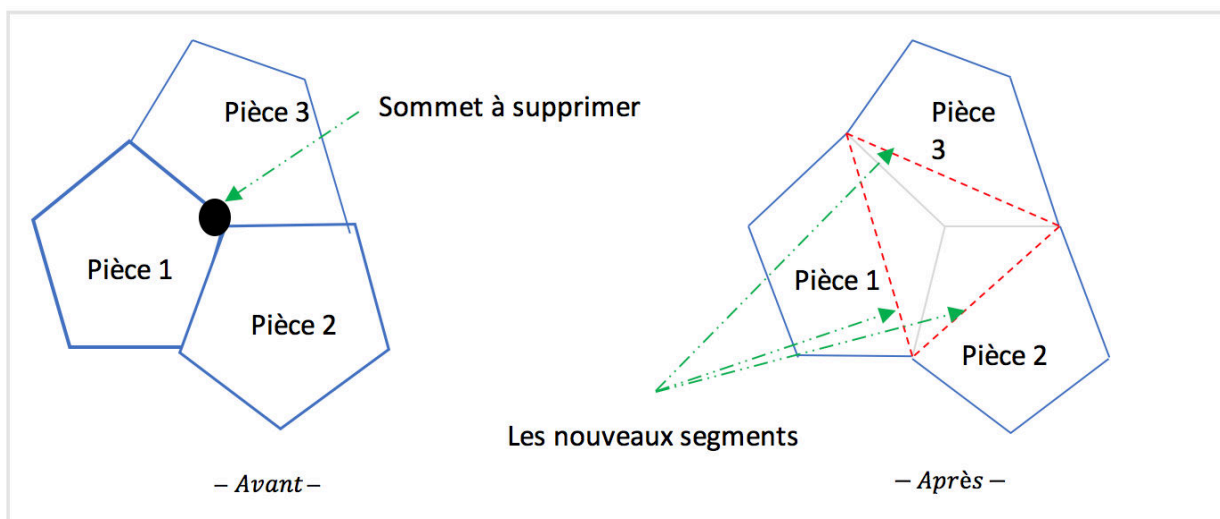


Figure 117. Exemple de suppression d'un sommet

La suppression est interdite si :

- un des polygones du sommet est un polygone avec trois sommets. La suppression d'un sommet invaliderait ce polygone ;
- le sommet à supprimer appartient au contour extérieur ;
- les segments créés coupent d'autres segments.

4.1.1.5. *Permuter deux pièces*

La permutation de deux pièces consiste à inverser le rôle des deux pièces données en paramètres. On ne modifie pas la forme des pièces mais uniquement leurs noms (Figure 118).

Cet opérateur peut toujours être exécuté. Il se peut que, dans l'évaluation qui suivra la modification, on constate que l'une des pièces échangées ne satisfasse pas certains critères et que la note du logement soit globalement moins bonne mais ce n'est pas une raison pour interdire l'échange.

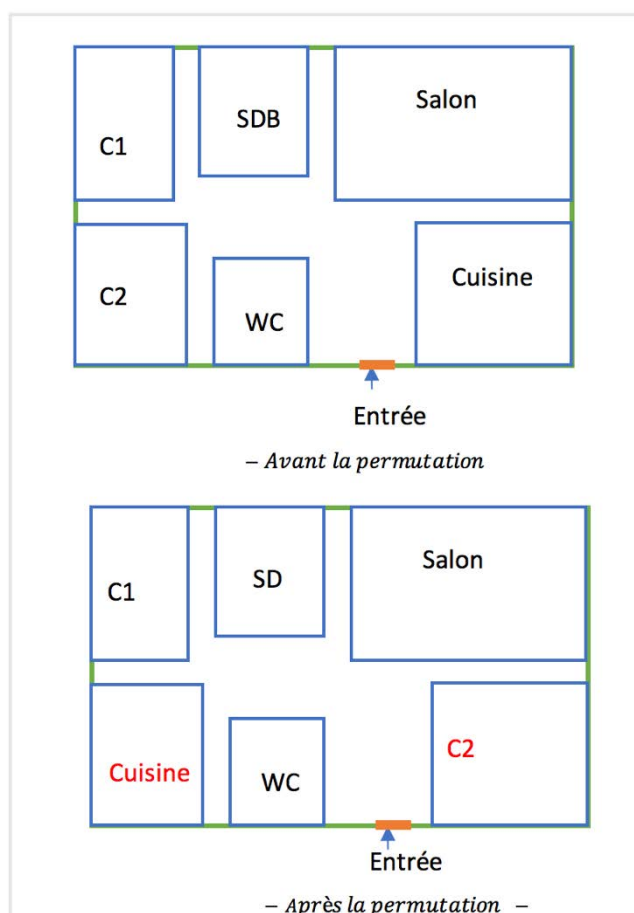


Figure 118. *Exemple de permutation de deux pièces*

Notons que dans le cas où deux couloirs sont adjacents suite à une permutation de deux pièces (une pièce quelconque et un couloir) ces deux couloirs sont fusionnés en un seul et les murs communs sont supprimés (Figure 119).

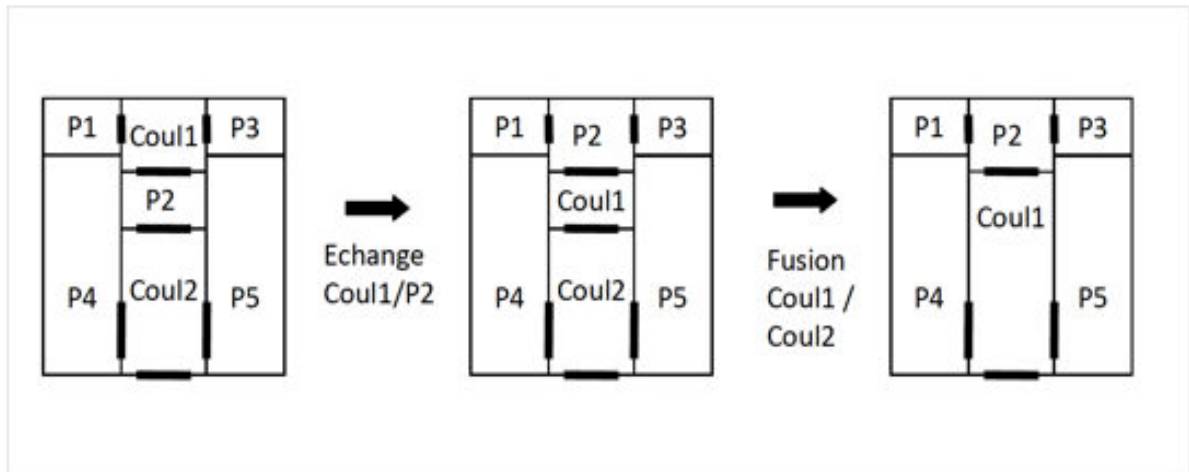


Figure 119. Exemple de la fusion de deux couloirs

Nous estimons qu'un logement ne doit pas avoir deux couloirs adjacents car c'est une perte d'espace pour aucune valeur ajoutée. Cette fusion est effectuée lors de la détection automatique des couloirs qui est déclenchée au début de l'évaluation du logement, aussitôt après la fin de l'exécution de l'opérateur. Autrement dit, il est exact que les formes des pièces ne sont pas altérées *pendant* l'échange mais, si un couloir est impliqué, ce dernier peut voir sa forme changer juste après l'échange.

4.1.2. Les opérateurs sur les portes

Nous avons défini quatre opérateurs pour agir sur les portes. Leur implémentation est simple. Il s'agit des opérateurs suivants :

- ajouter une porte ;
- supprimer une porte ;
- déplacer une porte sur un mur ;
- modifier la largeur d'une porte.

Ces traitements sont rejetés quand :

- l'espace occupé par la porte est déjà réservé pour une autre ouverture (porte ou fenêtre) ou quand l'espace nécessaire sur le mur n'est pas suffisant. Ces deux conditions sont à vérifier quand les opérateurs d'ajout et de modification de la largeur d'une porte sont appelés ;
- la porte déborde sur un côté. Cette condition est à vérifier si on est amené à ajouter, déplacer ou modifier la largeur d'une porte ;
- on supprime l'unique porte d'une pièce.

4.2. Cheminement dans l'espace des solutions

Nous avons vu en introduction que le mécanisme général d'exploration de l'espace des solutions est une recherche locale : la solution courante engendre une nouvelle solution et ceci se poursuit sous le contrôle d'heuristiques qui seront présentées dans la section suivante. Maintenant que les opérateurs de modification sont connus, on peut décrire plus précisément la manière dont la prochaine solution courante est obtenue et la définition de la notion de solution voisine :

- on tente d'appliquer un modificateur à la solution courante S ;

- si l'opération échoue, par exemple parce qu'elle provoquerait une superposition de deux pièces, il n'y a pas eu de progression dans l'espace des solutions donc on recommence le processus de construction de la prochaine solution à partir de S ;
- sinon la validité réglementaire de la nouvelle solution est vérifiée, par évaluation des critères associés. Il suffit qu'un des critères ne soit pas satisfait pour rejeter la solution et donc, à nouveau, on relance le processus à partir de S ;
- à ce stade, la construction a réussi et on a une nouvelle solution courante. Cette fois, il y a donc eu un pas dans l'espace des solutions. On évalue alors les critères qui traduisent les besoins du futur occupant. Si un besoin n'est pas satisfait, la note du logement vaut 0 et l'évaluation peut être interrompue ;
- sinon, on procède à une évaluation des critères (binaires ou gradués) qui correspondent aux souhaits et on agrège les notes partielles obtenues en une note globale.

Nous pouvons à présent définir de manière plus formelle la notion d'espace de solutions. Toute instanciation de la structure de données (même avec des valeurs farfelues) est une solution. Parmi les solutions, celles dont 1) les pièces ne se chevauchent pas et se trouvent à l'intérieur du contour extérieur et dont 2) le contour extérieur n'a pas été déformé forment l'espace des solutions *géométriquement valides*. A l'intérieur de ce domaine, il y a un sous-ensemble qui répond au deuxième niveau d'exigence. Il s'agit du domaine des aménagements qui respectent les normes imposées par le législateur. Nous l'appelons espace des *solutions réglementaires*. À son tour, ce domaine comporte un sous-ensemble qui est celui des aménagements qui obtiennent une note non nulle à l'évaluation. On parle ici de l'espace des solutions partiellement (note dans $]0, 1[$) ou parfaitement (note égale à 1) adaptées à l'occupant et l'objectif est de trouver un optimum de ce dernier ensemble. Dans ce qui suit, nous nous contenterons de dire « solutions adaptées » plutôt que « solutions partiellement ou parfaitement adaptées à l'occupant ».

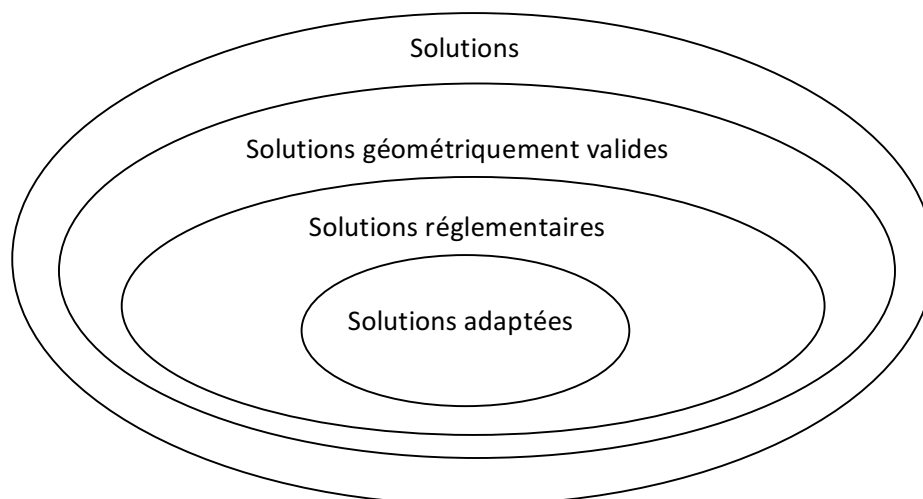


Figure 120. *Vision conceptuelle de l'espace des solutions*

Autrement dit, si on rapproche le processus de construction de la prochaine solution courante rappelé plus haut et ces définitions, on constate que l'on se déplace dans l'espace des solutions adaptées.

On aurait pu choisir de tolérer des incursions dans l'espace des solutions non réglementaires, voire dans les solutions invalides d'un point de vue géométrique. Nous n'avons pas fait ce choix de crainte que les réparations de la solution courante ne prennent trop de temps (l'unité de temps, dans ce cas, étant l'application d'un modificateur). En effet, nous verrons dans la présentation détaillée des heuristiques que le choix aléatoire des modificateurs, rend un retour dans l'espace adapté très ardu.

La section « discussion » donne quelques éléments sur d'autres stratégies que le hasard pour le choix du modificateur à appliquer.

Ouvrons une parenthèse. D'un point de vue formel, et bien que cela n'ait pas d'application immédiate dans ce qui suit, le processus complet d'obtention de l'aménagement optimal peut être décrit en programmation mathématique. Un premier modèle consiste à considérer les aspects géométriques, réglementaires, les besoins et les souhaits comme des contraintes, l'objectif étant de maximiser la fonction $f = \sum_{i=1}^n N_i \cdot x_i$, avec N_i : la note suite à l'évaluation du logement i et x_i : la variable de décision qui prends 1 quand le logement est choisi et 0 sinon. Le programme mathématique s'écrirait comme suit :

$$\mathcal{P} = \begin{cases} \text{Max} & f = \sum_{i=1}^n N_i \cdot x_i \\ \text{s. c} & \\ & \sum_{i=1}^n x_i = 1 \\ & A \cdot x_i = 1 \text{ (} m \text{ contraintes de type aspects géométriques et réglementaires)} \\ & B \cdot x_i = 1 \text{ (} p \text{ contraintes de type besoins)} \\ & C \cdot x_i \leq c \text{ (} t \text{ contraintes de type souhaits)} \\ & x_i \in \{0,1\} \end{cases}$$

Où $A \in \mathbb{R}^{n \times m}$, $B \in \mathbb{R}^{p \times m}$, $C \in \mathbb{R}^{t \times m}$ avec n est le nombre de logement à évaluer.

Ce n'est pourtant pas une vision satisfaisante car les souhaits ne doivent pas être vus comme des contraintes discriminantes au sens de la programmation mathématique, c'est-à-dire des contraintes qui rendent la solution invalide si elles sont pas respectées. Au contraire, les souhaits doivent permettre de comparer des solutions ; ils doivent donc être intégrés à la fonction objectif, ce qui mène au modèle suivant :

$$\mathcal{P}' = \begin{cases} \text{Max} & f = \sum_{i=1}^n N_i \cdot x_i + \alpha_i (c - C \cdot x_i) \\ \text{s. c} & \\ & \sum_{i=1}^n x_i = 1 \\ & A \cdot x_i = 1 \text{ (aspects géométriques et réglementaires)} \\ & B \cdot x_i = 1 \text{ (besoins)} \\ & x_i \in \{0,1\} \end{cases}$$

où $\alpha_i \in [0,1]$ est une pénalité positive, on est pénalisé si la contrainte est non respectée.

Nous refermons à présent cette parenthèse sur la formalisation mathématique et revenons au parcours de l'espace des solutions. Nous avons dit que les solutions rencontrées sont toujours dans l'espace des solutions adaptées ; dans ce qui suit, nous ne parlons donc plus de solutions adaptées mais simplement de solutions. Un des problèmes évoqués en introduction de cette partie sur la modification du logement est de garantir que, si $S1$ et $S2$ sont deux solutions quelconques, le jeu d'opérateurs de modification permet, au moins théoriquement, de transformer progressivement $S1$ en $S2$; la satisfaction de cette propriété, que nous appelons la propriété de *puissance* de l'ensemble des opérateurs, serait la garantie que, quelle que soit la solution de départ, il serait *possible* d'atteindre une solution optimale. Nous n'avons pas de preuve formelle de cette propriété mais nous avons des éléments qui vont dans ce sens : pour certains aménagements de départ et d'arrivée très représentatifs, on peut démontrer la possibilité de passer de l'un à l'autre avec notre jeu d'opérateurs. Ces cas représentatifs sont numérotés et énumérés ci-dessous. Nous ne rappellerons pas, dans les preuves, que les seules modifications permises sont celles qu'offrent les opérateurs de modification.

1. La forme d'une pièce peut-elle changer ? Peut-on, par exemple, passer d'une forme rectangulaire à une forme en L ? La réponse est « oui ».

On passe d'une forme rectangulaire à une forme en L au prix de deux insertions consécutives de sommets et de deux déplacements de sommets (Figure 121).

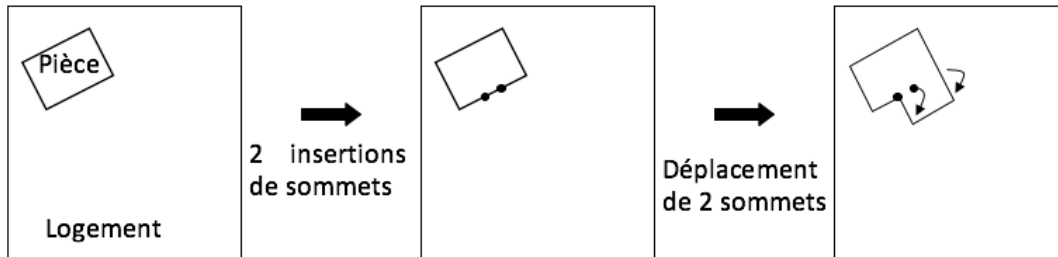


Figure 121. Transformation d'une pièce rectangulaire en une pièce en L

Plus généralement, on passe de n'importe quelle forme à n'importe quelle autre forme en combinant des insertions ou des suppressions de sommets et des déplacements de sommets. Evidemment, les déplacements peuvent être contrariés si les sommets sont partagés avec d'autres pièces ou si des pièces du voisinage interfèrent. Ceci sera traité au point 4. A ce stade, on peut simplement garder à l'esprit qu'une pièce peut être déformée de n'importe quelle manière si elle n'est pas liée, par des partages de sommets ou de murs, avec d'autres pièces.

2. Peut-on faire coulisser une pièce le long d'un mur ? La réponse est « oui », s'il n'y a aucun obstacle.

La Figure 122 montre par exemple dans quel ordre appliquer quatre déplacements de sommets à une pièce rectangulaire (*P1*) pour la faire coulisser jusqu'à son nouvel emplacement, en pointillés. L'ordre importe pour deux raisons. La première est qu'il faut éviter les croisements de murs pendant l'opération. La deuxième est que deux des sommets de *P1* sont partagés avec un mur extérieur ; en réalité, ce « mur » extérieur (le mur droit sur la figure) est subdivisé en trois murs, dans la structure de données, et un ordre maladroit dans les déplacements des quatre sommets de *P1* pourrait amener ces trois murs à se chevaucher temporairement.

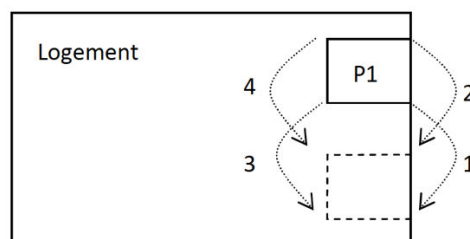


Figure 122. Exemple d'une série de déplacements de sommets pour faire coulisser une pièce

S'il y a des obstacles, c'est-à-dire si, après l'un des déplacements de sommets, la pièce en mouvement a une intersection d'aire non nulle avec une autre pièce, les choses deviennent plus compliquées mais restent possibles. Voir le point 4).

3. Peut-on décoller deux pièces et, ainsi, insérer un couloir entre les deux pièces ?

Ce cas en recouvre plusieurs autres, selon la manière dont les pièces sont accolées : par un ou plusieurs sommets ou par un ou plusieurs murs. Dans la Figure 123, la pièce P1 partage par exemple deux murs et un sommet avec le contour extérieur du logement et elle partage deux murs et deux sommets avec la pièce P2.

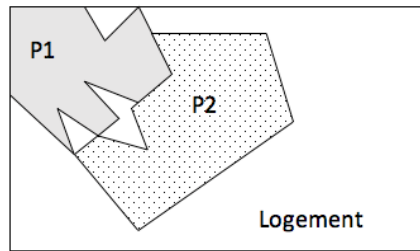


Figure 123. Partages de sommets et de murs

Dans tous les cas, la réponse reste « oui ».

En cas de partage d'un sommet S (Figure 124-gauche), on applique successivement une scission de sommet et un déplacement. Appelons $P1$ la pièce à isoler, $P2...Pn$ les pièces qui partagent S avec $P1$ et S' le sommet créé par la scission de S . La scission est faite de sorte que $P2, P3...Pn$ utilisent S mais aussi de sorte que, dans $P1, S$ soit remplacé par S' (Figure 124-milieu).

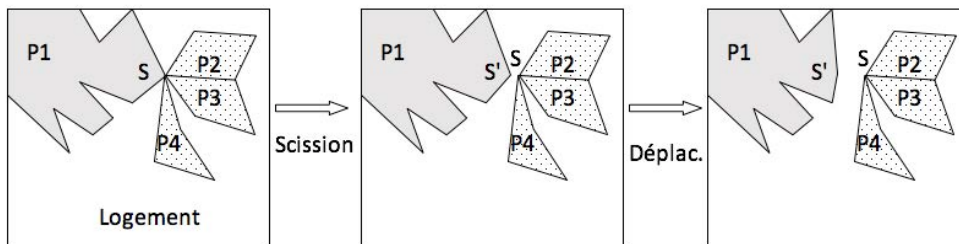


Figure 124. Annulation d'un collage par un sommet

On déplace ensuite S' vers l'intérieur de $P1$ (Figure 124-droite). Ce déplacement est forcément possible car il n'engendre aucune intersection, dès l'instant que l'amplitude du déplacement est suffisamment faible. Les partages de sommets peuvent donc être supprimés.

