UNIVERSITÉ DU QUÉBEC À MONTRÉAL

ASSIGNATION EN TEMPS RÉEL DE CLIENTS À DES RESSOURCES GÉOGRAPHIQUEMENT DISTRIBUÉES

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN INFORMATIQUE

PAR

GUY FRANCOEUR

NOVEMBRE 2018

REMERCIEMENTS

Un mémoire est une épreuve qui ne peut être exécutée seule. Il est très important pour moi de souligner la gentillesse, l'éthique, la persévérance et l'excellent travail de mon directeur de recherche Professeur Éric Beaudry. Un gros merci. Ça vaut la peine. Mention spéciale à Professeur Mohammed Bouguessa pour son appui et ses opinions.

Je profite de l'occasion pour remercier les membres du laboratoire GDAC, tous mes professeurs, ainsi que toutes les personnes qui ont contribuées de près ou de loin à la réalisation de ce mémoire.

Sur le plan personnel, merci à mon cœur Dong-Ly, qui est auprès de moi tous les jours et m'encourage dans ce que j'accomplis. Sans oublier mes enfants, Quynh-Ly, Mai-Ly et Hoang Vinh qui m'ont soutenu et motivé à leurs façons sans relâche, tout au long de cette aventure. Sans eux, ce mémoire aurait été moins intéressant.

À tous mes amis qui ont subi mes absences, vous avez été pour moi une source de motivation immense.

Sans vous tous il n'y aurait jamais eu de mémoire, c'est un peu de vous tous qui est dans ce travail.

En plus des personnes mentionnées, voici certaines phrases qui ont joué un rôle dans la rédaction de ce mémoire. Un petit retour sur une loi ancienne qui est toujours actuelle et qui met la table sur mon sujet.

— «If you change queues, the one you have left will start to move faster than the one you are in now.»
— «Your queue always goes the slowest.»
— «Whatever queue you join, no matter how short it looks, it will always take the longest for you to get served.»

Murphy' Laws on reliability and queueing

Une grande source de motivation pour ce mémoire :

— «La folie, c'est de faire toujours la même chose et de s'attendre à un résultat différent.»
— «Tout problème a une solution, ou bien vous faites parti du problème.»
— «Un problème sans solution est un problème mal posé.»
— «C'est le devoir de chaque homme de rendre au monde au moins autant qu'il en a reçu.»
— «La connaissance s'acquiert par l'expérience, tout le reste n'est que de l'information.»

Albert Einstein

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

BSP Binary Space Partition

CMQ Collège des Médecins du Québec

EMR Electronic Medical Records

FIFO First In First Out

FMOQ Fédération des médecins omnipraticiens du Québec

GA Algorithme Génétique, (*Genetic Algorithm*)

ICIS Institut canadien d'information sur la santé

ISQ Institut de la Statistique Québec

KD K Dimensions pour Arbre-KD

KP Problème du sac à dos, (*Knapsack Problem*)

LIFO Last In First Out

LP Problème géographique, (*Location Problem*)

ML Apprentissage Machine, (*Machine Learning*)

MSSS Ministère de la Santé et des Services Sociaux

NP classe de problème de décision *Nondeterministic Polynomial time*

PIB Produit intérieur brut

RS Traitement aléatoire (*Random Service*)

RTA Région de Tri d'Acheminement (Postes Canada)

TSP Voyageur de commerce, (*Travelling Sales Person*)

RÉSUMÉ

Les problèmes liés à la gestion de files d'attente sont multiples et variés. Tout le monde est confronté à l'attente plusieurs fois dans sa vie. Ce travail de recherche se veut une initiative informatique afin de vérifier la faisabilité de réalisation d'un système d'assignation à des files d'attente distribuées dans un environnement incertain en temps continu.

Ce mémoire traite de la construction d'un simulateur consacré au problème de l'optimisation multicritères par une approche heuristique ainsi que l'exécution du plan pour des clients ambulatoires pour un réseau de cliniques médicales. Pour résoudre ce problème, nous avons d'abord étudié quelques systèmes de santé dans le monde, mais plus spécifiquement celui du Québec.

Découvrir la meilleure clinique pour des clients qui sont en situation d'urgence n'est pas un problème trivial. Proposer et planifier des rendez-vous qui optimisent de multiples objectifs tels que réduire le délai global d'attente et minimiser les temps morts cliniques ceci avec des variables incertaines telles que : le temps de traitement variable, le nombre de patients qui se désistent, le nombre de tâches quotidiennes totales, ainsi que le nombre de ressources disponibles, font de ce problème un véritable défi. Notre objectif est de déterminer (fabriquer et construire) un plan de tâches optimal ou proche optimal qui satisfait toutes les contraintes en minimisant les temps et les coûts, mais aussi en maximisant son efficacité et son efficience.

MOTS-CLÉS : Assignation, planification, file d'attente, multicritères, simulation, temps réel, heuristique, clinique, santé, ambulatoire

INTRODUCTION

Personne n'aime attendre dans une salle d'attente. Encore moins lorsqu'on est malade et souffrant. Pourtant, dans le réseau de santé du Québec, l'attente dans des salles, cliniques ou urgences est une réalité encore très présente. Lorsqu'un rendez-vous est fixé, l'heure de rendez-vous n'est généralement ni garantie ni respectée. Ainsi, si un client [1] a un rendez-vous à 11h, il doit se présenter un peu avant 11h afin d'être prêt au cas où il serait appelé dès son arrivée. S'il n'est pas présent au moment de son appel, il risque de perdre sa place. Cependant, il est fréquent qu'un client soit appelé seulement quelques dizaines de minutes, ou même quelques heures, après son heure de rendez-vous initialement prévue. Pour ces raisons, l'attente est fréquente et significative.

Dans les services de type sans rendez-vous, la situation est souvent pire. Au Québec, les gens font la queue dehors le matin avant l'ouverture des portes. Une situation inacceptable. Ce qui se traduit par un ordre de traitement de type premier arrivé, premier servi. Du moins, pour les clients ayant le même niveau de priorité, ce qui est assez fréquent dans les cliniques de santé. On peut présumer que si une personne malade savait précisément de quels maux elle souffre et qu'elle pouvait se guérir seule, elle n'aurait sûrement pas besoin de services médicaux. Avec cette supposition, nous pouvons affirmer que le service, la consultation sans rendez-vous, est d'une durée inconnue à l'avance et variable pour chaque client. Quoi qu'il en soit, il est difficile de fournir une estimation de l'heure précise de

1. Dans ce mémoire, un client désigne un usager du réseau de la santé (parfois appelé patient).

passage en se basant sur le rang occupé dans la file d'attente. Par conséquent, les clients doivent généralement attendre un temps considérable.

En plus d'être désagréable, l'attente engendre des coûts importants pour la société. Le coût peut être calculé sous la forme de perte du produit intérieur brut (PIB). Plus un patient attend dans une salle d'attente, plus il doit s'absenter du travail, ce qui a un impact sur sa productivité et possiblement sur la productivité de son employeur. Tous ces facteurs peuvent être évalués et plus le temps d'attente augmente, plus le coût est susceptible de grandir dû à l'état qui pourrait s'aggraver (Bates-Eamer et Ronson, 2009). Des employés en santé sont moins absents et plus productifs au travail. Ainsi, une enquête de Statistique Canada (2001) révélait que les coûts liés à l'absentéisme pour raison de maladie et d'incapacité s'élevaient à 8,5 milliards de dollars en 2000. Ce montant correspond à la valeur de 85,2 millions de journées de travail perdues pour des raisons personnelles (Boulenger *et al.*, 2012).

Pour gérer les files d'attentes de clients, plusieurs solutions existent. Certaines permettent même de réduire la durée passée dans une salle d'attente. Voici quelques systèmes utilisés au Québec et au Canada.

— **Bonjour Santé** est une application sur une base de système téléphonique pour attribuer des clients à des plages de rendez-vous prédéterminées à l'avance. Aucune gestion adaptative de patient. Aucune gestion de l'absentéisme. Il s'agit d'un système payant pour lequel il faut débourser 15$ par rendez-vous par personne. Le client doit payer avant afin d'obtenir son rendez-vous. Il est possible de recevoir un rappel de l'heure (plage) de rendez-vous.

— **Chronométrique** est un système informatique de distribution de coupons et numéros comme à la boucherie. Dans le cas de la santé, le nombre de coupons est souvent limité, ce qui provoque une file d'attente devant la

porte avant l'ouverture. Ce système ne tient pas compte de l'émission de plusieurs coupons à la même personne. Système payant, le client doit payer entre 3$ à 8$ par visite. Le montant est établi en collaboration avec la clinique lors de l'installation initiale du distributeur de coupons. Une portion de ce montant est retournée à la clinique [2]. Aucune gestion dynamique des patients. Aucune gestion des désistements.

— **iamsick.ca** est une liste de cliniques en ligne (similaire à Yelp) accompagnée d'une application mobile pour trouver des cliniques sur une carte en fonction de la géolocalisation du patient.

— **Doctr** est une solution mobile pour les cliniques privées essentiellement. Il s'agit d'un bottin recherchable (*search engine*) pour trouver des points de services.

> «Affichez votre clinique dans l'application et rejoignez une communauté utilisateurs. Nous offrons plusieurs solutions pour mettre en avant votre clinique au moment où les utilisateurs ont le plus besoin de vos services» [3].

— **MediMap** est essentiellement une liste de cliniques qui offrent des services médicaux. Il s'agit d'une interface web qui prend les requêtes des clients qui les acheminent via courriels aux cliniques sélectionnées. Le système peut envoyer des requêtes que durant les heures d'ouverture. L'assignation et l'acceptation sont faites manuellement par la clinique.

— **Skip the Waiting Room** est un portail web pour trouver un point de service. Il propose quelques informations sur la clinique comme les heures (plages) d'ouvertures, l'adresse, etc. Il n'y a pas d'information sur l'usage des cliniques ou de prise de rendez-vous.