En cas de partage d'un mur M , la stratégie est semblable (Figure 125- gauche). Notons que le mur n'est partagé que par deux pièces ($P1$ et $P2$ sur la figure) mais que des pièces proches peuvent aussi être concernées ($P3$ et $P4$). La séparation est réalisée en appliquant deux fois la procédure d'annulation du partage d'un sommet présentée ci-dessus, une fois pour chaque extrémité du mur partagé (Figure 125- milieu et droite -).

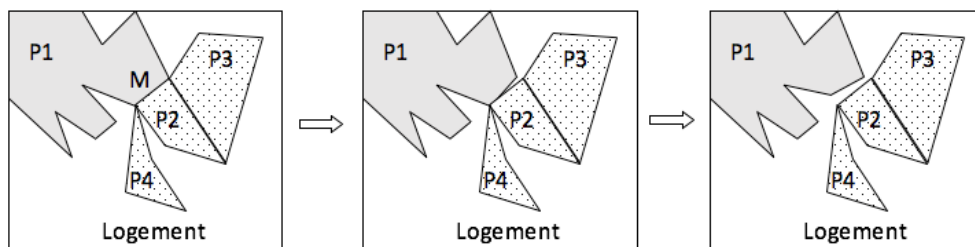


Figure 125. Annulation du partage d'un mur

Ceci prouve que la réponse à la question 3) est « oui » ; la manœuvre s'accompagne d'une (légère) déformation de P1 mais cela est sans importance puisque, si l'intention est effectivement d'insérer un couloir, il faut bien gagner de la place quelque part.

4. Peut-on déplacer une pièce de sa position actuelle vers une position valide donnée ? Une position « valide » doit être comprise comme une position « à laquelle la pièce ne chevaucherait aucune pièce existante ». La réponse est « oui ».

Le cheminement est le suivant :

- on désolidarise la pièce des éventuelles pièces adjacentes en appliquant le raisonnement du point 3,
- on réduit considérablement (à une taille proche du point) la taille de la pièce en appliquant le point 1),
- on détermine un itinéraire allant de la position de départ à la position d'arrivée, ce qui peut impliquer de créer des couloirs entre des pièces (point 3) pour libérer un passage,
- on fait suivre cet itinéraire à la pièce en multipliant les déplacements de sommets,
- arrivé à destination, on fait grandir la pièce à ses dimensions initiales (voir le point 1) ; si la pièce doit partager des sommets ou des murs avec d'autres pièces, on utilise des insertions de sommets dans les pièces auxquelles il faut s'accoler, là où se trouvent déjà les sommets à partager, et on fusionne les paires de sommets superposés.

On conclut donc qu'en utilisant uniquement les opérateurs de modification, on peut amener une pièce à n'importe quel endroit du logement.

5. La topologie du logement peut-elle changer ? Par exemple, peut-on passer de la disposition de la Figure 126-gauche à celle de droite ?

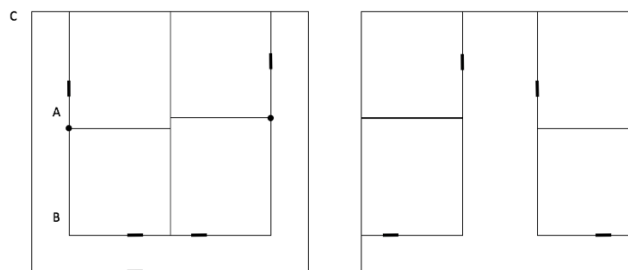


Figure 126. *Changement profond de topologie*

À nouveau, la réponse est « oui » et on le prouve en appliquant des raisonnements semblables à ceux qui ont été utilisés dans les points précédents.

Pour supprimer la branche verticale de gauche du couloir, on peut par exemple insérer des sommets sur le bord gauche du logement, en vis à vis de *A* et *B*. On fusionne ensuite ces deux sommets ainsi que *C* avec les sommets en vis à vis dans les deux pièces de gauche. La branche de droite du couloir est traitée de la même manière.

Pour faire apparaître la branche médiane du couloir, on utilise le point 3).

Cette énumération de cas n'a pas valeur de preuve. Pour qu'elle en devienne une, il faudrait prouver que toutes ces « macro-modifications » suffisent à transformer n'importe quel agencement en n'importe quel autre. Nous considérons cependant que les chances sont élevées que la propriété de puissance de notre ensemble de modificateurs soit avérée.

5. Méthodes de résolution

Le problème de l'amélioration d'un aménagement est un problème NP-difficile [76]. C'est un problème d'agencement, qui est une variante du problème de bin packing, avec comme particularité la taille des objets qui est variable. La résolution de ce problème est possible : 1) soit avec une méthode exacte pour chercher l'optimum global, soit avec une méthode approchée pour chercher une solution de bonne qualité dans un délai raisonnable. Une résolution avec une méthode exacte pour un problème NP-difficile est souvent coûteuse car le temps d'exécution est non polynomial en la taille des instances à traiter. Si les instances du problème sont de petites tailles, la résolution avec une méthode exacte reste envisageable. Par contre, si les instances sont de grandes tailles, une méthode de résolution approchée reste l'unique façon pratique de résoudre le problème. Le principe d'une telle approche est une exploration de l'espace des solutions à la recherche d'une bonne solution (pas forcément l'optimale) en un temps d'exécution raisonnable.

Nous présentons dans cette section les quatre méthodes approchées que nous avons utilisées. Toutes sont basées sur une recherche locale mais elles diffèrent dans la manière avec laquelle la prochaine solution étudiée est choisie.

Dans ce qui va suivre, nous allons commencer par rappeler le principe d'un algorithme de recherche locale, nous parlerons ensuite l'ensemble de ses paramètres et à la fin, nous détaillerons nos quatre méthodes. Nous rappelons juste que le terme « solution » est pour nous synonyme d'un aménagement adapté.

5.1. Définitions

Une recherche locale appelée aussi procédure d'amélioration est un algorithme itératif qui part d'une solution initiale S et considère un sous ensemble $N(S)$ de solutions proches en termes de structure, appelé *voisinage* de S . Ce voisinage est exploré pour trouver une solution S' qui devient à son tour la solution courante pour l'itération suivante. En d'autres termes, la procédure consiste à parcourir l'espace des solutions de proche en proche à la recherche d'une bonne solution (Figure 127) [78].

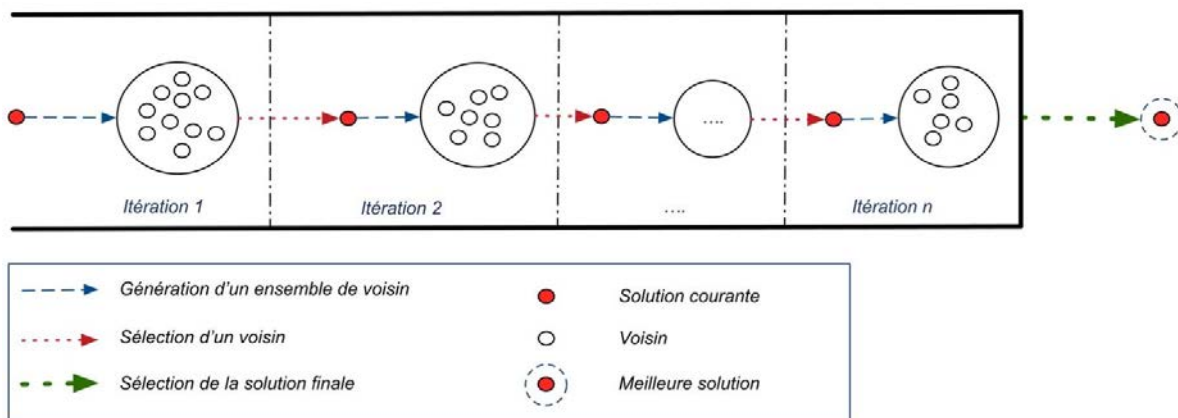


Figure 127. Procédure de recherche locale

Techniquement la recherche locale consiste en une succession de transformations de la solution afin de l'améliorer à chaque fois. La solution initiale est progressivement convertie en un optimum local. Le processus est interrompu quand les conditions d'arrêt sont atteintes (plus d'amélioration constatée, la durée impartie au traitement est écoulee...). L'algorithme suivant décrit le processus de recherche locale :

```

Fonction RechercheLocale ( $S_{initiale}, N, f$ ) : Solution  $S_{finale}$ 

// N : une fonction pour générer un ensemble de voisins
// f : une fonction d'évaluation d'une solution
//  $S_{initiale}$ : solution initiale
// S : solution courante
// SVs : ensemble des solutions voisines de S

Début

    S  $\leftarrow$   $S_{initiale}$ 

     $S_{finale}$   $\leftarrow$   $S_{initiale}$ 

Répéter

     $SV_s \leftarrow N(S)$  // générer les voisins de S

    S'  $\leftarrow$  sélectionner une solution dans  $SV_s$ 

    S  $\leftarrow$  S' // remplacer S

    Si  $f(S) > f(S_{finale})$  alors

         $S_{finale} \leftarrow S$  // garder la meilleure solution rencontrée

    FSI

Jusqu'à Condition d'arrêt

Fin

```

Algorithme 5 Algorithme de recherche locale

Si le voisinage $N(S)$ est *complètement* exploré, la recherche locale garantit l'obtention de la meilleure solution parmi toutes les solutions voisines. Toutefois, cet optimum local reste relatif à la transformation déployée. Il est bien sûr possible qu'un optimum local pour une transformation ne soit pas le même optimum local pour une autre transformation. Le choix des transformations et de leur mise en œuvre est donc une partie importante d'une recherche locale.

La performance de la recherche locale, que ce soit par la force de calculs qu'elle requiert ou la qualité de la solution qu'elle permet d'obtenir, dépend du pas utilisé et de la manière dont on choisit d'accepter une solution voisine comme solution courante.

5.2. Voisinage

En pratique, $N(s)$ est défini implicitement par un type de transformation $S \rightarrow S'$ appelé *mouvement*. Un mouvement est alors un pas dans l'espace des solutions. Il peut être un grand ou un petit pas selon qu'il est synonyme d'une transformation élémentaire ou d'une combinaison de plusieurs transformations élémentaires. Pour qu'il soit à la fois assez grand pour ne pas ralentir le processus et assez petit pour ne pas passer à côté de certaines solutions voisines qui peuvent améliorer la qualité de la solution finale.

Un « bon » voisinage est un voisinage riche de solutions de différentes. Les solutions générées ne doivent pas être très similaires au point d'avoir en réalité un voisinage composé d'une même solution. Plus le voisinage est riche de dispositions différentes, plus les chances de tomber sur des solutions intéressantes sont grandes. Cela est similaire à la démarche de l'architecte qui commence par imaginer plusieurs esquisses assez différentes et qui va ensuite les affiner pour, finalement, n'en garder que quelques-unes qui seront proposées au client.

Une solution voisine pour nous est une solution générée en appliquant un opérateur de bas niveau tiré et paramétré de manière aléatoire. Nous aurions pu bien sûr définir un mouvement autrement, comme étant un ensemble d'opérateurs par exemple, mais cela semblait inutile d'après les premiers tests effectués. Le mouvement a donc été défini de manière empirique mais reste facilement modifiable pour agrandir le pas si le besoin se manifeste.

De plus, le choix d'un opérateur tiré et paramétré aléatoirement permet de générer des solutions voisines réellement différentes. Si, par exemple, le seul mouvement possible était un déplacement de sommet, on verrait des voisins très semblables les uns aux autres. La possibilité de fusionner deux sommets, de supprimer un sommet... permet de créer ou supprimer des couloirs, permuter des pièces... et donc d'engendrer des dispositions différentes les unes des autres dans un même ensemble de voisins.

5.3. Pondération des opérateurs de modification

Il existe une corrélation entre les opérateurs les plus contraints (ceux dont les conditions d'application ont le moins de chance d'être réunies) et les modifications apportées à une solution : plus un opérateur modifie en profondeur un aménagement, plus il est contraint. Par exemple, un des opérateurs les plus contraints est l'opérateur de fusion : si on autorise la fusion de deux sommets d'une même pièce (autre qu'un couloir), on augmente le nombre de pièces, ce qui n'est pas souhaitable car ce nombre est figé à l'avance ; les 2 sommets fusionnés doivent donc appartenir à deux pièces différentes ou à un même couloir ; de plus, les déformations engendrées par la fusion ne doivent provoquer aucun recouvrement. Les données de la fusion (les deux sommets concernés) étant choisies aléatoirement, la probabilité que l'opération soit acceptée est donc faible.

Un opérateur contraint est un opérateur dont les chances d'aboutir sont faibles. Dans l'ordre décroissant des opérateurs contraints, on retrouve la fusion de deux sommets et la suppression d'un sommet (qui n'est possible que si la pièce a au moins quatre sommets). Le déplacement d'un sommet est en troisième position car un déplacement peut provoquer des chevauchements. Les opérateurs restants sont moins contraints, ils sont rarement rejetés.

Donner autant de chances à chaque opérateur provoquera un voisinage peu varié à chaque itération car les opérateurs les plus utilisés seront les opérateurs les moins contraints et donc ceux qui modifient le moins un aménagement. Cela impactera forcément la qualité de la solution finale car elle a de grandes chances d'être une variante très proche de la solution initiale.

Pour remédier à ce problème, nous avons décidé de pondérer les opérateurs. La pondération ici est synonyme d'un nombre de tentatives associées à la réussite d'un opérateur. Par exemple, on suppose qu'on a tiré au hasard l'opérateur de la fusion ainsi que ses paramètres pour générer un voisin. Cet opérateur a toutes les chances de ne pas réussir lors de la première tentative. Dans ce cas, on lui donne une deuxième chance avec un paramétrage différent. Si ça ne marche pas, on lui donne une troisième chance et ainsi de suite jusqu'à atteindre le nombre de tentatives maximal. C'est seulement si aucune de ces tentatives ne réussit que l'on choisit un autre opérateur. Le poids de chaque opérateur est défini comme étant la probabilité qu'il réussisse. Cette valeur est définie de manière empirique par une phase de calibrage.

A noter que les opérateurs les moins contraints, même s'ils ne modifient pas en profondeur un aménagement, restent importants et nécessaires pour générer des solutions voisines différentes. L'insertion d'un sommet, par exemple, permet de créer des points d'articulation qui vont servir de base aux opérateurs comme le déplacement ou la fusion et d'apporter des modifications plus profondes à un aménagement donné.

5.4. Les méthodes implémentées

La résolution de notre problème est faite avec quatre méthodes différentes. La première méthode est une recherche locale aléatoire qui s'appuie sur la seule puissance de calcul des ordinateurs (force brute). Les deux méthodes suivantes sont des méthodes de descente ; sélection d'un voisin pour ces deux méthodes faites de manière qualitative : dans le premier cas, le voisin choisi est la première solution améliorante rencontrée dans le voisinage ; dans le deuxième, le voisin est la meilleure solution améliorante parmi un ensemble de solutions voisines. La quatrième méthode est un recuit simulé pour remédier à certaines limites des méthodes précédentes.

Le choix de ces quatre méthodes est fait pour les raisons suivantes :

- on a privilégié la manipulation d'une solution à la fois car une famille de solutions, comme dans un algorithme génétique, est très gourmande en mémoire;
- l'utilisation d'un algorithme génétique nécessite la définition de fonctions de croisement et de mutation. La mutation pourrait être assimilée à nos opérateurs de modifications par contre, le croisement de deux solutions s'avère être une tâche complexe, dont l'analyse et la mise en œuvre n'était pas compatible avec le temps dont nous disposions ;
- les méthodes qu'on a implémentées sont rapides à mettre en œuvre. Leur rôle était principalement de valider notre approche et de nous permettre de vérifier le bon fonctionnement de toutes les couches sur lesquelles elles s'appuient (structures de données du logement, évaluation, modification). De ce point de vue, elles ont bien joué leur rôle.

5.4.1. Recherche locale aléatoire

Un unique voisin est généré en modifiant aléatoirement un aménagement à l'aide d'un opérateur tiré et paramétré au hasard. Peu importe s'il dégrade ou améliore la solution courante. La solution finale est la meilleure solution rencontrée. L'algorithme s'arrête quand on atteint un nombre de modifications maximal fixé au préalable. L'algorithme est le suivant :

```

Fonction RechercheLocaleAléatoire ( $S_{initiale}, N, f$ ) : Solution  $S_{finale}$ 

// N : une fonction pour générer un seul voisin
// f : une fonction d'évaluation d'une solution
//  $S_{initiale}$ : solution initiale
// S : solution courante

Début

    S ←  $S_{initiale}$ 

     $S_{finale}$  ←  $S_{initiale}$ 

Répéter

    S ← N(S) // remplacer la solution courante par son voisin

    Si  $f(S_{finale}) > f(S)$  alors

         $S_{finale}$  ← S // garder la meilleure solution rencontrée

    FSI

Jusqu'à Condition d'arrêt

Fin

```

Algorithme 6 Algorithme de recherche locale aléatoire

Cet algorithme a l'avantage d'être simple à implémenter. Il repose sur la force des calculs pour générer des solutions. Il est utilisé en général comme une base de comparaison avec d'autres méthodes de résolution plus élaborées [78]. Nous l'utilisons ici pour les mêmes raisons, c'est à dire, une base de comparaison pour les méthodes utilisées ci-dessous.

5.4.2. Les méthodes de descente

Les méthodes de descente (nous conservons le terme traditionnel de « descente », bien que nous cherchions à maximiser la valeur du logement ; « montée » aurait peut-être été préférable) sont des méthodes rapides [63,64]. Les voisins sont générés comme précédemment, c'est à dire, par applications d'opérateurs de modification tirés et paramétrés aléatoirement (mais, comme nous l'avons expliqué plus haut, le tirage n'est pas équiprobable).

Concernant la sélection du voisin qui remplacera la solution courante, Plusieurs procédures existent. Deux procédures en particuliers nous intéressent : First Improvement Hill-Climbing (FIHC) et le best Improvement Hill-Climbing (BIHC).

5.4.2.1. Premier voisin améliorant : First Improvement Hill Climbing (FIHC)

Cette procédure consiste à faire une exploration non exhaustive du voisinage pour choisir comme solution courante à chaque étape, le premier voisin améliorant rencontré. Elle n'est donc pas obligée d'explorer tous les voisins et se contente d'avancer vers une meilleure solution à chaque fois que cela est possible. La procédure s'arrête quand on a atteint un optimum local (qui est caractérisé par le fait qu'on n'arrive plus à améliorer la solution courante) ou quand le nombre de modifications maximal autorisé est atteint. L'algorithme suivant décrit cette procédure :

```
Fonction FIHC ( $S_{initiale}, N, f$ ) : Solution  $S_{finale}$ 

// N : une fonction pour générer un seul voisin
// f : une fonction d'évaluation d'une solution
//  $S_{initiale}$ : solution initiale
// S : solution courante
// S' : Solution voisine générée

Début

    S ←  $S_{initiale}$ 

     $S_{finale}$  ←  $S_{initiale}$ 

Répéter

    S' ← N(S) // Solution voisine de S générée

    Si  $f(S') > f(S)$  alors

        S ← S' // Remplacer la solution courante

    FSI

Jusqu'à Condition d'arrêt

 $S_{finale} = S$  //La solution finale est la dernière solution courante

Fin
```

Algorithme 7 Algorithme FIHC

En général cette procédure converge rapidement vers des solutions meilleures que la solution initiale. Néanmoins, le fait de ne pas explorer tout le voisinage à chaque étape peut s'avérer pénalisant car souvent la première solution améliorante rencontrée n'est pas la meilleure de tout le voisinage. Ce n'est pourtant pas systématiquement une cause de moins bon fonctionnement, nous revenons sur ce point après la présentation de la méthode suivante.

5.4.2.2. Meilleur voisin améliorant : Best Improvement Hill Climbing (BIHC)

Cette procédure se repose sur les mêmes principes que la procédure First Improvement, à un détail près : on ne s'arrête pas au premier voisin améliorant, on explore tout le voisinage pour sélectionner le meilleur améliorant possible.

Ces deux dernières méthodes, toutes les deux strictement améliorantes, mènent à la même solution si la fonction à maximiser a un seul maximum dans le voisinage du point de départ. En effet, dans ce cas, peu importe que l'on suive la ligne de plus forte pente (trait continu issu de A dans la Figure 128) ou un itinéraire améliorant (trait en pointillés), on aboutit au même point (trait plein issu de A). Quant au temps d'exécution, il est difficile de tirer la moindre conclusion puisque la trajectoire est plus courte dans le cas de BIHC mais l'exploration du voisinage, comme expliqué dans le paragraphe précédent, est plus longue.

Quand il n'y a pas qu'un seul maximum dans le voisinage du point de départ (pour dire les choses de manière plus précise, on devrait dire : quand toutes les trajectoires strictement ascendantes ne mènent pas au même maximum local), il est tout aussi compliqué d'établir un comparatif du comportement des deux méthodes. Par exemple, dans le cas du point B, BIHC termine sur un moins bon maximum (trait plein) que FIHC.

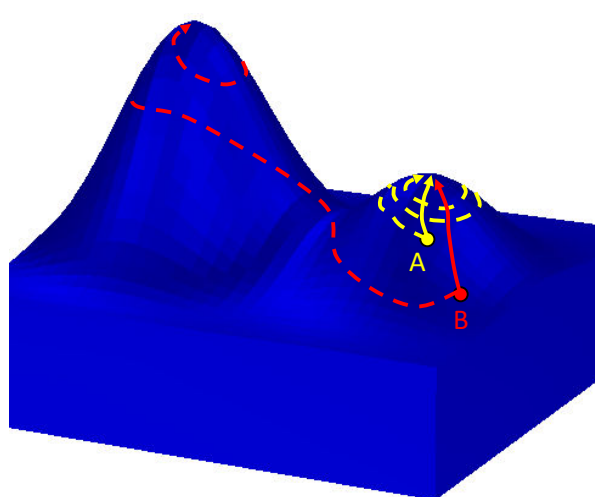


Figure 128. Comparaisons de FIHC et BIHC

5.4.3. Le recuit simulé

5.4.3.1. Origine

La structure complexe de l'espace des configurations d'un problème d'optimisation difficile a inspiré une analogie avec des phénomènes physiques, qui a conduit trois chercheurs de la Société IBM à proposer, en 1982, et à publier, en 1983, une nouvelle méthode itérative : la méthode du recuit simulé [43], qui évite les minimums locaux. Des travaux semblables, développés indépendamment à la même époque par V.Cerny [14] ont été publiés en 1985 [78].

Cette technique est inspirée du recuit en métallurgie, qui consiste à porter le matériau à haute température, puis à abaisser la température par paliers. À titre d'illustration, nous représentons sur la Figure 129 l'effet de la technique du recuit, et celui de la technique opposée de la trempe, sur un système formé d'un ensemble de particules.

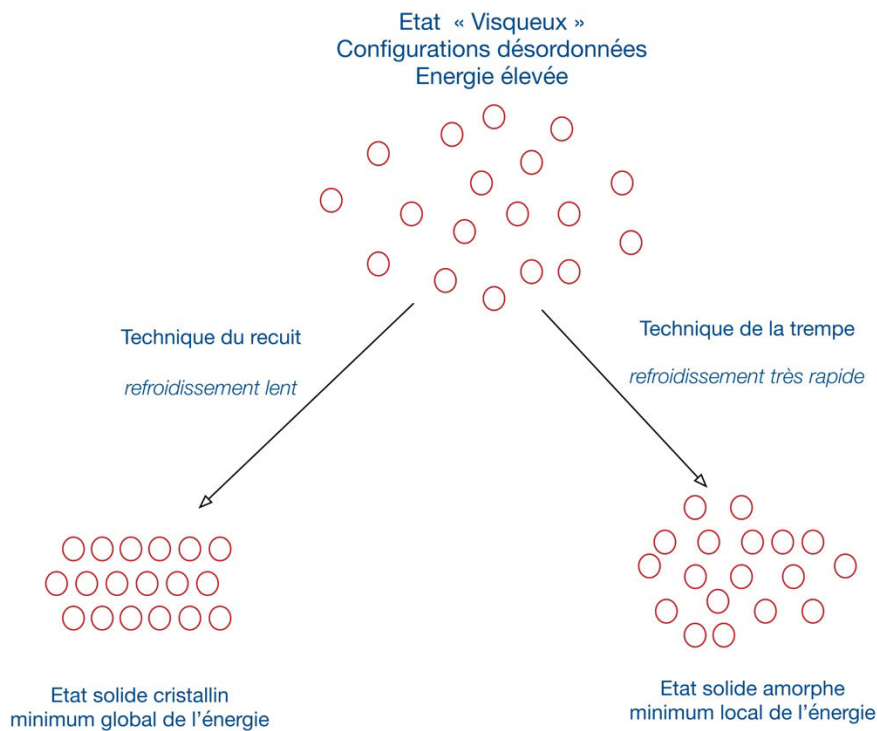


Figure 129. Comparaison des techniques du recuit et de la trempe d'après P.Siarry [78]

Le phénomène qui amène les atomes à prendre une organisation cristalline (à « se ranger bien ») est le suivant. On appelle *entropie* la quantité d'énergie nécessaire au maintien de la cohésion d'un solide. Les lois de la physique font que la matière tend à réduire cette entropie. Quand le solide est chauffé, l'agitation thermique agite les atomes et les amène à se déplacer. Quand la nouvelle disposition est moins bien rangée, l'énergie nécessaire au maintien de la cohésion est plus grande donc la matière tend à refuser cet arrangement mais, au début du processus, l'agitation thermique est telle que la matière en est empêchée. Quand la nouvelle disposition est mieux rangée, elle est acceptée puisque l'entropie est moindre.

Au fur et à mesure de la diminution de la température, l'agitation thermique diminuant, il arrive de moins en moins souvent que les dégradations de l'arrangement soient acceptées. Les dégradations qui ont été acceptées au début ont permis de tenter de nombreuses nouvelles organisations des atomes et, globalement, les améliorations étant systématiquement acceptées, la probabilité est élevée qu'à la fin, les atomes soient bien rangés.

Ce même principe fait que, lorsqu'on prend une poignée d'allumettes et qu'on les pose dans une boîte qu'on agite, ou bien lorsque qu'on verse du café moulu dans un récipient qu'on agite, les allumettes tendent à se placer parallèles les unes aux autres et le niveau du café diminue.

5.4.3.2. Principe du recuit simulé

Dans le recuit *simulé*, c'est la fonction coût (ou fonction objectif) qui tient le rôle de l'entropie et on fait pour l'instant l'hypothèse (bien que ce ne soit pas notre cas) qu'elle doit être minimisée. Pour cela, on exploite l'algorithme de *Metropolis* [58], qui permet de décrire le comportement d'un système en « équilibre thermodynamique » à une certaine température T : partant d'un aménagement donné, on fait subir à l'aménagement une modification élémentaire (par exemple, dans notre cas, on permute

deux pièces) ; si cette transformation a pour effet d'améliorer (de diminuer) la fonction coût, par analogie avec le recuit métallurgique, elle est acceptée ; si elle provoque au contraire une augmentation de la fonction coût C c'est à dire un $\Delta C > 0$, elle peut être acceptée mais avec la probabilité de $e^{-\frac{\Delta C}{T}}$. On constate que, ΔC étant positif, $-\Delta C/T$ est négatif donc son exponentielle est entre 0 et 1, ce qui correspond bien à une probabilité. De plus, plus la température est forte, plus le ratio est proche de 0 donc plus la probabilité est proche de 1 et, donc, plus la chance d'accepter la détérioration est forte : on reproduit le comportement du recuit métallurgique qui, au début du refroidissement, accepte toutes les dégradations. Symétriquement, quand T devient petite, le rapport tend vers $-\infty$ et la probabilité de tolérer une dégradation de la solution devient très faible.

On répète cette opération, en gardant la température constante, jusqu'à ce que l'équilibre thermodynamique soit atteint, concrètement jusqu'à un nombre « suffisant » de modifications. On abaisse alors la température avant d'effectuer une nouvelle série de transformations : la loi de décroissance par paliers de température est empirique tout comme le critère d'arrêt du processus.

5.4.3.3. Paramétrages et algorithmes

Un recuit au sens métallurgique consiste à minimiser l'énergie nécessaire pour atteindre une structure stable du matériau. La fonction objectif d'un recuit simulé est donc une fonction à minimiser. Dans le cas de l'amélioration d'un lieu de vie, l'objectif est de maximiser la note de l'évaluation. La fonction coût, qui est une note de l'aménagement et qui varie entre 0 et 1, est donc à maximiser. Pour adapter le recuit simulé à notre problématique, il suffit de remplacer *Coût* par $1 - \text{Valeur}$. Ainsi, minimiser le *Coût* revient à maximiser la *valeur*.

Les paramètres du recuit simulé que nous avons utilisé sont les suivants : la solution initiale (à optimiser), les températures initiale et finale, le nombre et la durée des paliers de température.

Pour calculer les deux températures, le recuit demande qu'on lui fournisse le pourcentage de dégradations qu'il faut accepter au début de recuit (qui doit logiquement être inférieur à mais proche de 1). Pendant une phase de calibrage (dont la durée est une entrée de l'algorithme), le recuit enchaîne alors des modifications successives de la solution initiale et mémorise la plus grande variation de coût ΔC_{Max} qu'une modification a provoquée. Les températures initiale T_i et finale T_{finale} en sont déduites avec le raisonnement suivant.

Rappelons d'abord que la probabilité d'accepter une dégradation d'amplitude ΔC à une température T est $e^{-\frac{\Delta C}{T}}$.

Initialement, quand la température est T_i , la probabilité d'accepter une dégradation doit être égale au taux fourni en entrée, que nous appelons *TauxDéteriorations acceptées au début*. On en tire :

$$e^{-\frac{\Delta C_{Max}}{T_i}} = \text{Taux}_{\text{Déteriorations acceptées au début}} \Rightarrow T_i = -\frac{\Delta C_{Max}}{\ln(\text{Taux}_{\text{Déteriorations acceptées au début}})}$$

En fin d'algorithme, la probabilité d'accepter une dégradation doit être très faible. Nous l'avons fixée à un pour mille (0.001). Avec le même raisonnement que précédemment, on en tirerait que :

$$e^{-\frac{\Delta C_{Max}}{T_{finale}}} = 0.001 \Rightarrow T_{finale} = -\frac{\Delta C_{Max}}{\ln(0.001)}$$

Ceci soulève un problème qui a été révélé par l'expérimentation : quand la dégradation est très faible, par exemple de l'ordre d'un dixième de ΔC_{Max} , et quand on est proche de la température finale, la probabilité d'accepter la dégradation est :

$$e^{-\frac{\Delta C_{Max}/10}{T_{finale}}} = e^{-\frac{\Delta C_{Max}/10}{\ln(0.001)}} = e^{\frac{\ln(10^{-3})}{10}} = \frac{1}{10^{3/10}} \approx 0,5 !!$$

ce qui constitue une probabilité beaucoup trop élevée en fin d'algorithme. Nous avons donc calculé la température finale en imposant que la probabilité d'accepter une détérioration *1000 fois plus petite* que ΔC_{Max} soit égale à 1 pour mille, ce qui donne :

$$e^{-\frac{\Delta C_{Max}/1000}{T_{finale}}} = 0.001 \Rightarrow T_{finale} = -\frac{\Delta C_{Max}}{1000 \cdot \ln(0.001)}$$

Un raisonnement semblable nous a amené à ne pas accepter, en cours d'algorithme, les très petites dégradations de la solution courante. En effet, quand la détérioration est très faible, $-\frac{\Delta C}{T}$ est proche de 0 quelle que soit la température T donc la probabilité d'accepter la détérioration est proche de 1. La conséquence est la suivante : quand de nombreuses détériorations de petite ampleur se succèdent, elles sont toutes acceptées. De ce fait, même en fin d'algorithme, la solution courante se détériore alors que ce ne devrait plus être le cas. Une autre entrée du recuit est donc un seuil ; quand la variation de coût est positive (dégradation) et quand elle est sous ce seuil, on ne la soumet pas à la règle de *Metropolis* pour lui donner une chance d'être acceptée : elle est immédiatement rejetée.

La durée d'un palier est contrôlée par deux derniers paramètres d'entrée : le nombre maximum de tentatives de modification autorisées durant un palier et le nombre maximum de modifications acceptées durant un palier. Dès que l'un de ces seuils est atteint, on passe au palier suivant.

L'algorithme du recuit simulé peut être formalisé comme suit :

```
Fonction RecuitSimulé (Sinitiale, TauxDétériorations acceptées au début, SeuilDegradation, NombreMaximumTentatives, NombreMaximumModifications) : Sfinale
```

```
// Sinitiale : solution initiale
```

```
// Scourante : solution courante
```

```
// SeuilDegradation : la plus petite détérioration acceptable
```

```
// Sfinale : solution finale
```

```
// Tinitiale : température initiale
```

```
// Tfinale : température finale
```

Début

```
CalculerTempératuresInitialeEtFinale(SolutionInitiale, TauxDétériorations acceptées au début, Tinitiale, Tfinale)
```

```
Température ← Tinitiale
```

```
MeilleureSolution ← Scourante ← Sinitiale
```

```
CoutSolutionCourante = CoutMeilleureSolution = Cout (Sinitiale)
```

```
Stationnaire = faux
```

```
// On réduit la température par paliers
```

```
Tant que (Température > Tfinale) et (Non Stationnaire) faire
```

```
// On fait des modifications pendant toute la durée du palier
```

```
NombreTentatives = NombreModifications = 0
```

```
Tant que (NombreTentatives < NombreMaximumTentatives) et
```

```
(NombreModifications < NombreMaximumModifications)
```

```
faire
```

```
Solution = Modification (Scourante)
```

```
CoutSolution = Cout(Solution)
```

```
Si (CoutSolution < CoutSolutionCourante) alors
```

```
// on accepte la modification car elle améliore
```

```
SolutionAcceptee = vrai
```

```
Sinon
```

```

Si (CoutSolutionCourante - CoutSolution > SeuilTolérance)
alors

    // on n'accepte la dégradation qu'avec
    // une certaine probabilité et que si elle
    // n'est pas trop petite

    Δcoût = CoutSolution - CoutSolutionCourante

    SolutionAcceptee = (aleatoire (0,1) < e-Δcoût/Température)

Fsi

Fsi

Si (SolutionAcceptee) alors

    Scourante = Solution

    CoutSolutionCourante = Cout (Solution)

    // Peut-être améliore-t-on la meilleure solution

    Si (CoutSolutionCourante < CoutMeilleureSolution) alors

        MeilleureSolution = Scourante

        CoutMeilleureSolution = CoutSolutionCourante

    Fsi

Fsi

Fin tant que // Fin de la du palier

    Stationnaire = (NombreModifications = 0)

    Temperature = Evolution (Temperature)

Fin tant que

retourner MeilleureSolution

Fin

```

Algorithme 8 Algorithme de recuit simulé

6. Le logiciel réalisé et les expérimentations

6.1.1. Introduction

Une fois définie la méthodologie qui devait permettre l'amélioration d'un aménagement, il nous est vite apparu que la principale difficulté à laquelle nous allions faire face serait la mise au point des opérateurs de modification, de leur conception détaillée à leur stabilisation. Précisons simplement que, pour les tester de manière exigeante, il nous fallait un contexte réaliste : le principal rôle des quatre méthodes d'amélioration a été de fournir ce contexte, puisqu'elles utilisent intensivement tous les modificateurs. La phase post-doctorat de ce travail consistera à définir des heuristiques produisant des résultats de meilleure qualité pour orchestrer la coopération entre les briques « Evalueur » et « Modificateur », et pourra s'appuyer sur des composants logiciels ayant fait l'objet d'une validation rigoureuse.

6.1.2. Présentation générale

Les concepts présentés dans ce chapitre ont tous été implantés dans un logiciel graphique interactif opérationnel, dont l'interface est présentée ci-dessous. Le logement est dessiné en mode filaire sur la partie gauche de l'image ; les paires de croix matérialisent les portes et les petits segments perpendiculaires aux murs en leur milieu indiquent l'épaisseur et la position de la matière des murs. La partie droite affiche une version plus réaliste du plan, dans laquelle les murs ont une épaisseur et le graphe de mobilité est affiché.

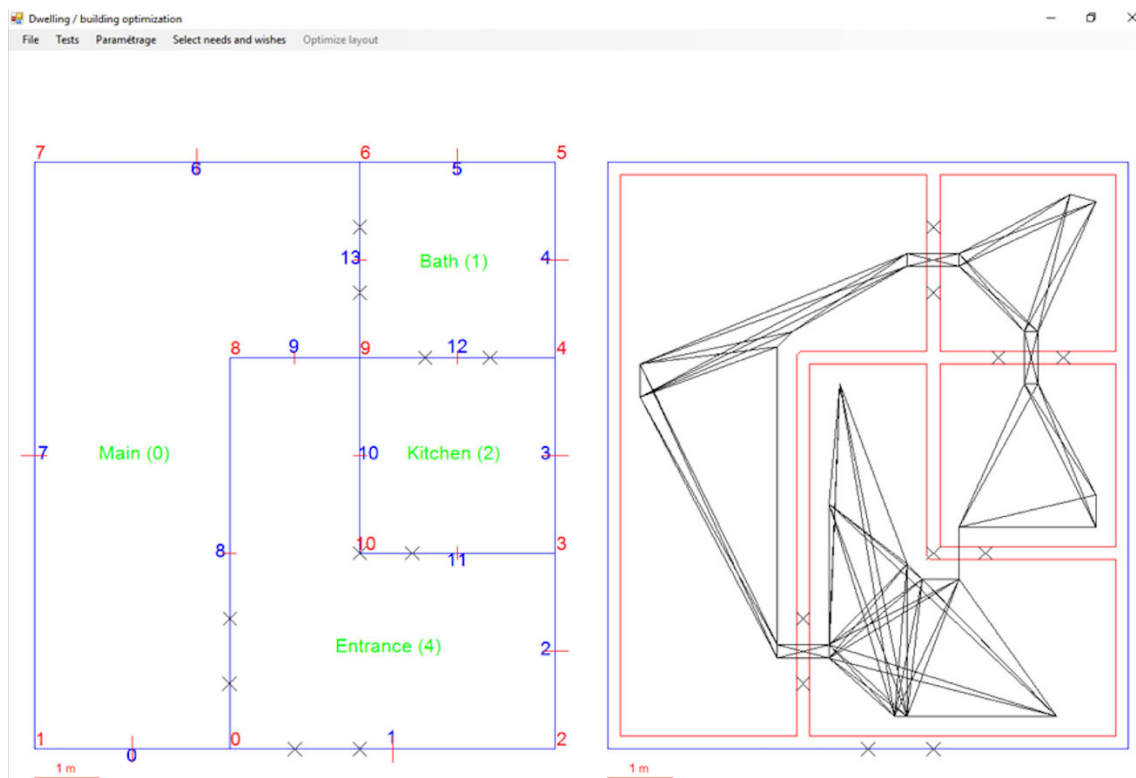


Figure 130. L'interface du logiciel

Le dernier menu (Optimize layout) permet d'appliquer l'une des quatre heuristiques au logement courant déjà aménagé (toujours visible dans la moitié gauche), préalablement chargé avec le premier menu. On choisit le lot de critères à optimiser avec le menu "Select needs and wishes". Pendant

l'optimisation, les évolutions du logement apparaissent à droite et, à la fin du processus, le logiciel propose à l'utilisateur de remplacer le logement courant par le résultat de l'optimisation ; on peut ainsi appliquer séquentiellement plusieurs heuristiques, éventuellement différentes, à un logement.

La figure ci-dessous présente le résultat de l'évaluation (et non de l'optimisation) d'un logement de type T5 au regard des critères systématiquement et automatiquement instanciés, qui traduisent soit des obligations légales, soit des évidences. Ces critères sont les suivants : tout point du logement doit être accessible à une personne non handicapée ; les pièces dans lesquelles on ne tolère pas le passage (salle à manger, bureau, chambre) ne doivent pas être adjacentes à la portion de mur extérieur qui contient la porte d'entrée ; une des pièces doit avoir plus de 9 m² ; les chambres ne doivent pas être borgnes ; il doit exister une salle de bains et, enfin, les angles des pièces doivent être supérieurs à 60 ° (cet angle est paramétrable). On constate que la note obtenue est nulle car tous les critères sont évalués à 1 sauf l'avant-dernier, qui vaut 0 car le logement n'a pas de salle de bains (voir les valeurs cerclées). Comme on l'a dit plus haut, ces critères devant impérativement être tous satisfaits, il suffit que l'un d'eux ne le soit pas pour que la note globale soit nulle.

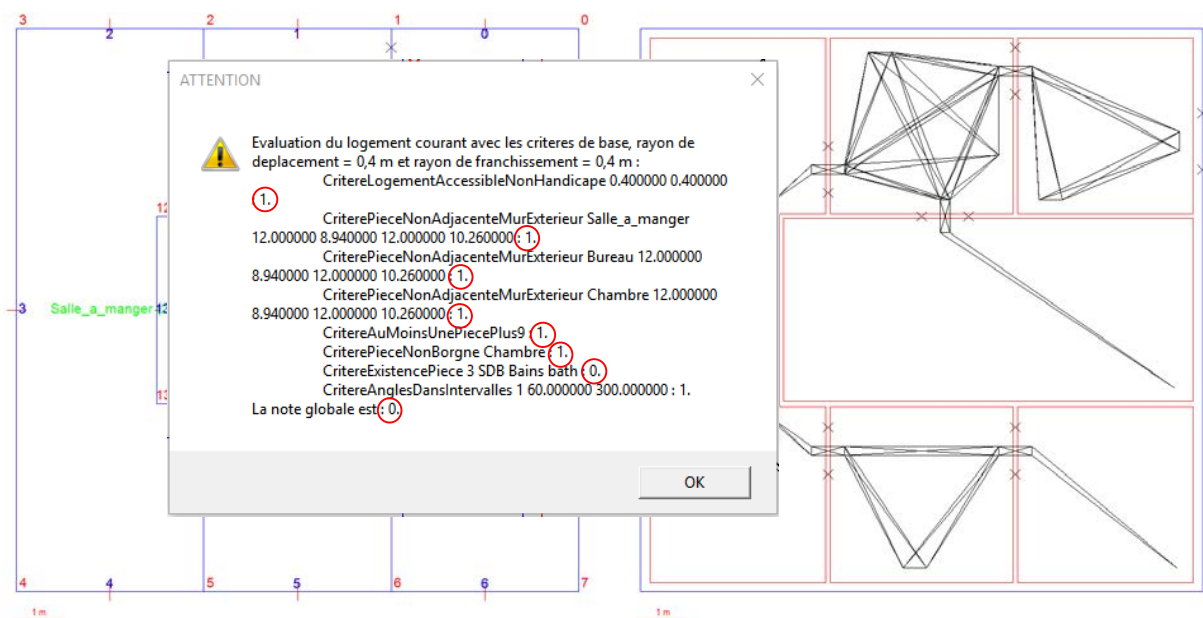


Figure 131. Évaluation d'un logement avec les critères réglementaires et de bon sens

Les heuristiques fonctionnant à nombre de pièces constant (hors couloirs), on remarque que si la proposition initiale de l'utilisateur ne contient pas toutes les pièces requises pour le logement, les heuristiques, dans leur version actuelle, ne permettent pas de « faire apparaître » les pièces manquantes. Nous revenons sur ce point à la fin de cette section d'expérimentation.

6.1.3. Illustration du bon fonctionnement des opérateurs

En fin de la section 4.2 de ce chapitre, nous avons exposé des arguments tendant à prouver que les modificateurs permettent de faire évoluer n'importe quel arrangement vers n'importe quel autre. Nous donnons en particulier une séquence de modificateurs dont l'application transformait le logement de la Figure 132-gauche en celui de la Figure 132-droite.

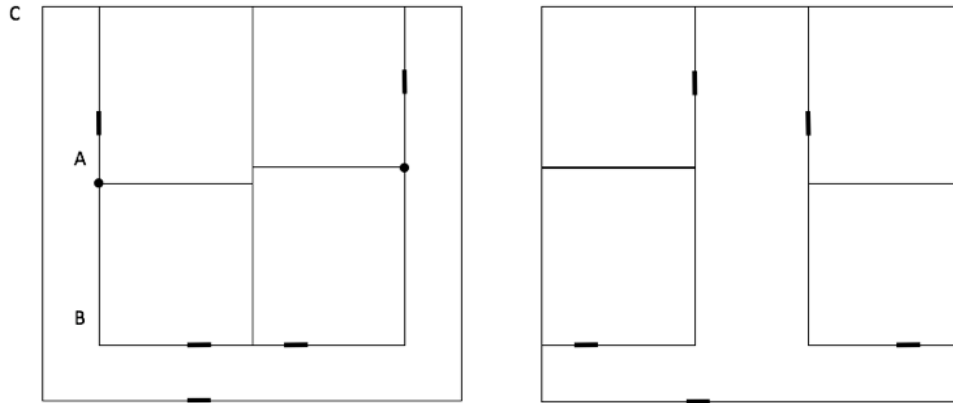


Figure 132. *Un changement profond de topologie*

Ceci représente un profond changement de la topologie du logement et il faudra que la prochaine version d'heuristiques soit capable de le proposer. Dans l'immédiat, dans le but de montrer le bon fonctionnement des opérateurs impliqués (en particulier la fusion et la scission de sommets), nous avons programmé cette séquence de modificateurs et les figures qui suivent présentent 1) la portion de code qui implémente la séquence et 2) les copies d'écran des différents logements obtenus.

```

// Insertion d'un sommet à 40 % du mur 6
LeLogementGlobal.InsererSommet(0.4, 6, RayonHabitant, &Succes, &Moitie2, &NumP1, &NumP2,
&NumSomAj);

// Insertion d'un sommet à 66 % du deuxième mur engendré par l'insertion précédente
LeLogementGlobal.InsererSommet(4. / 6., Moitie2, RayonHabitant, &Succes, &Moitie2, &NumP1,
&NumP2, &NumSomAj);

// Insertion d'un sommet à 60 % du mur 1
LeLogementGlobal.InsererSommet(0.6, 1, RayonHabitant, &Succes, &Moitie2, &NumP1, &NumP2,
&NumSomAj);

// Insertion d'un sommet à 33 % du mur 1
LeLogementGlobal.InsererSommet(2. / 6., 1, RayonHabitant, &Succes, &Moitie2, &NumP1, &NumP2,
&NumSomAj);

// Fusion des sommets 6 (coin haut-gauche de BathWC) vers 7 (coin haut gauche du logement)
LeLogementGlobal.FusionSommets(6, 7, RayonHabitant, AngleDesMursTolere, &Succes);

// Fusion des sommets 8 (coin bas-gauche de BathWC) vers 7 (en vis à vis sur le mur extérieur
gauche)
LeLogementGlobal.FusionSommets(8, 7, RayonHabitant, AngleDesMursTolere, &Succes);

// Fusion des sommets 10 (coin bas-gauche de Kitchen) vers 13 (en vis à vis sur le mur
extérieur gauche)
LeLogementGlobal.FusionSommets(10, 13, RayonHabitant, AngleDesMursTolere, &Succes);

// Fusion des sommets 4 (coin haut-droit de Bed) vers 3 (coin haut-droit du logement)
LeLogementGlobal.FusionSommets(4, 3, RayonHabitant, AngleDesMursTolere, &Succes);

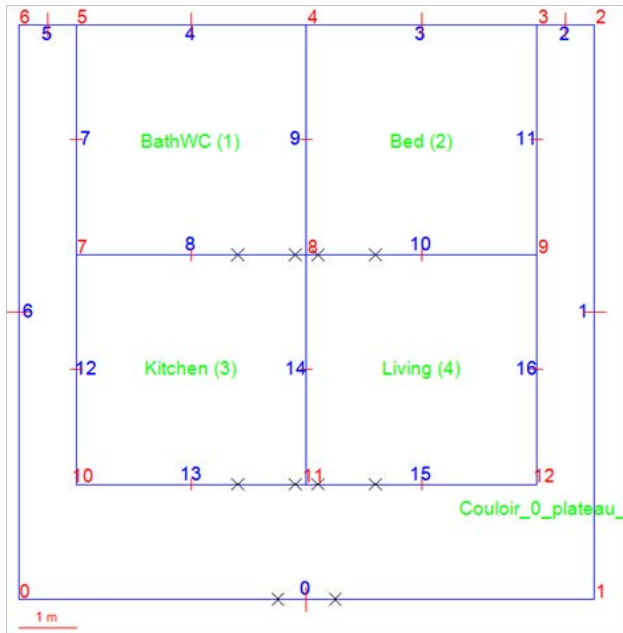
// Fusion des sommets 8 (coin bas-droit de Bed) vers 12 (en vis à vis sur le mur extérieur
droit)
LeLogementGlobal.FusionSommets(8, 12, RayonHabitant, AngleDesMursTolere, &Succes);

// Fusion des sommets 11 (coin bas-droit de Living) vers 2 (en vis à vis sur le mur extérieur
droit)
LeLogementGlobal.FusionSommets(11, 2, RayonHabitant, AngleDesMursTolere, &Succes);

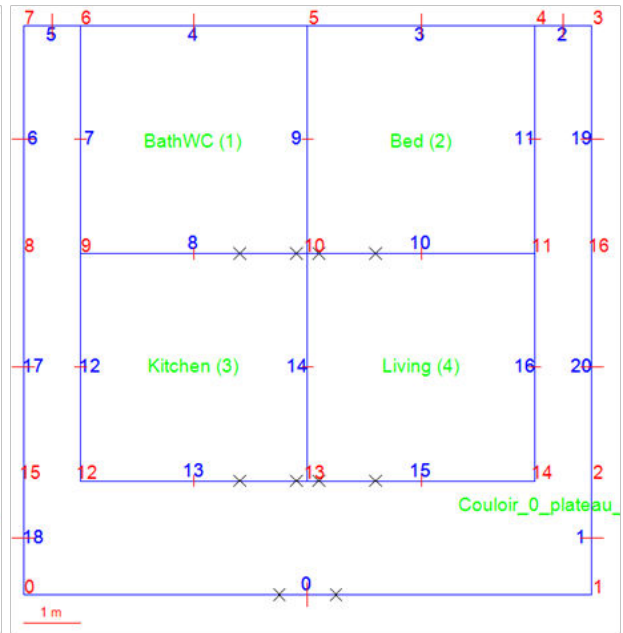
// Scission du sommet 10 (coin bas_droit de Kitchen) pour amorcer la progression du couloir
entre Kitchen et Living
vector <int> VPI = { 3 }; // Pièces non affectées par la scission (3 = couloir)
vector <int> VPC = { 5, 4 }; // Pièces affectées par la scission (5 = Living, 4 = Kitchen)
LeLogementGlobal.ScissionSommet(10, VPI, VPC, AngleDesMursTolere, RayonHabitant, &Succes);

// Scission du sommet 7 (coin haut-droit de Kitchen) pour poursuivre la progression du couloir
entre BathWC et Bed
vector <int> VPI2 = { 1, 3 }; // Pièces non affectées par la scission (1 = BathWC, 3 =
Kitchen)
vector <int> VPC2 = { 5, 4, 2 }; // Pièces affectées par la scission (5 = Couloir, 4 = Living,
2 = Bed)
LeLogementGlobal.ScissionSommet(7, VPI2, VPC2, AngleDesMursTolere, RayonHabitant, &Succes);

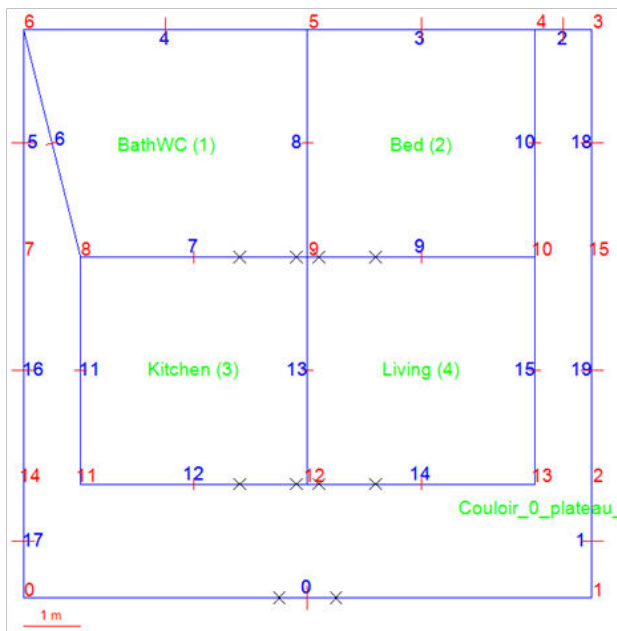
```



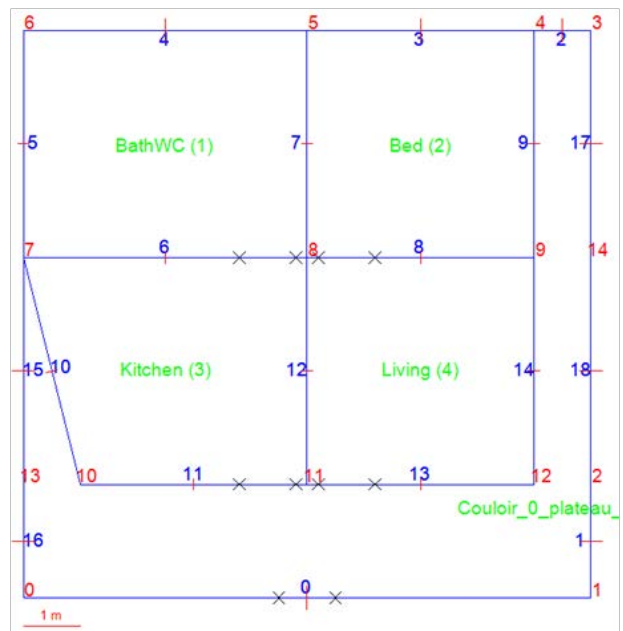
Logement de départ avec un couloir en U



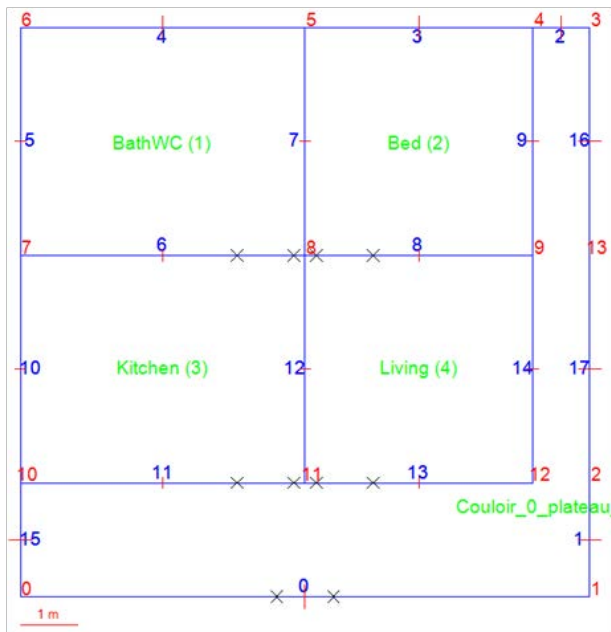
Après les 4 insertions de sommets



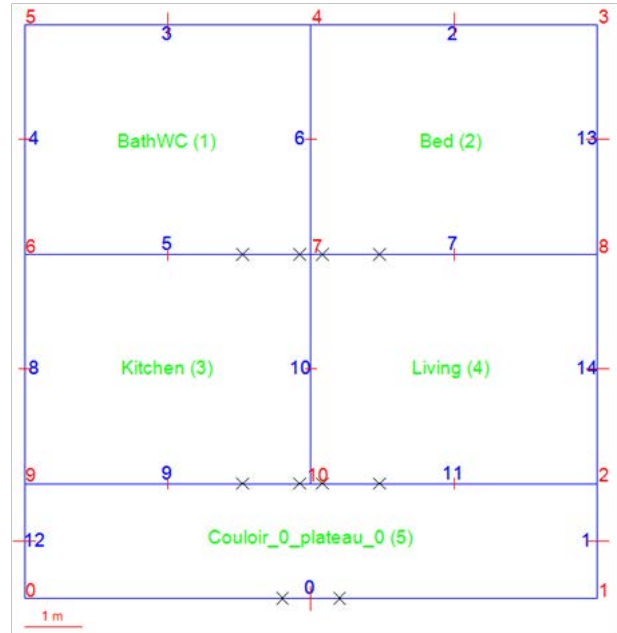
Après la première fusion



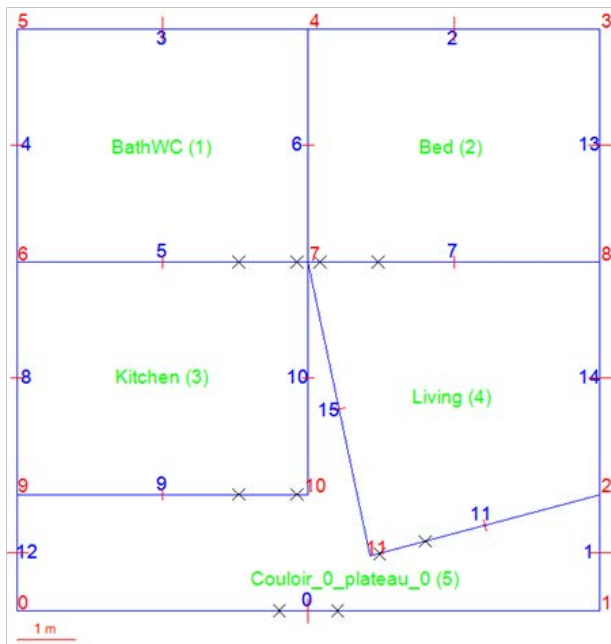
Après la deuxième fusion



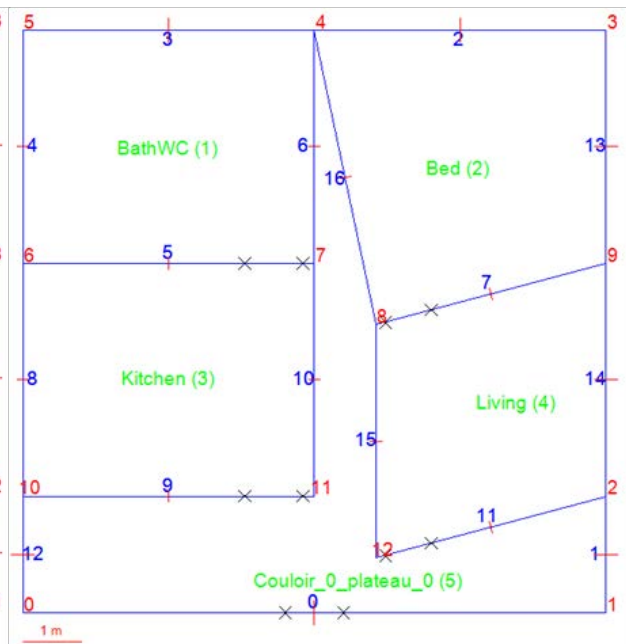
Après la troisième fusion



Après les 3 dernières fusions



Après la première scission



Après la deuxième scission

Figure 133. Différents logements obtenus

6.1.4. Influence des critères sur les agencements proposés

La maîtrise des aires des pièces est un des aspects primordiaux dans la conception d'un aménagement. Pour commencer cette section, nous soulignons que l'on ne peut se contenter d'utiliser les critères de majoration et minoration de l'aire d'une pièce si l'on souhaite *encadrer* l'aire de cette pièce. Viennent ensuite trois exemples, l'un qui illustre l'impact, sur le plan calculé, du critère d'encadrement de l'aire, un deuxième qui combine plusieurs critères et un troisième qui utilise les mêmes critères mais avec des seuils.

Précision concernant la gestion de l'aire des pièces

Rappelons que les courbes d'évaluation des critères de minoration, majoration et encadrement de l'aire d'une pièce sont les suivantes :

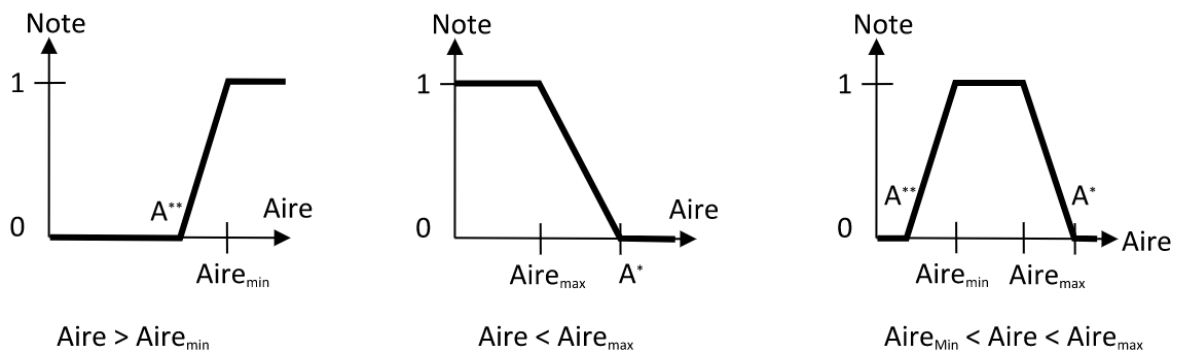


Figure 134. Les trois courbes linéaires par morceaux utilisées pour évaluer le critère de l'aire d'une pièce

Imaginons que l'on souhaite un séjour dont l'aire est comprise entre 25 et 30 m². Les courbes ci-dessus deviennent :

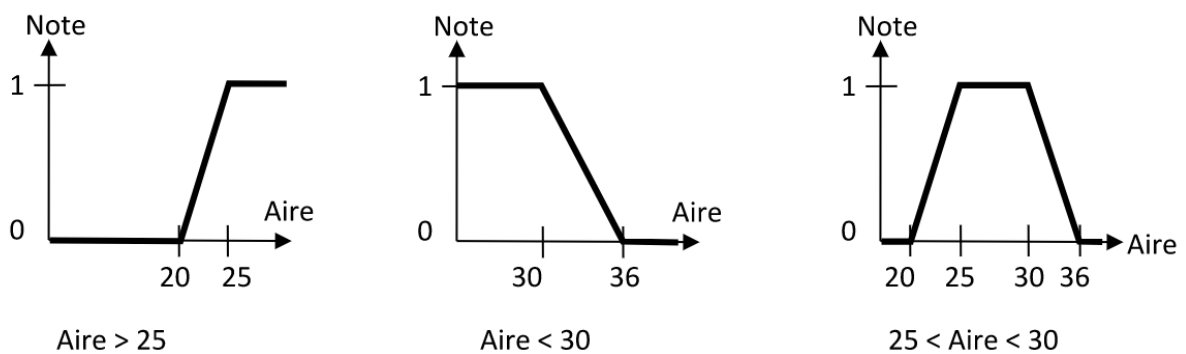


Figure 135. Les trois courbes linéaires par morceaux utilisées pour évaluer le critère de l'aire d'une pièce, comprise entre 25 et 30 m²

Pour une aire de 20 m², le troisième critère obtient une note de 0, ce qui est le résultat attendu puisque 20 m² est hors de la tolérance autorisée (par exemple $\pm 20\%$) autour de l'intervalle optimal [25, 30]. Si, à la place d'utiliser le critère d'encadrement, on instancie les deux premiers critères (minoration

avec 25 et majoration avec 30), et si on calcule leur moyenne, une aire de 20 m² est cette fois sanctionnée par la note de $\frac{0+1}{2} = 0.5$, autrement dit une note satisfaisante alors qu'elle devrait être mauvaise. On illustre ainsi la nécessité d'offrir un critère d'encadrement de l'aire *en plus* de critères de minoration et majoration.

Exemple d'utilisation du critère d'encadrement de l'aire

Les copies d'écran ci-dessous illustrent l'évolution du logement présenté en introduction quand il doit satisfaire d'une part les critères de base binaires (évoqués au début de cette section) et d'autre part un unique critère gradué :

$$18 \text{ m}^2 < \text{Aire de la pièce principale (Main)} < 23 \text{ m}^2$$

On rappelle que si les critères de base sont satisfaits, l'évaluation est autorisée à se poursuivre ; elle calcule alors les notes des autres critères et en fait la moyenne pondérée ; dans cet exemple, le seul autre critère est celui de l'encadrement de l'aire de la pièce « Main », la progression de la note est donc uniquement conditionnée par ce dernier critère.

Avant optimisation, l'aire initiale de la pièce « Main » est de 28,62 m² ; la valeur initiale du critère et la note du logement sont de 0.097.

Pour les tests qui suivent, nous avons accepté que les angles formés par deux murs consécutifs d'une pièce descendent jusqu'à 10°, alors que cette même valeur est réglée à 60° par défaut. Ce réglage par défaut a le mérite d'interdire des angles très fermés et contribue donc significativement à la qualité des formes des pièces ; la tâche pour le critère « Forme raisonnable » est alors nettement plus facile. En contrepartie, le nombre de fusions et de scissions de sommets est considérablement réduit car ces opérations engendrent fréquemment des angles inférieurs à 60°. Comme ces modificateurs sont, de notre point de vue, les facteurs principaux d'évolution de la topologie des agencements, nous avons choisi de les privilégier en réduisant l'angle minimum toléré à 10°.

Les quatre méthodes d'optimisation ont été appliquées mais le résultat de la méthode aléatoire n'est pas reporté ici : malgré 100 000 modifications (soit 16 539 insertions de sommets, 1080 fusions de sommets, 11245 déplacements de sommets, 641 scissions de sommets, 16 690 suppressions de sommets, 23 754 échanges de pièces, 11 619 insertions de portes, 7 686 déplacements de portes et 10 746 suppressions de portes), aucune amélioration n'a été découverte. Ceci confirme, si besoin était, que les chances que le pur hasard et la force brute suffisent à rencontrer des solutions intéressantes sont très faibles.

Avec la méthode 2 (premier voisin améliorant)

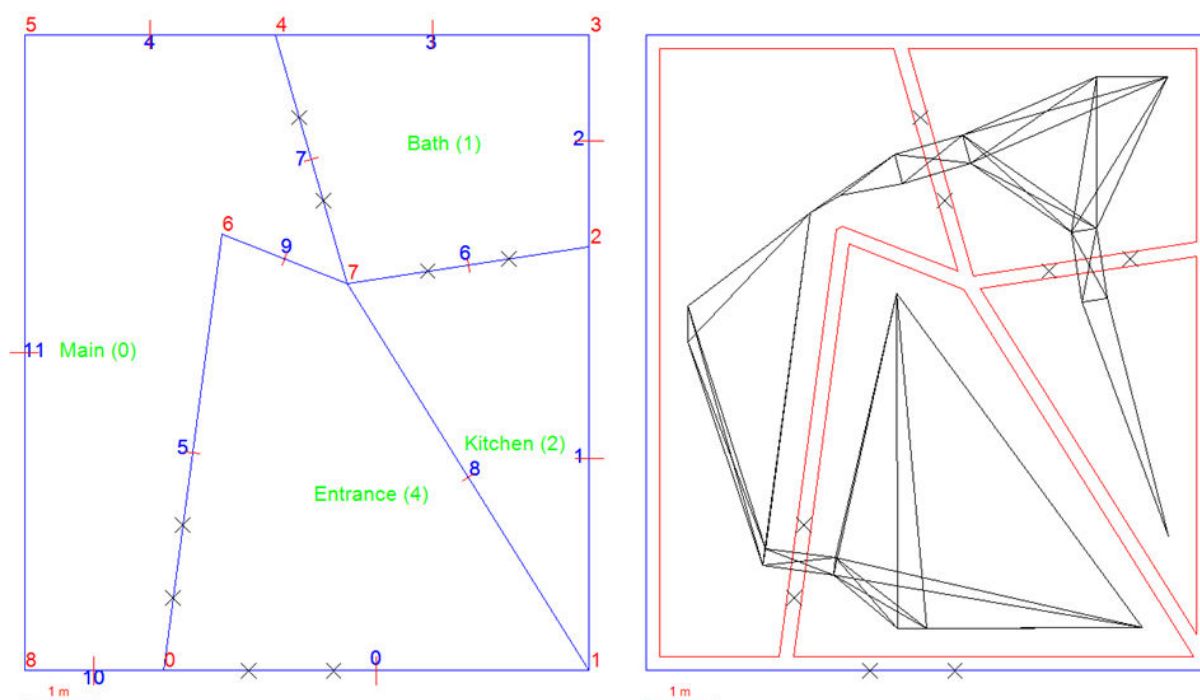


Figure 136. Le logement produit par la méthode 2 (premier voisin améliorant)

L'exécution a provoqué 39 évolutions du plan initial et a duré 6 secondes, au terme desquelles l'aire de la pièce principale était de 22,46 m². La note du logement était donc de 1, ce qui a interrompu le processus.

On constate que le fait de ne contrôler que l'aire de la pièce « Main » a les conséquences suivantes :

- la forme des pièces est déplaisante (d'où l'intérêt du critère « Forme raisonnable ») ;
- les angles sont relativement éloignés d'angles droits, ce qui compliquerait la construction et l'ameublement (d'où l'intérêt du critère qui ramène les angles dans des intervalles donnés) ;
- des murs sont quasiment alignés, il serait judicieux de les aligner (nous n'avons pas encore développé un critère spécifique en ce sens mais, dans une certaine mesure, le critère de maîtrise des angles peut aider ; il suffit d'imposer des angles proches de 180°, en plus d'angles droits, par exemple) ;
- il faut traverser la salle de bains pour atteindre la cuisine (il faudrait éviter les enfilades) ;
- il n'y a pas d'accès de l'entrée vers la cuisine (exiger que ces deux pièces partagent une porte serait une bonne idée).

Avec la méthode 3 (meilleur voisin améliorant)

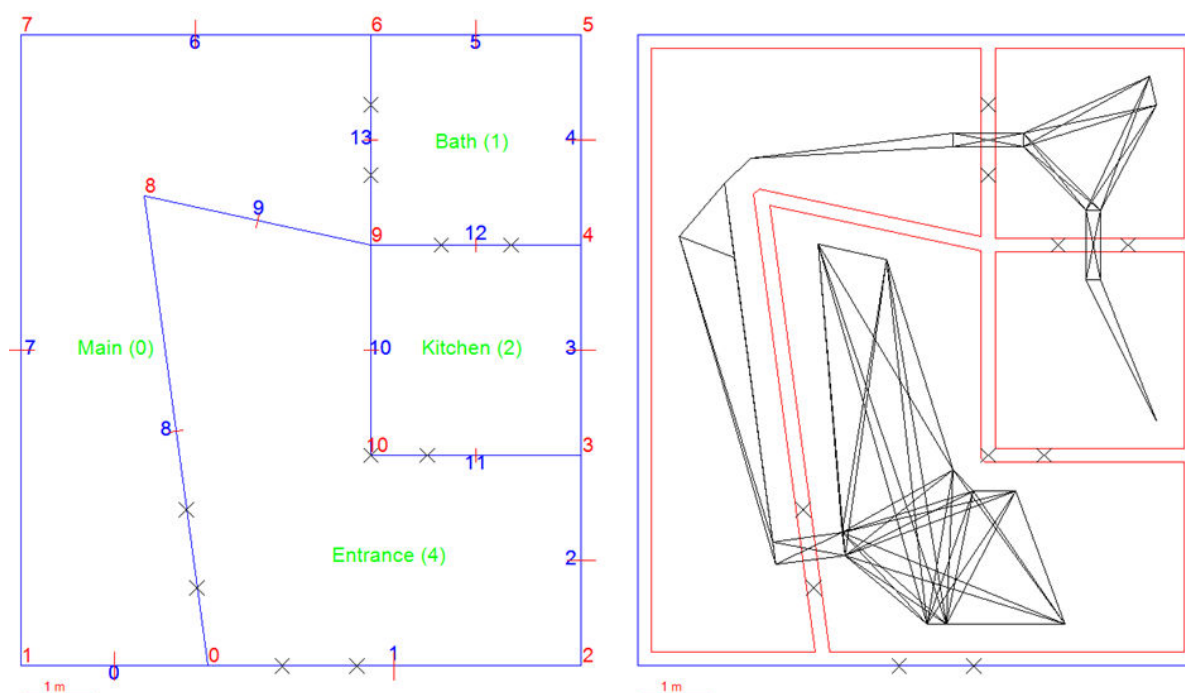


Figure 137. Le logement produit par la méthode 3 (meilleur voisin améliorant)

L'exécution a provoqué 5 évolutions du plan initial et a duré 30 secondes, au terme desquelles l'aire de la pièce principale était de 22,97 m². La note du logement était donc de 1, ce qui a interrompu le processus.

On peut faire les constats suivants :

- la disposition des pièces est à nouveau relativement proche de la disposition initiale. Nous pensons que cette absence de remise en cause profonde de la topologie de l'agencement est causée par la mutuelle neutralisation des modificateurs (les statistiques confirment que tous les modificateurs sont utilisés et ceci dans des proportions proches). Pendant les exécutions, il arrive que l'on voie apparaître une amorce de couloir mais elle disparaît généralement après quelques instants, à cause d'opérateurs ayant un effet symétrique ;
- ceci confirme un besoin pour d'autres heuristiques, dans lesquelles le choix d'un modificateur sera (aussi) conditionné par les modificateurs précédemment appliqués.

Avec la méthode 4 (recuit simulé)

Le temps de fonctionnement du recuit, c'est inhérent à sa nature, est très variable. Pour ce jeu de données, les durées ont varié de 36 secondes à plus de 3 minutes, auxquelles il faut ajouter un temps de calibrage toujours proche de la minute. Aucune tentative n'a échoué. La solution affichée ci-dessous est le résultat de 1346 évolutions du plan initial, dont certaines étaient des améliorations et d'autres des détériorations ; elle a duré un peu moins de 3 minutes, au terme desquelles l'aire de la pièce principale était de 18,27 m². La note du logement était donc de 1, ce qui a interrompu le processus. On remarque que la pièce principale s'est déplacée sur le côté droit du logement et que la salle de bains a d'une part grandie considérablement et d'autre part migré du côté gauche :

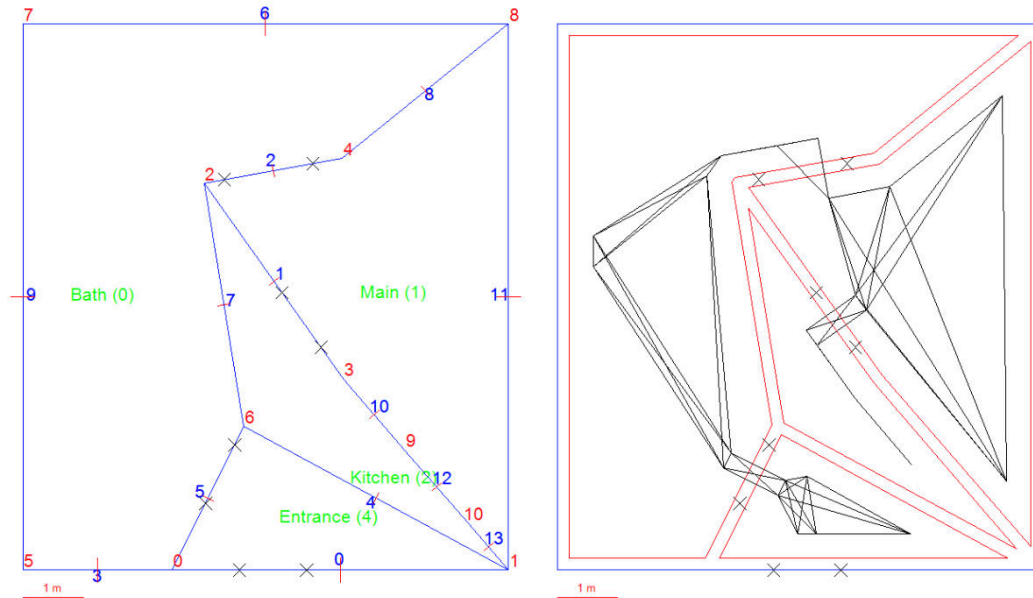


Figure 138. Un logement produit par la méthode 4 (le recuit simulé)

Avec davantage de critères

Voici à présent la progression du même logement initial, cette fois avec les critères supplémentaires suggérés dans le test précédent : réduire le nombre d'enfilades, appliquer « FormeRaisonnée » à chaque pièce et demander que les angles des pièces s'inscrivent dans les intervalles, en degrés, [85,95], [175,185] et [265,275] ; autrement dit, on tente de les rapprocher d'angles droits (90° et 270°) ou de l'angle plat (180°). On demande aussi que l'entrée partage une porte avec la cuisine, la pièce principale et la périphérie du logement.

Par ailleurs, nous avons *successivement* appliqué à l'agencement initial la méthode 2, la méthode 3 et la méthode 4 dans cet ordre. Il ne s'agit donc pas d'une comparaison des résultats des trois méthodes mais d'une illustration de la possibilité de les enchaîner.

La valeur initiale du logement était : 0.661.

La méthode 2 était paramétrée avec une taille de voisinage de 1000 (pour chaque solution, on est susceptible d'étudier 1000 voisins mais on s'arrête dès qu'un améliorant est trouvé). 76 déplacements ont été effectués, le tout en 36s. La valeur s'est améliorée jusqu'à 0,854 et la solution correspondante était la suivante :

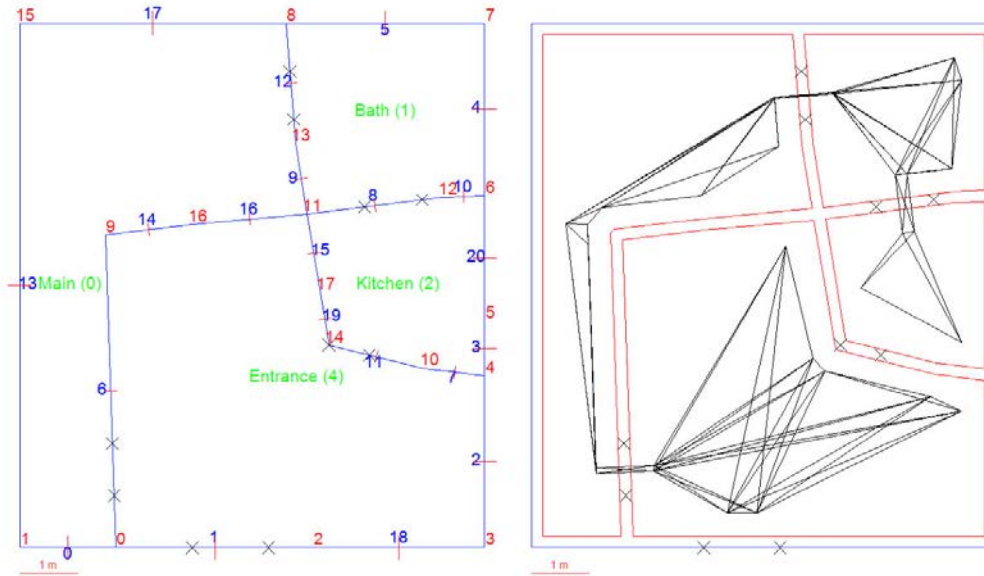


Figure 139. Le logement produit par la méthode 2

La méthode 3 était paramétrée avec une taille de voisinage de 1000 (pour chaque solution, on étudie 1000 voisins et on conserve le meilleur). 8 déplacements ont été effectués, le tout en 1' 48". La valeur s'est améliorée jusqu'à 0,857 et la solution correspondante était la suivante :

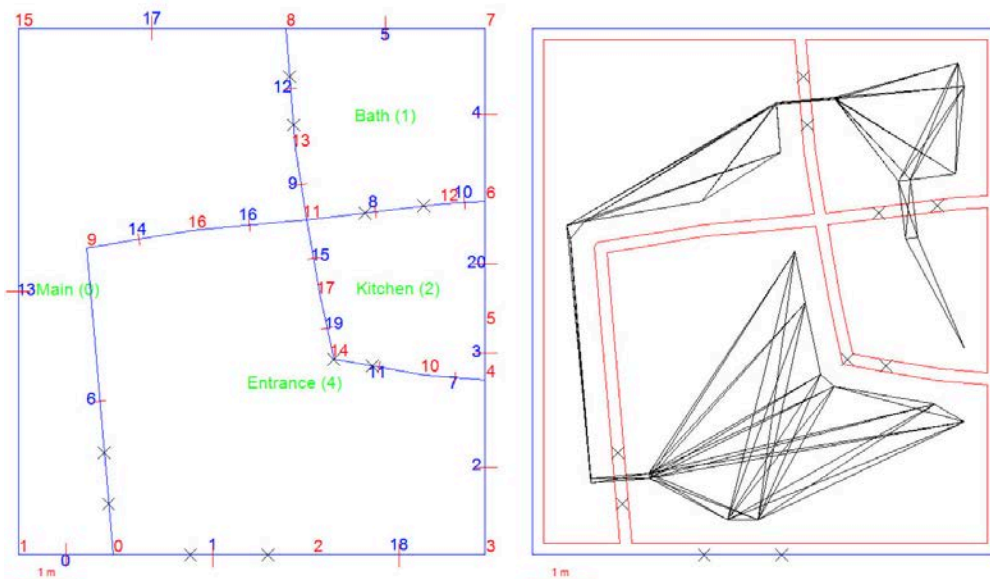


Figure 140. Le logement produit par la méthode 3

Il ne faut pas s'étonner que la méthode 3 ait pu améliorer (légèrement) le résultat de la méthode 2 : cette dernière s'arrête quand elle ne trouve pas de voisin améliorant pour la solution courante. La méthode 3 recommence avec cette solution courante et parcourt à nouveau les voisins de cette solution pour trouver le plus améliorant. Si elle rencontrait les mêmes voisins, elle échouerait ; mais comme les voisins sont obtenus en appliquant un modificateur choisi aléatoirement et comme les paramètres du modificateur sont eux aussi aléatoires (dans des bornes raisonnables), de nouveaux voisins sont rencontrés.

Le recuit était paramétré comme suit : un palier autorisait 1000 tentatives de modifications et 20 paliers étaient demandés.

Au cours de l'exécution (pendant le neuvième palier), la solution suivante a été rencontrée :

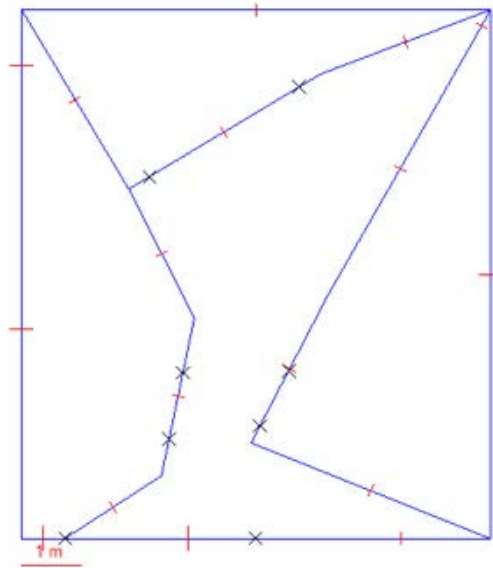


Figure 141. Solution intermédiaire intéressante

Nous la trouvons intéressante car elle annonce une disposition très différente des pièces du logement initial, avec une entrée qui dessert les trois autres pièces, ce qui est probablement une solution raisonnable dans ce type de petit logement. La forme des pièces n'est pas encore satisfaisante mais nous constatons malgré tout que même une métaheuristique simple comme le recuit est capable de proposer des solutions innovantes. C'est précisément dans cet optique que le processus d'amélioration a été conçu : proposer des agencements qui, sans nécessairement être parfaits d'un point de vue géométrique, sont une force de suggestion pour l'utilisateur du logiciel.

Le recuit s'est finalement terminé en 27 minutes en n'ayant pas amélioré la solution proposée par la méthode 3. Ceci amène la réflexion suivante : la solution la mieux notée n'est pas forcément celle qui amène les plus intéressantes suggestions. Une extension intéressante de ces travaux serait donc de développer une métrique pour mesurer la différence entre les topologies de deux solutions ; cela permettrait de proposer, en plus de la solution la meilleure au sens actuel, des solutions moins bien notées mais très éloignées dans leur conception.

Avec des contraintes de seuils

Le seuil est la valeur sous laquelle la note d'un critère est ramenée à 0. Sans réduire le critère à un objet binaire et absorbant (comme le sont les critères légaux), cela casse l'aspect continu de l'évaluation du critère concerné et accentue la dégradation de la moyenne globale lorsque la note du critère passe sous le seuil.

Exemple :

Si on a deux critères évalués à 0.4 et 0.6 et si le deuxième a un seuil à 0.5

alors la moyenne n'est plus de $(0.4 + 0.6) / 2 = 0.5$ mais de $(0 + 0.6) / 2 = 0.3$.

Nous avons repris le scénario précédent (enchaînement des méthodes 2, 3 et 4) en utilisant le même paramétrage des méthodes et en ajoutant les seuils suivants :

- pour les quatre instances de « forme raisonnable » : 0.8 ;
- pour la réduction du nombre d'enfilades et le maintien des angles dans les trois intervalles mentionnés plus haut (angles droits ou plat) : 0.9.

La note initiale du logement est de 0.61, une valeur différente de la note initiale précédente à cause du seuillage. Voici ce que nous avons obtenu.

Après application de la méthode 2 :

79 déplacements ont été effectués, le tout en 66'. La valeur s'est améliorée jusqu'à 0,7 et la solution correspondante est la suivante :

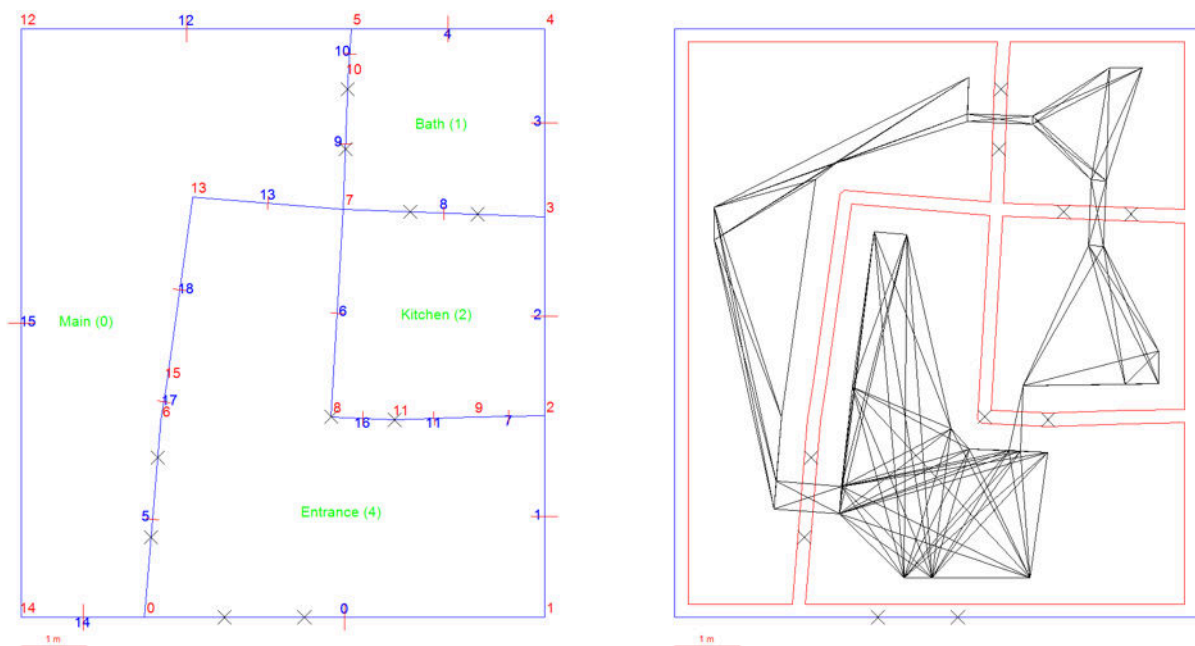


Figure 142. Le logement produit par la méthode 2

On remarque que la présence des seuils a eu les conséquences positives suivantes : l'enfilade a disparu, les formes des pièces sont de meilleure qualité et les angles aigus ont disparu.

Après application de la méthode 3 :

À l'inverse de ce qui s'est passé dans le scénario précédent, la méthode 3 n'a pas réussi à améliorer le résultat de la méthode 2 (aucun voisin améliorant n'a été trouvé). Il était donc intéressant de voir si la faculté du recuit de tolérer des dégradations, surtout en début d'exécution, allait lui permettre de quitter cet optimum local obtenu avec la méthode 2.

Après application de la méthode 4 (recuit simulé) :

Le recuit a effectivement quitté l'optimum local (au point de rencontrer des solutions de valeur 0.2) mais s'est terminé après 2'48" sans rencontrer de solution meilleure que la solution initiale.

6.1.5. Synthèse via amélioration

On a dit plus haut que la proposition initiale de l'architecte, celle que le logiciel doit améliorer, doit contenir toutes les pièces attendues (hors couloir). On peut alors espérer que les heuristiques ajoutent la dose de couloirs, portes, déformations, déplacements ... qui rendra le plan réaliste et, pourquoi pas, conforme aux attentes de l'architecte ou du futur occupant. Mais pour peu que l'on parvienne à construire automatiquement un premier plan grossier contenant toutes les pièces nécessaires (du type de celui de la Figure 143), on peut envisager que l'outil d'amélioration d'un plan existant soit utilisé pour réaliser un outil de synthèse, concurrent de celui qui a été présenté au chapitre précédent.

La réalisation du plan initial ne présente guère de difficultés ; il suffit par exemple de placer autant de carrés que de pièces souhaitées le long du contour du logement, ces carrés devant être de taille suffisamment petite pour qu'ils tiennent dans le contour du logement sans se recouvrir les uns les autres (Figure 143).

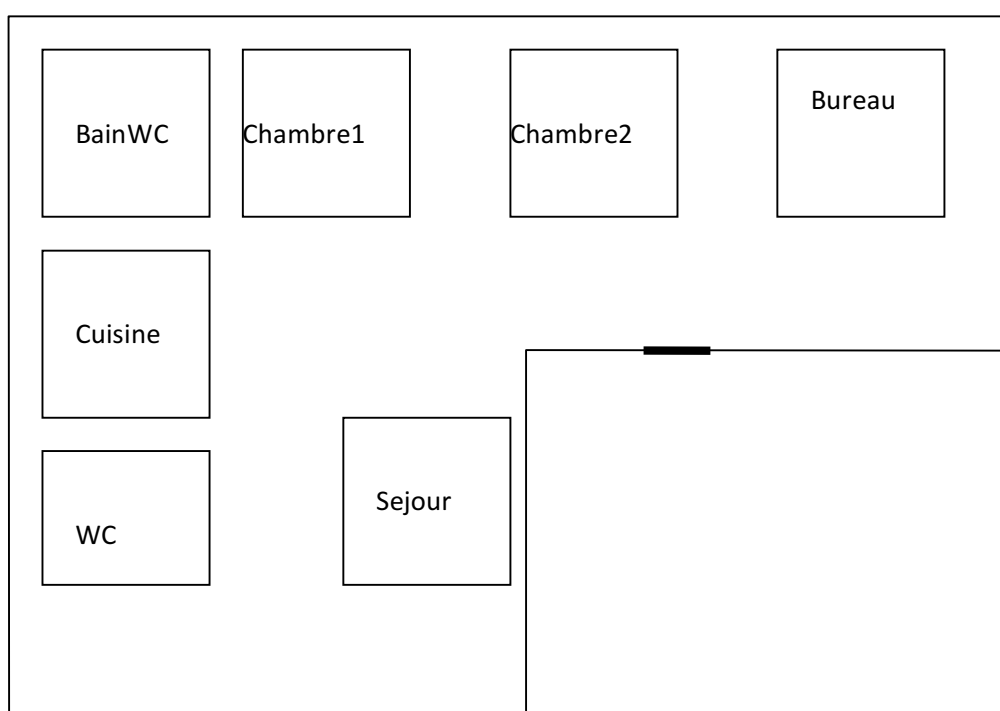


Figure 143. Un plan grossier permettant d'utiliser l'outil de l'amélioration pour faire de la synthèse

Cette idée d'utiliser l'optimisation pour synthétiser un aménagement n'a pas encore fait l'objet d'une expérimentation poussée ; elle constitue l'une des extensions possibles de ce travail.

7. Bilan

La méthode d'amélioration d'un agencement commence par l'acquisition du plan à améliorer et des besoins et souhaits de l'occupant ; ensuite, elle évalue à chaque modification le degré d'adaptation de l'aménagement à l'occupant jusqu'à soit atteindre un aménagement satisfaisant soit échouer parce qu'il n'y a plus d'amélioration ou parce que la durée maximum est atteinte.

Quatre composants sont nécessaires à la mise en œuvre de cette méthode d'amélioration :

- la modélisation du logement : elle est faite à l'aide de trois structures de données : une structure géométrique 2D qui permet de conserver toutes les informations liées à la position et aux dimensions des pièces, murs, portes, etc. ; un graphe appelé « graphe de mobilité » qui recense tous les déplacements rectilignes possibles de l'occupant dans le logement entre des points caractéristiques (centre d'une porte, point d'une pièce le plus éloigné des portes, sommet concave d'une pièce ou d'un couloir), tout itinéraire dans le logement étant approché par une succession de ces déplacements élémentaires ; un deuxième graphe appelé « graphe d'adjacences » qui recense les adjacences (au sens partage d'une porte) entre les pièces. Ces trois structures de données permettent une description plus complète du logement pour une meilleure évaluation des besoins et souhaits de l'occupant. Par exemple, si l'évaluation d'un besoin ou d'un souhait de l'occupant concerne un aspect touchant au déplacement de ce dernier dans le logement, le graphe de mobilité sera la structure sur laquelle l'évaluation sera faite car c'est la structure qui recense tous les déplacements possibles de l'occupant ;
- la modélisation des règles liées à l'habitat et des besoins et souhaits de l'occupant. Tous sont exprimés par des combinaisons de *critères*, à savoir dans un format exploitable d'un point de vue informatique. Par exemple, deux des critères que nous avons cités sont : « l'aire de la pièce i doit être supérieure à une aire donnée » et « le circuit ($pièce\ i, pièce\ j, \dots, pièce\ 0$) ne doit pas dépasser une longueur donnée » ;
- l'évaluation de l'adaptation d'un aménagement à l'occupant : elle consiste à vérifier le degré d'adaptation d'un aménagement aux besoins et souhaits de l'occupant. C'est un traitement en trois niveaux : 1) Les opérateurs d'évaluation sont des opérateurs de bas niveau. Par exemple, la distance géométrique entre deux points A et B, le coût d'un aménagement, l'orientation d'une pièce, le plus court chemin entre deux pièces, la connexité d'un graphe... ; 2) L'évaluation d'un critère, qui consiste à évaluer à l'aide des opérateurs un critère qui traduit des règles, des besoins ou des souhaits de l'occupant. Par exemple, le critère « chaque point du logement doit être accessible à partir de n'importe quel autre point » revient à appliquer l'opérateur de base « connexité d'un graphe » au graphe d'adjacence de l'aménagement ; 3) l'évaluation du logement, qui consiste à combiner les évaluations de tous les critères en une note globale du logement, afin de mesurer son degré d'adaptation au futur occupant ;
- l'ensemble des opérateurs de modification d'un logement s'appuient aussi sur d'autres opérateurs de bas niveau. Ces derniers sont conçus pour permettre tout type de modifications : déformation / déplacement d'une pièce, ajout / suppression de portes, ajout / suppression de couloirs.

La résolution est faite avec quatre méthodes approchées qui s'appuient toutes sur ces quatre composants. Une première méthode (recherche locale aléatoire) consiste à faire une série de modifications aléatoires à partir d'une solution initiale. La deuxième méthode (First Improvement Hill Climbing) n'accepte une modification que si elle améliore la solution ; elle s'arrête entre autres quand aucune solution voisine n'est améliorante. La méthode suivante (Best Improvement Hill Climbing) choisit à chaque itération la meilleure solution améliorante parmi un ensemble de solutions voisines. Elle aussi s'arrête quand aucune voisine améliorante n'est trouvée. La dernière méthode est un recuit simulé, dont l'avantage est de ne pas être bloquée sur le premier maximum local trouvé, à l'inverse des méthodes précédentes.

8. Discussion

Nous concluons ce chapitre avec quelques commentaires sur la méthode d'amélioration d'un logement qui y est proposée et quelques possibilités d'étendre le travail qui a été réalisé.

Le processus proposé est pour l'instant limité au traitement d'un seul niveau d'un logement, il doit donc être réutilisé autant de fois qu'il y a de niveaux. Néanmoins, il est possible de l'adapter pour prendre en compte plusieurs niveaux. Pour cela, il suffit d'ajouter une paire d'arcs dans le graphe de

mobilité pour chaque dispositif de changement de niveau (escalier, ascenseur, rampe), chacun des deux arcs correspondant à un sens de parcours. Ces arcs sont étiquetés avec la longueur ou, pour une prochaine version, avec l'effort représenté par le franchissement du dispositif. Ceci fait, les calculs d'itinéraires entre des niveaux différents est immédiat. Pour le placement des pièces, soit on sait à l'avance quelles sont les pièces qui appartiendront à chaque niveau, et dans ce cas, le placement se fera par niveau, soit on ajoute des critères supplémentaires sur le placement pour imposer par exemple qu'une pièce soit sur tel niveau ou sur le même niveau que telle autre pièce ; quoi qu'il en soit, le schéma général de la méthode n'est aucunement remis en cause.

Rappelons aussi que l'objectif premier des aménagements générés n'est pas la perfection géométrique mais qu'ils doivent servir de source d'inspiration pour l'utilisateur du logiciel et exigent probablement des retouches de sa part.

Nous avons défendu dans le point 5.2 l'idée que la panoplie d'opérateurs de modification que nous proposons suffit à atteindre n'importe quel aménagement adapté en partant d'un quelconque autre aménagement adapté. Un autre sujet de préoccupation est la probabilité que des modifications tirées et paramétrées aléatoirement permettent d'atteindre tôt ou tard un optimum. Il est même probable que les deux approches strictement améliorantes (approches 2 et 3) n'atteignent que des maximums locaux car il faut vraisemblablement, pour atteindre un maximum global, accepter de détériorer, par moment, la solution courante.

A moyen terme, il y a plusieurs stratégies que nous aimerions tester car nous pensons qu'elles amélioreraient les chances d'atteindre un optimum de meilleure qualité :

- quand la méthode « se lance » dans une voie, il faut prendre des mesures, à l'instar de la recherche tabou, pour que le hasard ne la fasse pas immédiatement revenir sur ses pas. Par exemple, quand un sommet a été inséré, il faut interdire sa suppression pendant un long moment pour que l'algorithme ait le temps de tester toutes les - concrètement une part significative des - possibilités apportées par l'arrivée de ce sommet. De même, quand deux sommets se sont rapprochés, il faut tenter de les fusionner, pour continuer d'explorer les possibilités offertes par leur rapprochement. Citons également la possibilité d'insérer un couloir entre deux ou plusieurs pièces accolées, pour changer drastiquement la topologie du logement ;
- on peut aussi envisager des modifications plus intelligentes encore, guidées cette fois par les résultats de l'évaluation du logement. Illustrons cette approche avec deux exemples. Dans le cas où l'on constate, pendant l'évaluation, qu'une pièce ne peut être atteinte (aucun chemin n'y mène), on peut privilégier l'application à cette pièce des opérateurs d'ajout et d'élargissement d'une porte. Par ailleurs, si une enfilade non désirée est détectée, c'est-à-dire qu'on est obligé de traverser une pièce $P1$ pour atteindre une autre pièce $P2$, plusieurs stratégies peuvent être envisagées : ajouter une porte dans un mur que $P2$ partage avec un couloir, échanger $P1$ et $P2$ ou encore insérer un couloir entre $P2$ et une de ses voisines, dans l'espoir de rendre $P2$ accessible autrement que par $P1$.

Nous avons expliqué que la modélisation des attentes concernant l'aménagement, autant d'un point de vue légal qu'au niveau des attentes du futur occupant, se fait au travers de critères. Ces critères sont des classes qui héritent toutes d'une classe mère, ce qui permet une très facile extension de la panoplie des critères par un informaticien. Potentiellement, le nombre de critères qui peuvent ainsi être ajoutés est considérable ; c'est un premier niveau *d'extensibilité*.

Nous souhaitons également mettre en place l'évaluation des itinéraires par la notion d'effort plutôt que par la seule notion de longueur. Nous avons par exemple évoqué le cas des escaliers, dont le franchissement dans le sens de la montée est beaucoup plus pénible que dans le sens de la descente.

Mais même un itinéraire sur un sol plat nécessite cette extension : on peut par exemple imaginer qu'une personne cardiaque marche à une allure normale pendant un temps puis doit ralentir l'allure pour récupérer puis ré-accélère progressivement. Il est donc un peu erroné, pour certaines pathologies, de ne raisonner que sur la longueur ; une prise en compte de l'effort cumulé et de l'effort instantané serait probablement plus représentative de la difficulté réelle que représente un itinéraire.

On ne peut pas *ne pas* penser qu'un logement adapté à une personne est un logement moins adapté à de nombreuses autres personnes, autrement dit un logement qu'il sera plus difficile de revendre. L'optimisation de l'aménagement comme elle est définie dans ce manuscrit se heurte donc à l'optimisation de la valeur marchande du logement. Pour rendre les deux notions compatibles, il faudra impérativement investir plus encore dans la notion de modularité des logements, à savoir la possibilité de transformer facilement l'agencement au moment du changement d'occupant. C'est un des thèmes développés dans le mouvement "design for all", qui vise à concevoir des appareils, du mobilier, des aménagements domestiques et urbains, des signalétiques ... qui conviendraient à toutes les personnes, sans distinction de taille, d'âge ou de handicap. Le lecteur intéressé par ces aspects peut, entre autres choses, trouver la liste des projets européens consacrés au design for all dans [86] ; contrairement à ce que suggère le titre (Design For All (in ICT)), le document ne concerne pas que l'accès pour tous aux technologies de l'information et de la communication.

Pour conclure, on peut souligner que la méthode d'amélioration d'un logement peut facilement être adaptée pour devenir une méthode de synthèse. L'aménagement initial disparaît des données, il est *remplacé* par le nombre N de pièces, leurs natures respectives (cuisine, salle de bains, chambre 1 ...) et le contour extérieur du logement. Les autres données (contraintes réglementaires, besoins, souhaits) restent les mêmes. Il est alors facile de placer N polygones distincts dans le contour du logement et d'appliquer au logement factice ainsi obtenu le processus d'amélioration décrit dans ce chapitre.

Conclusion et perspectives

Dans cette thèse, nous nous sommes intéressés à l'agencement d'un espace intérieur, en particulier celui d'un logement, dans le but de l'adapter aux besoins et souhaits de son futur occupant en situation de handicap ou non. Notre principal objectif était de prouver la possibilité d'utiliser des techniques d'optimisation et de modélisation mathématique et géométrique pour concevoir un outil d'aide à la décision, capable d'aider les architectes à aménager des espaces intérieurs, en prenant en compte plus de critères que ce qu'ils peuvent considérer sans une assistance logicielle.

Un tour d'horizon des problématiques en lien avec le handicap et l'aménagement au sens de la loi du 11 février 2005 pour l'égalité des droits et des chances a été effectué dans le premier chapitre. Un cadre théorique a ensuite été proposé pour notre problématique, en faisant un parallèle entre le problème d'agencement, incluant des contraintes traduisant les besoins et souhaits de l'occupant, et un problème de type bin-packing à deux dimensions intégrant des contraintes d'ordre géométrique et fonctionnel (non chevauchement, une distance minimale entre deux objets, le placement d'un objet autour d'une zone en particulier ...). Plusieurs travaux en lien avec notre problématique ont été résumés, ce qui a mis en évidence le fait que toutes les approches décomposent le problème global en plusieurs sous-problèmes traités consécutivement et se limitent aux pièces de forme rectangulaire. L'obligation de garantir l'existence d'itinéraires entre tous les points du logement, ce qui revient à dénombrer et placer judicieusement des couloirs, est traitée à des degrés différents mais ne va jamais jusqu'à prendre la longueur de ces itinéraires en compte. Les portes ne sont jamais placées autrement que manuellement ou de manière arbitraire.

Dans le chapitre qui a suivi, nous avons détaillé l'ensemble des méthodes géométriques spécifiques à la gestion d'un logement. Cette partie était nécessaire à la mise en œuvre de l'approche développée dans le Chapitre IV dédié à l'amélioration d'un aménagement. Les algorithmes implémentés sont tous issus ou inspirés de la littérature. La structure de données du logement n'étant pas un maillage au sens classique, leur mise en œuvre a cependant souvent exigé une adaptation (c'est par exemple le cas du calcul de l'offset d'un polygone et de la détection automatique des couloirs). Ces considérations, le fait que la compréhension de ces algorithmes est nécessaire à celle des méthodes d'amélioration d'un agencement et l'effort considérable que nous a coûté la mise en œuvre de ces algorithmes, nous ont conduits à leur consacrer un chapitre.

Le troisième chapitre a détaillé notre première contribution à savoir, la synthèse d'un aménagement. L'objectif était d'aménager un lieu à partir des dimensions du contour extérieur du logement, du nombre souhaité de pièces, des intervalles autorisés pour les deux dimensions de chaque pièce, de contraintes d'inclusion de certaines zones par certaines pièces et de contraintes d'orientation sur certaines pièces, en plus des besoins et souhaits de l'habitant. Cet objectif a été atteint. L'approche développée est décomposée en trois étapes, il s'agit de :

1. placer et dimensionner les pièces. L'objectif de cette étape est la génération de solutions qui respectent les contraintes de placement et les contraintes de dimensionnement, imposées par les besoins et souhaits de l'occupant, tout en maximisant la surface utile de l'aménagement. Entre chaque paire de pièces, un espacement minimum équivalent au diamètre de l'occupant est assuré. Cet espacement est considéré comme un couloir potentiel qui est conservé totalement, partiellement ou supprimé lors de la dernière étape. Cette approche a été modélisée en utilisant la programmation mathématique en nombres mixtes. La résolution a été faite à l'aide du solveur IBM-Cplex ;

2. l'étape 2 consiste à placer une porte par pièce. Le placement est dicté par un ensemble d'itinéraires que l'occupant a besoin ou/et souhaite maximiser ou minimiser. Cette étape est modélisée à l'aide d'un programme linéaire en nombres entiers, qui s'appuie lui-même sur un graphe décrivant tous les déplacements élémentaires possibles dans le logement. La résolution est assurée à nouveau par le solveur IBM-Cplex ;
3. la dernière étape consiste à supprimer les couloirs créés par la première étape et rendus inutiles par la seconde étape. Basée sur une représentation géométrique de la solution calculée, l'algorithme proposé parcourt, dans un ordre heuristique, l'ensemble des couloirs et supprime ceux qui ne servent à aucun des itinéraires entre deux pièces du logement.

Le dernier chapitre a présenté notre seconde contribution, qui porte sur l'amélioration d'un aménagement existant. La méthode prend, en entrée, un plan initial et des besoins et souhaits de l'occupant. Elle itère ensuite sur la modification du logement et son évaluation.

Quatre composants ont été nécessaires à la mise en œuvre de cette méthode :

- la modélisation du logement : elle est faite à l'aide d'une structure de données 2D et de deux graphes : un graphe de mobilité pour recenser tous les déplacements rectilignes possibles de l'occupant dans le logement entre des points dits caractéristiques (points d'accès à une porte, sommet concave d'une pièce ...) ;
- la modélisation d'une part des règles liées à l'habitat et d'autre part des besoins et souhaits de l'occupant : toutes sont exprimées par des combinaisons de ce que nous avons appelé des *critères*, pour les rendre exploitables d'un point de vue informatique ;
- l'évaluation d'un aménagement : elle vérifie le degré d'adaptation d'un aménagement aux besoins et souhaits de l'occupant. C'est un traitement en trois niveaux : 1) les opérateurs d'évaluation, par exemple la distance entre deux points A et B ; 2) l'évaluation d'un critère, qui s'appuie sur les opérateurs d'évaluation pour attribuer une note entre 0 et 1 ; 3) l'évaluation du logement, qui combine les notes des critères d'évaluation en une note globale, afin de mesurer son degré d'adaptation au futur occupant ;
- la modification d'un logement. Cette modification consiste à appliquer l'un des modificateurs que nous avons développés. Ces derniers sont conçus pour permettre tout type de modifications : déformation / déplacement d'une pièce, ajout / suppression de portes, ajout / suppression de couloirs.

Le processus itératif décrit plus haut a été instancié de quatre manières différentes. Une première méthode (recherche locale aléatoire) consiste à faire une série de modifications aléatoires à partir d'une solution initiale et à restituer la meilleure solution rencontrée. La deuxième méthode (First Improvement Hill Climbing) n'accepte une modification que si elle améliore la solution ; elle s'arrête entre autres quand aucune solution voisine n'est améliorante. La méthode suivante (Best Improvement Hill Climbing) choisit à chaque itération la meilleure solution améliorante parmi un ensemble de solutions voisines. Elle aussi s'arrête quand aucune voisine améliorante n'est trouvée. La dernière méthode est un recuit simulé, dont l'avantage est de ne pas être bloqué sur le premier maximum local trouvé, à l'inverse des méthodes précédentes.

Ces deux approches (synthèse et amélioration d'un aménagement) ont été implantées dans deux logiciels opérationnels, dont le fonctionnement a été illustré dans les deux chapitres correspondants.

Travaux futurs

Les perspectives de ce travail sont nombreuses. Nous les présentons en deux catégories : des perspectives d'ordre général et des perspectives par approche.

Dans la première catégorie, nous envisageons de développer une base de connaissances qui permettra d'effectuer automatiquement la traduction, actuellement manuelle, des contraintes de haut niveau (exprimées par l'utilisateur du logiciel) en des ensembles de critères de bas niveau.

Nous comptons, aussi, adapter et tester les deux approches développées sur des exemples de plus grandes tailles, tels que les ERP (Etablissements Recevant du Public). Pour cela, nous avons besoin de satisfaire d'autres exigences, par exemple l'évacuation d'un maximum de personnes en un minimum de temps, en cas d'incendie, ce qui pourrait correspondre à un problème de flot maximum à coût minimum.

Nous comptons améliorer la méthode de la synthèse d'un aménagement comme suit :

- concernant le module de placement et de dimensionnement des pièces, deux pistes se présentent :
 - pour améliorer les performances de ce module, et donc permettre une convergence rapide vers une solution optimale, il sera nécessaire de mener une étude polyédrale qui débouchera sur l'ajout de coupes spécifiques à notre problème ;
 - nous envisageons aussi, sur le moyen terme, une extension du modèle mathématique sous forme quadratique, cette forme se prêtant naturellement à la modélisation des contraintes sur les aires et sur la consommation énergétique du logement, la quantité de lumière traversant une pièce étant estimée à l'aide d'une fonction quadratique (voir [59]).
- concernant le module de placement des portes, on peut envisager, en plus d'une étude polyédrale, d'adapter le modèle mathématique pour offrir la possibilité de placer plusieurs portes par pièce ;

Nous comptons enrichir notre méthode d'amélioration d'un aménagement comme suit :

- nous envisageons d'implémenter des modificateurs de plus haut niveau pour que les modifications soient plus profondes, et en particulier pour diversifier les solutions considérées ;
- les modifications sont faites en translatant des sommets ce qui donne des angles souvent non droits. Une piste que nous envisageons d'explorer est la translation de murs entiers ou de portions de murs, pour conserver des formes plus traditionnelles ;
- l'hypothèse de base dans la structure de données qui décrit un logement est que toute pièce et tout couloir est entouré d'une unique boucle de murs. En termes mathématiques, cela revient à poser que chaque pièce/couloir est topologiquement équivalent à (ou continuement déformable en) un disque. Ce n'est malheureusement pas le cas dans le plan ci-dessous, où le couloir est une couronne : il faut deux contours pour limiter ce couloir, un contour extérieur, comme d'habitude, et un contour intérieur. Ce type de plan ne pourra donc être proposé comme un résultat ni être un point de départ de la méthode d'optimisation proposée dans ce manuscrit.

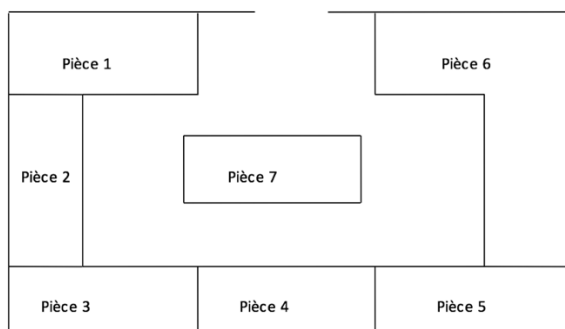


Figure 144. Cas non géré, où le couloir est une couronne

La structure de données devra donc supporter que chaque pièce ait un ou plusieurs contours intérieurs ;

- les quatre heuristiques développées ont permis de valider le jeu de modificateurs. Il serait intéressant, maintenant, d'intégrer ces modificateurs dans d'autres méthodes de résolution, plus sophistiquées, en particulier les algorithmes génétiques car le chapitre bibliographique a montré qu'ils ont souvent été utilisés avec succès. Nous nous attendons à rencontrer des difficultés particulières au moment de décrire l'opérateur de croisement.
- l'approche développée est une approche aléatoire, notamment le choix des opérateurs et leurs arguments. Une approche plus intelligente, dans laquelle les modifications sont guidées par des préconditions, est une piste qui nous semble prometteuse ;
- l'ensemble actuel des critères d'évaluation est conséquent et a autorisé la prise en compte d'une panoplie de spécifications plus étendue que dans les autres études publiées. Si l'on souhaite aller jusqu'à une industrialisation, il devra pourtant être étendu. Par exemple, autoriser des formes non rectangulaires est un apport réel mais il permet que des murs consécutifs soient presque alignés. Il faudra donc prévoir une correction automatique de ces quasi-alignements.

1 Segment parallèle à un autre

Cette opération est utilisée pour calculer un segment parallèle à un élément de trajectoire, à une distance r . Parmi les deux segments parallèles possibles, il faut veiller à construire celui qui est du côté d'un mur $[A, B]$. Appelons $[C, D]$ le segment d'origine et $[C', D']$ le segment parallèle à construire. Pour obtenir C' et D' , il suffit d'ajouter à C et D un vecteur perpendiculaire à $[C, D]$, de norme r et orienté vers $[A, B]$. Ce vecteur V est obtenu par les formules suivantes :

$$U = CD \wedge (CD \wedge Dir), \text{ où } \wedge \text{ désigne le produit vectoriel}$$

$$V = r * U / ||U||$$

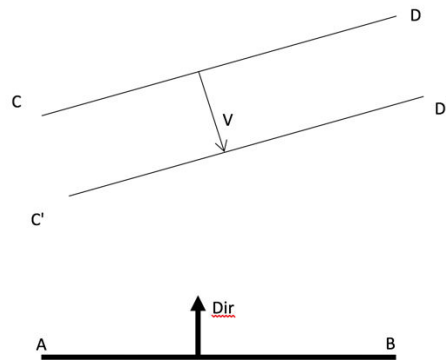


Figure 145. Construction d'un segment parallèle et du bon côté

Ces formules se justifient comme suit :

- comme le produit vectoriel restitue un vecteur perpendiculaire à ses arguments, $CD \wedge Dir$ donne un vecteur, appelé W , perpendiculaire à la figure, de la forme $(0, 0, z)$. Si C est "à gauche" de D , ce vecteur W est orienté vers le haut ($z > 0$), sinon vers le bas ($z < 0$) ;
- le produit vectoriel U de CD avec W , étant perpendiculaire à W , se trouve dans le plan. Etant perpendiculaire à CD , il est, à sa norme près, égal au vecteur cherché ;
- U est ensuite divisé par sa norme et multiplié par r pour ajuster sa longueur comme souhaité.

Ces formules ont une exception : lorsque $[C, D]$ est orthogonal à $[A, B]$, CD et Dir sont colinéaires donc leur produit vectoriel est le vecteur nul. Dans ce cas, il n'y a pas lieu de calculer un segment parallèle à $[A, B]$ car il serait à même distance de $[A, B]$ que $[C, D]$.

2 Intersection des parallèles à deux droites sécantes

Soit $U1(x_1, y_1), U2(x_2, y_2)$ deux vecteurs non colinéaires [62];

soit $n1(x_{n1}, y_{n1}), n2(x_{n2}, y_{n2})$ des vecteurs normés respectivement orthogonaux à $U1$ et $U2$;

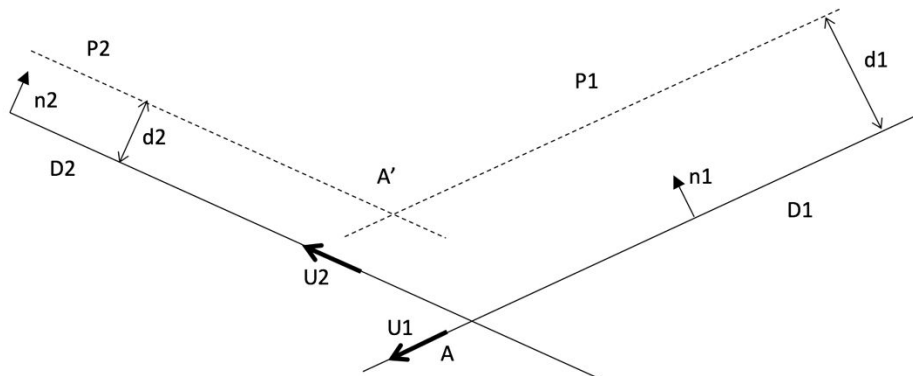
soit A un point.

Appelons $D1$ et $D2$ les droites passant par A et orientées par $U1$ et $U2$.

$P1$ est la parallèle à $D1$, à une distance $d1$, du côté indiqué par $n1$ (orthogonal à $U1$).

$P2$ est la parallèle à $D2$, à une distance $d2$, du côté indiqué par $n2$ (orthogonal à $U2$).

Quelles sont les coordonnées du point d'intersection A' de $P1$ et $P2$?



Un point de $P1$ est $A + d_1 \cdot n_1$; par ailleurs, $U1$ est aussi un vecteur directeur de $P1$ donc les équations paramétriques de $P1$ sont :

$$P1(t) = A + d_1 \cdot n_1 + t \cdot U_1$$

De même, celles de $P2$ sont :

$$P2(r) = A + d_2 \cdot n_2 + r \cdot U_2$$

Le point d'intersection a les paramètres t_0 et r_0 tels que :

$$A + d_1 \cdot n_1 + t_0 \cdot U_1 = A + d_2 \cdot n_2 + r_0 \cdot U_2$$

En résolvant ce système de deux équations (l'une en x, l'autre en y) à deux inconnues (t_0 , r_0), on trouve :

$$t_0 = [x_2 \cdot (d_2 \cdot y_{n2} - d_1 \cdot y_{n1}) - y_2 \cdot (d_2 \cdot x_{n2} - d_1 \cdot x_{n1})] / (y_1 \cdot x_2 - x_1 \cdot y_2)$$

Il s'en suit que $A' = P1(t_0) = A + d_1 \cdot n_1 + t_0 \cdot U1$

Pour information, et bien que ce soit inutile dans ce cas précis :

$$r_0 = [x_1 \cdot (d_2 \cdot y_{n2} - d_1 \cdot y_{n1}) - y_1 \cdot (d_2 \cdot x_{n2} - d_1 \cdot x_{n1})] / (y_1 \cdot x_2 - x_1 \cdot y_2)$$

3 Calcul d'un angle capable

Un angle capable est défini par un point A et un segment $[P_1, P_2]$. Il s'agit de l'angle orienté α formé par les segments $[A, P_1]$ et $[A, P_2]$ (Figure 146).

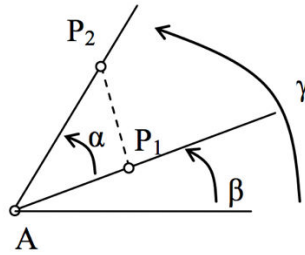


Figure 146. Angle Capable

On le calcule efficacement en utilisant une et une seule fois la fonction arc tangente et les formules qui expriment le cosinus et le sinus d'une différence de deux angles.

Avant toute chose, notons que si (x_A, y_A) , (x_1, y_1) et (x_2, y_2) sont les coordonnées de A , P_1 et P_2 , on a :

$$\cos(\beta) = (x_1 - x_A) / AP_1$$

$$\sin(\beta) = (y_1 - y_A) / AP_1$$

$$\cos(\gamma) = (x_2 - x_A) / AP_2$$

$$\sin(\gamma) = (y_2 - y_A) / AP_2$$

On remarque que $\alpha = \gamma - \beta$

et comme :

$$\cos(a - b) = \cos(a) \cdot \cos(b) + \sin(a) \cdot \sin(b)$$

$$\sin(a - b) = \sin(a) \cdot \cos(b) - \sin(b) \cdot \cos(a)$$

on obtient :

$$\cos(\alpha) = \cos(\gamma - \beta) = \cos(\gamma) \cdot \cos(\beta) + \sin(\gamma) \cdot \sin(\beta) = \frac{(x_1 - x_A) \cdot (x_2 - x_A) + (y_2 - y_A) \cdot (y_1 - y_A)}{AP_1 \cdot AP_2}$$

$$\sin(\alpha) = \sin(\gamma - \beta) = \sin(\gamma) \cdot \cos(\beta) - \sin(\beta) \cdot \cos(\gamma) = \frac{(y_2 - y_A) \cdot (x_1 - x_A) - (y_1 - y_A) \cdot (x_2 - x_A)}{AP_1 \cdot AP_2}$$

La fonction arc tangente (atan2 en langage C) calcule un angle à partir de son sinus et de son cosinus ou à partir d'un même multiple du sinus et du cosinus, par exemple :

$$\text{atan}(s, c) = \text{atan}(10 \cdot s, 10 \cdot c)$$

Nous multiplions donc les sinus et cosinus calculés ci-dessus par $AP_1 \cdot AP_2$ et nous obtenons l'angle capable α avec :

$$\alpha = \text{atan2}((y_2 - y_A) \cdot (x_1 - x_A) - (y_1 - y_A) \cdot (x_2 - x_A), (x_1 - x_A) \cdot (x_2 - x_A) + (y_2 - y_A) \cdot (y_1 - y_A))$$

Autrement dit, l'angle capable est obtenu au prix d'un *unique* appel à une fonction trigonométrique.

Notons que si A est aligné avec P_1 et P_2 , alors il existe un réel k tel que :

$$A = (1 - k).P_1 + k.P_2$$

Dans ce cas, en faisant abstraction du dénominateur que l'on sait être inutile, on a :

$$\cos(\alpha) = -k.(1 - k).[(x_2 - x_1)^2 + (y_2 - y_1)^2]$$

$$\sin(\alpha) = 0$$

Autrement dit

- si A est superposé à P_1 ou P_2 ($k = 0$ ou 1), le sinus et le cosinus sont nuls tous les deux, ce qui interdit le calcul de l'angle ;
- si A est strictement entre P_1 et P_2 ($0 < k < 1$), $\cos(\alpha)$ est négatif, $\sin(\alpha)$ est nul donc l'angle calculé est $+\pi$, ce qui est naturel ;
- si A est aligné avec P_1 et P_2 mais sans être entre P_1 et P_2 ($k > 1$ ou $k < 0$), $\cos(\alpha)$ est positif, $\sin(\alpha)$ est nul donc l'angle calculé est 0 , ce qui est naturel aussi.

Bibliographie

- [1] ANAH (Agence nationale de l'habitat, "ADAPTATION DES LOGEMENTS EXISTANTS AU HANDICAP ET AU VIEILLISSEMENT," 2005.
- [2] M Arrafzadeh and D, T Lee, *Algorithmic Aspects of VLSI Layout.*: World Scientific, 1993, vol. 2.
- [3] S Aupetit, A Puret, P, Monmarché, N Gaucher, and M Slimane, "Visual tools to select a layout for an adapted living area," in *Assistive Technology from Adapted Equipment to Inclusive Environments, AAATE2009*, Florence, Italy, 2009, pp. 345-349.
- [4] C A Bana e Costa and J-C Vansnick, "Applications of the MACBETH Approach in the Framework of an Additive Aggregation Model," *Journal of multi-criteria decision analysis*, vol. 6, pp. 107-114, 1997.
- [5] S Ben Mena, "Introduction aux méthodes multicritères d'aide à la décision," *Biotechnol. Argon.Soc.Environ*, vol. 4, no. 2, pp. 83-93, 2000.
- [BO] M P Bendsøe and O Sigmund, *Topology Optimization: Theory, Methods, and Applications*, 2nd ed. New York: Springer-Verlag Berlin Heidelberg, 2004.
- [7] M Bierlaire, *Introduction à l'optimisation différentiable.*: Presses Polytechniques Universitaires Romandes Editeur, 2006.
- [8] A Billionnet, *Optimisation Discrète : De la modélisation à la résolution par des logiciels de programmation mathématique.* Paris: DUNOD, 2007.
- [9] T Boukherroub, A Guinet, and J Fondervelle, "Méthode d'aide à la décision multicritères pour l'internalisation/ externalisation "DURABLE"," in *Performance, interopérabilité et sécurité pour le développement durable*, Bordeaux, France, Juin 2012.

- [10] Y Bouzoubaa, C Minich, A Nagih, and A Pruski, "EVALUATING THE ADAPTATION OF A DWELLING TO A DISABLED PERSON: A GRAPH-BASED APPROACH," in *45th International Conference on Computers & Industrial Engineering 2015 (CIE45)*, vol. 2, Metz, 2015, pp. 728-735.
- [11] Y Bouzoubaa, C Minich, A Nagih, and A Pruski, "Synthesis of a dwelling layout as a packing problem using objects of varying sizes," in *Control, Decision and Information Technologies (CoDIT), 2016 International Conference on*, Saint Julian's, 2016, pp. 758-762.
- [12] R Caillet, "Analyse multicritère: Etude et comparaison des méthodes existantes en vue d'une application en analyse de cycle de vie," Montréal, 2003.
- [13] CCAH. Les différents types de handicap. [Online]. <http://www.ccah.fr/former/formations/les-differents-types-de-handicap>
- [14] V Cerny, "Thermodynamical approach to the traveling salesman problem : an efficient simulation algorithm ," *Journal of Optimization Theory and Applications*, vol. 45(1), pp. 41–51, 1985.
- [15] Certu, "Handicaps mentaux, cognitifs et psychiques. Quelles pistes pour améliorer l'accessibilité?," Lyon, Guide ISSN : 0247-1159, 2013.
- [16] A Chamaret, V Reichel, and M O'Connor, "Aménagement durable des territoires : quel apport d'un outil multimédia d'aide à la délibération ?," vol. 22, no. 3/4, pp. 287-302, 2008.
- [17] F Cherqui, "Thèse : Methodologie d'évaluation d'un projet d'aménagement durable d'un quartier," La Rochelle, 2005.
- [18] T H Cormen, C E Leiserson, R L Rivest, and C Stein, *Algorithmique*, 3rd ed. Paris: DUNOD, 2010.
- [22] G Dòsa and J Sgall, "First fit bin packing: a tight analysis," *Proc. of STACS 2013*, vol. 20, pp. 538-549, 2014.

- [19] J.C Damski and J.S Gero, "an Evolutionary Approach to Generating Constraint-Based Space Layout Topologies," pp. 149- 162, 1997.
- [20] M de Berg, M van Kreveld, M Overmars, and O Schwarzkopf, *Visibility Graphs. In Computational Geometry*. Berlin Heidelberg, Germany: Springer , 1997.
- [21] DGALN/DHUP. (2010) Site web accessibilité-batiment. [Online]. <http://www.accessibilite-batiment.fr/>
- [23] A. Drira, H. Pierreval, and S Hajri-Gabouj, "Facility layout problems: A survey," *Annual Reviews in Control*, vol. 31(2), pp. 255-267, 2007.
- [24] H Dyckhoff, "A typology of cutting and packing problems," *European Journal of Operational Research*, pp. 145-159, Jan. 1990.
- [25] J Egeblad, "Placement Techniques for VLSI Layout Using Sequence-Pair Legalization," Department of Computer Science, University of Copenhagen, Copenhagen, Mémoire de D.E.A 2013.
- [26] S Elloumi, "Contributions à l'Optimisation Combinatoire," CEDRIC Laboratory, Paris, Post-Doctoral Degree (Phd students supervisor) Habilitation à Diriger des Recherches de l'Université de Paris-Nord 2002.
- [27] European Commission, "Access City Award 2014," European Commission, Luxembourg, ISBN 978-92-79-36165-4, 2014.
- [28] FNATH, La Poste, DGCS, "Le GUIDE du REPRESENTANT des personnes HANDICAPÉES dans les commissions d'ACCESSIBILITÉ," Fédération Nationale des Accidentés du travail et des Handicapés F.N.A.T.H, Paris, Guide 2011.
- [29] P Fouilhoux, "Graphes K-partis et conception de circuits VLSI," Blaise Pascal – Clermont Ferrand II University, Clermont Ferrand, Thèse de doctorat 2004.

- [30] P Fouilhoux and A R Mahjoub, "Solving VLSI design and DNA sequencing problems using bipartization of graphs," *Computational Optimization and Applications*, vol. 2, pp. 749-781, 2012.
- [31] P Galle, "An algorithm for exhaustive generation of building floor plans," vol. 24, no. 12, pp. 813-825, 1981.
- [32] Y Gardan, *Traitements de base dans le plan.*, 2012, livre en version électronique (guide MICADO) <http://www.simulation-numerique.fr/>. [Online]. <http://www.simulation-numerique.fr/>
- [33] F Glover, "Improved linear integer programming formulations of nonlinear integer problems," *Management Science*, pp. 455-460, 1975.
- [34] M Grötschel, A Martin, and R Weismantel, "The Steiner tree packing problem in VLSI design," *Mathematical Programming*, vol. 78(2), pp. 265-281.
- [35] Handiaccess. Handicass - isManager - loi du février 2005. [Online]. http://www.handiaccess.org/site/element_153.php
- [36] ANLH (Association Nationale pour le Logement des personnes Handicapées). Accessibilité des voiries - les dimensions anthropométriques. [Online]. <http://www.anlh.be/accessvoirie/acc07.htm>
- [37] M Hifi, I Kacem, S Negre, and L Wu, "A linear programming approach for the three-dimensional bin-packing problem," *Electronic Notes in Discrete Mathematics*, pp. 993-1000, 2010.
- [38] J H Hughes et al., *Computer Graphics: Principles and Practice*, 3rd ed. New Jersey: Addison-Wesley, 2014.
- [39] INPES. (2012, Novembre) www.santepubliquefrance.fr. [Online]. <http://inpes.santepubliquefrance.fr/sante-handicap/france/defintion.asp>

- [40] G Jacquenot, "Méthode générique pour l'optimisation d'agencement géométrique et fonctionnel," Nantes, thèse de doctorat 2010.
- [41] K., Mahdian, M., Salavatipour, M. R. Jain, "Packing steiner trees," *Society for Industrial and Applied Mathematics.*, pp. 266-274, Jan. 2003.
- [42] Z H Jin, T Ito, and K Ohno, "The three-dimensional bin packing problem and its practical algorithm," *SME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, pp. 60-66, 2003.
- [43] S Kirkpatrick, C D Gelatt, and M P Vecchi, "Optimization by simulated Annealing," *Science, New Series*, vol. 220, pp. 671-680, Mai 1983.
- [44] G, J Kott, "A method for mixed variable constrained nonlinear optimization based on new function bounding technique," Rensselaer Polytechnic Institute, thèse 1998.
- [45] G,J Kott and G Gabriele, "A tunnel based method for mixed discrete constrained nonlinear optimization," in *ASME Design Automation Conference DETC98/DAC 5592*, 1998.
- [46] H Kuhn,. Basel: Springer Basel, 2014, pp. 393-414.
- [47] La ville de Nantes. Handicap : renforcer l'égal accès à la ville et à la citoyenneté. [Online]. www.nantes.fr/handicap
- [48] T Lai and S.M Leinwand, "Algorithms for floorplan design via rectangular dualization," *IEEE Transactions on Computer-Aided Design*, pp. 1278-1289, 1988.
- [49] J Leloup, "Le projet HM2PH, habitat modulaire et mobile pour personnes handicapées : spécification d'un espace de vie adapté pour personne en déficit d'autonomie," Tours, Thèse de doctorat 2004.
- [50] J Leloup and P Gaucher, "HMPH — “Habitat Mobile pour Personnes Handicapées” — Software for the Design of an Adapted Living Area," in *Computers Helping People with Special Needs: 8th International Conference, ICCHP, Linz, Austria, 2002*, pp. 659-661.

- [51] J Leloup, P Gaucher, and S Pellieux, "The HM2PH Project: A Database to Help Prescription of Assistive Technologies," in *Computers Helping People with Special Needs: 9th International Conference, ICCHP 2004*, Paris, France, 2004, pp. 930-936.
- [52] T Lengauer and M Lûgering, "Integer Program Formulations of Global Routing and Placement Problems," *Reihe Informatik*, vol. 95, 1991.
- [53] A. Martinez-Sykora, R. Alvarez-Valdés, J. A., Ruiz, R. Bennell, and J. M Tamarit, "Matheuristics for the irregular bin packing problem with free rotations," *European Journal of Operational Research*, vol. 258(2), pp. 440-455, 2017.
- [54] MDPH, "Guide d'évaluation des besoins de compensation de la personne handicapée," 2008.
- [55] B Medjdoub, "Methode de conception fonctionnelle en architecture: une approche CAO basee sur les contraintes: ARCHiPLAN," Ecole Centrale Paris, Paris, thèse de doctorat 1996.
- [56] B Medjdoub and B Yannou, "Separating topology and geometry in space planning," *Computer-aided design*, vol. 32(1), pp. 39-61, 2000.
- [57] P. Merrell, E Schkufza, and V Koltun, "Computer-generated residential building layouts," vol. 29 (6), p. 181, 2010.
- [58] N Metropolis and al, "Equations of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087-1092, 1953.
- [59] J Michalek, R Choudhary, and Papalambros, "Architectural layout design optimization.," *Engineering optimization*, vol. 34, no. 5, pp. 461-484, 2002.
- [60] Ministère de l'Emploi de la Cohésion Sociale et du Logement, , Août 2006.
- [61] OMS, *Classification internationale du fonctionnement, du handicap et de la santé*. Genève, 2001.

- [62] J O'Rourke, *Computational Geometry in C*, 2nd ed. Cambridge, UK: Cambridge University Press, 1998.
- [63] C.H Papadimitriou, "The complexity of combinatorial optimization problems," Princeton University, 1976.
- [64] C.H Papadimitriou and K Steiglitz, *Combinatorial optimization : algorithms and complexity*. New Jersey: Prentice-Hall, Inc, 1982.
- [65] D Pedoe, *Circles: A Mathematical View*, 2nd ed. Minnesota: the mathematical association of America (Incorporated), 1997.
- [66] F. P Preparata and M Shamos, *Computational geometry: an introduction*. New York, USA: Springer Science & Business Media, 1985.
- [67] R C Prim, "Shortest connection networks and some generalizations," *Bell Labs Technical Journal*, vol. 36(6), pp. 1389-1401, 1957.
- [68] A Puret, "Projet HM2PH, Génération automatique de plans et visite virtuelle d'habitats adaptés pour personnes handicapées," Tours, Thèse de doctorat 2007.
- [69] A, Aupetit, S, Gaucher, P, Monmarché, N, & Slimane, M Puret, "PROJET HM2PH: TRANSFORMATION INTERACTIVE DE PLANS D'HABITATION," in *Congrès de la Société Française des Technologies pour l'Autonomie et de Gérontechnologie*, 2009.
- [70] A Puret, P Gaucher, N Monmarché, and N Slimane, "Towards a Virtual Environment for the Design, Validation and Exploitation of an Adapted Living Area," in *European Conference for the Advancement of Assistive Technology, AAATE 2007*, vol. 20, San Sebastian, Spain, 2007, pp. 101 - 105.
- [71] A Requicha, *Geometric modeling: A first course*. Los Angeles, USA: University of South California., 1996.
- [72] B Roy and D Bouyssou, *Aide Multicritère à la Décision : Méthodes et Cas*, 1993rd ed. Paris, 1993.

- [73] B Roy and J Figuera, "Determining the weights of criteria in the ELECTRE type methods with a revised Simos' procedure," *European Journal of Operational Research*, no. 139, pp. 317-326, 2002.
- [74] T L Saaty, *Décider face à la complexité ; une approche analytique multicritère d'aide à la décision*, Collection Université-Entreprise dirigée par Alain Chevalier et Georges Hirsch ed. paris: Entreprise moderne d'édition, 1984.
- [75] A Schwarz, D, M Berry, and E Shaviv, "On the use of the automated building design system," *Computer-Aided Design*, vol. 26, no. 10, pp. 747-762, 1994.
- [76] A Schwarz, D, M Berry, and E Shaviv, "Representing and solving the automated building design problem," *Computer-aided design*, vol. 26, no. 9, pp. 689-698, 1994.
- [77] E Shaviv and D Gali, "A model for space allocation in complex buildings," *Build International*, pp. 493-518, 1974.
- [78] P Siarry, *Métaheuristiques*. Paris: Eyrolles, 2014.
- [79] J Simos, "L'évaluation environnementale : Un processus cognitif négocié. Thèse de doctorat," Lausanne, 1990.
- [80] B Teissier, "Volumes des corps convexes, géométrie et algèbre," Institut de mathématiques de Jussieu,.
- [81] T A.M Toffolo and E Eline, "A two-dimensional heuristic decomposition approach to a three-dimensional multiple container loading problem," *European Journal of Operational Research*, vol. 257, pp. 526-538, Mar. 2017.
- [82] E Tomor and G Franck, "Genetic algorithms in support of creative architectural design," in *eCAADe17*, Liverpool, 1999, pp. 645-651.
- [83] S Warshall, "A Theorem on Boolean Matrices," *Journal of the ACM (JACM)*, vol. 9(1), pp. 11-12, 1962.

- [84] G Wäscher, H Haußner , and H Schumann, "An improved typology of cutting and packing problems," *European Journal of Operational Research*, pp. 1109– 1130, 2007.
- [85] L WU, "Contribution à la programmation linéaire en nombres entiers : problèmes de placement/chargement et knapsack," Université de Picardie Jules VErne, Thèse de doctorat.
- [86] www.wikipedia.org. [Online]. [en.wikipedia.org/wiki/Design_for_All_\(in_ICT\)](http://en.wikipedia.org/wiki/Design_for_All_(in_ICT))
- [87] B Yannou and F Limayem, "Les méthodes de comparaison par paires - intérêt fondamental, Méthodes pratiques, Avancées scientifiques, Logiciel".

Liste des illustrations

Illustration 1.	Les niveaux des contrastes visuels en fonction des matériaux utilisés [21]	11
Illustration 2.	Exemple d'un point d'appui assis-debout [21].....	11
Illustration 3.	Les dimensions à respecter d'une banque d'accueil [21].....	14
Illustration 4.	Les angles à respecter pour un système adapté d'ouverture des bâtiments [21] 15	
Illustration 5.	La largeur minimale d'une main courante [21].....	15
Illustration 6.	Le diamètre minimal à respecter dans une salle de bain [21].....	16

Liste des algorithmes

Algorithme 1	Troncature d'un obstacle	66
Algorithme 2	Fonction d'éloignement maximal d'un mur	86
Algorithme 3	Procédure de suppression des couloirs	87
Algorithme 4	Procédure de construction du graphe d'adjacence	132
Algorithme 5	Algorithme de recherche locale	159
Algorithme 6	Algorithme de recherche locale aléatoire	162
Algorithme 7	Algorithme FIHC.....	163
Algorithme 8	Algorithme de recuit simulé	169

Liste des tableaux

Tableau 1	Liste des opérateurs de bas niveau	35
Tableau 2	Exemples de modélisations des contraintes de haut niveau	35
Tableau 3	Quelques rapports d’aspect pour des rectangles de même aire	60
Tableau 4	Caractéristiques du logement	91
Tableau 5	Les dimensions minimales et maximales et des zones de placement des pièces 91	
Tableau 6	Les paramètres des entrées	92
Tableau 7	Résultats de la phase 1	93
Tableau 8	Résultats de la phase 2	94
Tableau 9	Résultats des trois phases	96
Tableau 10	Résultats de la phase 1	97
Tableau 11	Résultats de la phase 2	97
Tableau 12	Résultats des trois phases	99
Tableau 13	Résultats de la phase 1	100
Tableau 14	Résultats de la phase 2	101
Tableau 15	Résultats des trois phases	103
Tableau 16	Tableau récapitulatif.....	103
Tableau 17	Intervalles des surfaces par type de logement.....	104
Tableau 18	Tableau des surfaces.....	105
Tableau 19	Résultats des tests.....	107
Tableau 20	Résultats des tests.....	109
Tableau 21	Tableau récapitulatif de l’ensemble des tests	111

Liste des figures

Figure 1.	Classification internationale du handicap adoptée par l’OMS en 1980.....	5
Figure 2.	Classification Internationale du Fonctionnement, du Handicap et de la Santé adopté par l’OMS en 2001	6
Figure 3.	Aménagement durable et aspects concernés [17]	9
Figure 4.	Approche globale à la confluence des trois préoccupations, dites « les trois piliers du développement durable »	10
Figure 5.	Typologie des problèmes de découpe et de conditionnement proposé par Dyckhoff [24].....	18
Figure 6.	Classification des problèmes de découpe et de conditionnement proposée par Wäscher [84,40].....	19
Figure 7.	Constituants principaux d’un problème d’agencement [40].....	20
Figure 8.	Exemples de contraintes géométriques	21
Figure 9.	Exemples de contraintes fonctionnelles.....	21
Figure 10.	Exemple d’un plan généré	24
Figure 11.	Processus de génération de plans automatique	25
Figure 12.	Processus de génération de plans interactif	26
Figure 13.	Exemple d’une topologie [59]	28
Figure 14.	Deux interprétations géométriques d’une grille topologique.....	28
Figure 15.	Exemple d’un plan généré avec l’outil automatique.....	29
Figure 16.	Sens de parcours des murs	39
Figure 17.	Cas de figures de la position relative d’un mur par rapport à une pièce.....	40
Figure 18.	En trait plein, les murs filaires ; les doubles flèches donnent la position de la matière sur les murs filaires ; en pointillés, le pourtour de la pièce.....	40
Figure 19.	Variation du produit scalaire de deux vecteurs.....	41
Figure 20.	Exemples de vecteurs de produits vectoriels	42

Figure 21.	Pourtour d'une pièce	43
Figure 22.	La différence des dimensions entre le polygone d'une pièce et le polygone de son pourtour	43
Figure 23.	Calcul du pourtour	44
Figure 24.	Calcul du point en vis à vis d'un sommet convexe ou peu concave.....	45
Figure 25.	Les trois valeurs possibles pour la distance entre le mur filaire et le mur du pourtour	45
Figure 26.	Calcul des deux points en vis à vis d'un sommet très concave	46
Figure 27.	Exemple d'une direction souhaitée si l'angle est à 0°	47
Figure 28.	Quelques points à l'intérieur d'un polygone.....	48
Figure 29.	Exemple de plusieurs points à l'intérieur d'un polygone	49
Figure 30.	Exemples de Relations d'inclusions sans intersections franches.....	50
Figure 31.	Exemple d'angle capable	51
Figure 32.	Vérification de l'inclusion d'un point dans un polygone à l'aide de la méthode dite des angles capables	51
Figure 33.	Exemple d'un angle d'un sommet à mesurer.....	52
Figure 34.	Offset à un polygone.....	53
Figure 35.	Comparaison entre l'offset approché (à gauche, en pointillés) et l'offset théorique (à droite, en pointillés).....	54
Figure 36.	Deux cas de croisement d'un offset à une distance R	54
Figure 37.	Exemple d'un polygone inclus dans un autre	56
Figure 38.	Cas particulier qui exige la vérification de l'appartenance d'une pièce dans une autre	56
Figure 39.	Une permutation qui provoque l'inclusion d'une pièce dans un couloir	57
Figure 40.	Les trois cas possible quand deux polygones n'ont pas d'intersection franche.	57
Figure 41.	Deux cas dans lesquels l'incluant de A1 et A2 est différent.....	58

Figure 42.	Un aménagement adapté aux exigences de l'occupant mais avec une cuisine en étoile	59
Figure 43.	Exemple de détection de deux couloirs	61
Figure 44.	Choix du prochain mur dans la reconstruction d'un couloir	62
Figure 45.	Choix du prochain mur dans la reconstruction d'un couloir	62
Figure 46.	Distance d'un segment à un obstacle	63
Figure 47.	Les vecteurs v ayant un produit scalaire positif avec u	64
Figure 48.	Un mur et des obstacles	64
Figure 49.	Les obstacles réduits à leur partie utile	65
Figure 50.	Aménagement généré automatiquement pour un habitat de dimensions 10x15 mètres	67
Figure 51.	Configuration non souhaitable : pièces scindant le logement (à gauche), une des configurations souhaitable (à droite)	74
Figure 52.	Pas de discrétisation et formes potentielles pour une pièce	76
Figure 53.	MinDistPorteMur et sommets du graphe.....	78
Figure 54.	Le graphe utilisé pour les calculs des itinéraires	80
Figure 55.	Exemple d'une sélection de portes sans considérer la troisième contrainte	82
Figure 56.	(a) couloirs inutiles en gris, (b) après la suppression des couloirs inutiles.....	83
Figure 57.	Les quatre obstacles pour HM4	84
Figure 58.	Mur artificiel associé à une porte.....	85
Figure 59.	Une redistribution possible des couloirs inutiles	87
Figure 60.	Un couloir isolé (zone rayée) qui apparaît après l'élargissement des pièces R1..R4	88
Figure 61.	Une poignée de porte, inaccessible parce que MinDistPorteMur est trop petit	89
Figure 62.	Impact des contraintes d'optimisation	90
Figure 63.	Solution de la première étape.....	93

Figure 64.	(a) Plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues.....	94
Figure 65.	Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera	95
Figure 66.	Solution finale.....	95
Figure 67.	Solution de la première étape.....	96
Figure 68.	(a) Plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues.....	97
Figure 69.	Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera	98
Figure 70.	Solution finale.....	98
Figure 71.	Solution de la première étape.....	100
Figure 72.	(a) plan avec toutes les portes candidates et le graphe de mobilité ; (b) solution de la deuxième étape : plan avec les portes retenues.....	101
Figure 73.	Solution finale ; les traits en rouge sont les itinéraires que l'occupant empruntera	102
Figure 74.	Solution finale.....	102
Figure 75.	L'évolution du temps de calcul en fonction du type du logement.....	107
Figure 76.	L'évolution du temps de calcul en fonction de la surface	108
Figure 77.	L'évolution du temps de calcul en fonction du type du logement.....	110
Figure 78.	L'évolution du temps de calcul en fonction de la surface	110
Figure 79.	Les trois étapes du processus d'optimisation en IAO	114
Figure 80.	Architecture logicielle de l'optimiseur	116
Figure 81.	(à gauche) Description en 2D filaire du logement, (à droite) description en 2D avec la prise en compte des épaisseurs et de la position de la matière des murs.....	118
Figure 82.	Une vue simplifiée de l'hôpital de Thionville (Moselle)	120
Figure 83.	Exemple de murs communs sur un plateau.....	120
Figure 84.	Exemple d'un graphe de mobilité pour une personne en fauteuil roulant	121

Figure 85.	En trait plein, les murs filaires ; les doubles flèches donnent la position de la matière sur les murs filaires ; en pointillés, le pourtour de la pièce.....	122
Figure 86.	L'encombrement pour une personne en fauteuil roulant [36]	123
Figure 87.	Un plus court chemin exact de A à B (à gauche) et le plus court chemin effectivement calculé, passant par plusieurs points caractéristiques (à droite)	124
Figure 88.	Franchissement de porte optimal pour aller d'A à B	124
Figure 89.	Franchissement de porte sous-optimal pour aller de A à B	125
Figure 90.	Exemple d'un point concave.....	126
Figure 91.	Le centre de masse G de la pièce en L est en dehors de la pièce.....	127
Figure 92.	Exemple d'un point de départ situé au centre d'une pièce avec deux portes ..	127
Figure 93.	Les points de départ d'une pièce à deux portes	128
Figure 94.	Offset d'une pièce.....	128
Figure 95.	(a) Le graphe des déplacements possibles entre points caractéristiques d'une pièce et (b) les ajouts à ce graphe pour le calcul de la porte la plus proche d'un point O_i de l'offset	129
Figure 96.	Des points de départ qui ne sont pas des sommets de l'offset (l'offset n'est pas représenté).....	130
Figure 97.	Graphe de mobilité d'un appartement de six pièces, pour une personne d'un rayon de 0.5 m le rayon de franchissement étant de 0.35 m.....	130
Figure 98.	Exemple d'un graphe d'adjacence	131
Figure 99.	Les pièces 4,5,6 ne sont pas accessibles depuis la porte d'entrée.....	136
Figure 100.	Les trois courbes linéaires par morceaux utilisées pour évaluer le critère avec un seuil	138
Figure 101.	Les courbes des fonctions d'évaluation simplifiées.....	138
Figure 102.	Les deux courbes d'évaluation avec un seuil raisonnable maximum	139
Figure 103.	La courbe de la fonction d'évaluation de la compacité.....	140
Figure 104.	La courbe de la fonction d'évaluation du coût de l'aménagement	141
Figure 105.	La courbe de la fonction d'évaluation de la forme raisonnable	142

Figure 106.	Une figure rectangulaire peut avoir plus de quatre murs	142
Figure 107.	La courbe d'évaluation pour la minimisation de la distance d'une pièce à une autre	143
Figure 108.	Exemple d'une distance marchée supérieure au diamètre du logement.....	143
Figure 109.	Exemple de l'ajout d'un sommet sur un mur partagé	146
Figure 110.	Exemple de la fusion de deux sommets	147
Figure 111.	Exemple de la fusion de deux sommets avec un segment en doublon.....	147
Figure 112.	Suppression d'une pièce avec la fusion de sommets.....	148
Figure 113.	Exemple de scission d'une pièce.....	148
Figure 114.	Recouvrement suite à la fusion de deux sommets.....	149
Figure 115.	Scission d'un sommet	149
Figure 116.	Exemple de déplacement d'un sommet.....	150
Figure 117.	Exemple de suppression d'un sommet	150
Figure 118.	Exemple de permutation de deux pièces	151
Figure 119.	Exemple de la fusion de deux couloirs.....	152
Figure 120.	Vision conceptuelle de l'espace des solutions	153
Figure 121.	Transformation d'une pièce rectangulaire en une pièce en L	155
Figure 122.	Exemple d'une série de déplacements de sommets pour faire coulisser une pièce	155
Figure 123.	Partages de sommets et de murs.....	156
Figure 124.	Annulation d'un collage par un sommet	156
Figure 125.	Annulation du partage d'un mur	156
Figure 126.	Changement profond de topologie	157
Figure 127.	Procédure de recherche locale.....	158
Figure 128.	Comparaisons de FIHC et BIHC.....	164
Figure 129.	Comparaison des techniques du recuit et de la trempe d'après P.Siarry [78]	165

Figure 130.	L'interface du logiciel	170
Figure 131.	Évaluation d'un logement avec les critères réglementaires et de bon sens..	171
Figure 132.	Un changement profond de topologie	172
Figure 133.	Différents logements obtenus.....	175
Figure 134.	Les trois courbes linéaires par morceaux utilisées pour évaluer le critère de l'aire d'une pièce.....	176
Figure 135.	Les trois courbes linéaires par morceaux utilisées pour évaluer le critère de l'aire d'une pièce, comprise entre 25 et 30 m ²	176
Figure 136.	Le logement produit par la méthode 2 (premier voisin améliorant)	178
Figure 137.	Le logement produit par la méthode 3 (meilleur voisin améliorant).....	179
Figure 138.	Un logement produit par la méthode 4 (le recuit simulé)	180
Figure 139.	Le logement produit par la méthode 2	181
Figure 140.	Le logement produit par la méthode 3	181
Figure 141.	Solution intermédiaire intéressante	182
Figure 142.	Le logement produit par la méthode 2	183
Figure 143.	Un plan grossier permettant d'utiliser l'outil de l'amélioration pour faire de la synthèse	184
Figure 144.	Cas non géré, où le couloir est une couronne.....	190
Figure 145.	Construction d'un segment parallèle et du bon côté.....	192
Figure 146.	Angle Capable.....	194