---

2. Selon des informations recueillies auprès de Chronométrique en septembre 2016

3. Source : www.doctr.ca ; avril 2017

— **Twilio** est une compagnie internationale qui propose un système de gestion de messages pour différentes plateformes. Qu'il s'agisse de SMS, courriel ou notification bidirectionnelle, twilio rend ceci possible.
— **Yelp** est un système en ligne qui permet a des utilisateurs de trouver des commerces de proximité, tel que : dentistes, des coiffeurs ou des mécaniciens.

Les systèmes mentionnés ci-haut ne proposent pas d'utiliser le réseau de santé dans son ensemble. Ces systèmes web ou mobiles n'ont aucune intelligence en matière d'assignation, d'ordonnancement, gestion du désistement, gestion des absences ou gestion de l'achalandage. Ces systèmes ne sont pas capables de planifier la demande cliente et d'ordonnancer des calendriers locaux (horaires par clinique) ou globaux, c'est-à-dire pour le réseau de cliniques en entier. Ces systèmes sont pour la plupart des bottins, des listes électroniques accompagnées d'une recherche géolocalisée pour trouver les cliniques avoisinantes en fonction de votre situation géographique, en utilisant une adresse par exemple. De plus, ces systèmes existants sont peu évolués. Le travail et la prise de décision reviennent essentiellement à l'usager ou par l'entremise d'un humain.

L'usager doit lui-même lancer la recherche dans l'interface (similaire à Google) et parcourir les résultats sans même savoir si les cliniques listées sont convenables. On entend par convenable, être en mesure de traiter dans un délai raisonnable. Aucune information n'est disponible pour informer le client sur l'état ou la pertinence de prendre l'une ou l'autre des cliniques, car aucune des solutions n'inclut la planification et la gestion dynamique de la file d'attente. L'état, mentionné précédemment, est composé de plusieurs indicateurs qui visent à informer des humains ou des algorithmes pour aider la prise de décisions. L'état pour nous, est basé sur le volume d'affluence, de temps moyen d'attente, le nombre de traitements, le nombre de désistements et de probabilité de traitement. Pour le patient, quelle

clinique est le meilleur choix ? Une clinique plus proche susceptible de le traiter plus efficacement ? De plus, aucun système n'offre la replanification dans les cas où la situation du réseau ou de la clinique viendrait à changer. En résumé les systèmes existants mentionnés précédemment n'offrent aucune aide à la décision et aucune forme d'intelligence, aucune proposition ou suggestion afin d'optimiser l'accès à la santé.

Théoriquement, il serait éventuellement possible d'utiliser une méthode comme l'apprentissage machine supervisé (ML) pour découvrir à l'aide d'un modèle, celui-ci préalablement construit sur des données, des solutions qui seraient fort acceptable.

Il existe aussi des algorithmes qui sont en mesure de calculer des solutions dites optimales. Ceux-ci sont généralement utiles lorsque le temps n'est pas en cause. Ce type d'algorithme est couteux en temps de calcul et difficilement applicable dans un environnement temps réel.

Dans ce mémoire, nous proposons un système original pour gérer un réseau de cliniques de santé. Notre système est conçu pour gérer, de façon centralisée, un réseau de cliniques établies sur un large territoire.

Le système propose d'éliminer la double-réservation [4] de clients. Nous expliquons, dans la section 1.3.3, les impacts des demandes multiples sur les clients et l'état du système. Notre approche propose une solution innovante afin d'atténuer les effets des désistements clients.

Notre système est basé sur une solution algorithmique et logicielle. Cette solution vise à assigner des clients à des cliniques et leur fixer une heure de rendez-vous

---

4. Un client peut actuellement appeler plusieurs cliniques et demander une plage horaire dans chacune. En anglais *doublebooking*, souvent rencontré dans le monde de l'aviation.

aussi précise que possible. Notre approche optimise un compromis, qui est une somme pondérée de divers paramètres dont :

- la durée d'attente globale, soit le délai entre la prise de rendez-vous et le moment où un client débute sa consultation ;
- le temps de déplacement pour se rendre à la clinique ;
- le temps d'attente passé dans la salle d'attente rattachée à la clinique ;
- la précision de l'heure prévue, soit la différence entre l'heure initialement prévue et l'heure effective de passage.

Notre approche est basée sur des algorithmes d'exploration et des heuristiques qui permettent de générer des solutions proches optimales. Ce système est conçu pour fonctionner en temps réel dans un environnement dynamique. En effet, dans un réseau de cliniques, plusieurs événements surviennent en temps réel, comme l'apparition et la disparition de clients, l'augmentation et la diminution des ressources.

Dans les expérimentations présentées dans ce mémoire, notre système gère une simulation de cliniques de santé, tiré de la réalité du Québec. Nous avons évalué le système proposé à l'aide de simulations. Pour cela, nous avons développé un simulateur qui tente de reproduire, aussi fidèlement que nécessaire, un réseau de cliniques et un ensemble de clients. Nous avons comparé notre système à des systèmes existants. Les résultats obtenus en simulation suggèrent que notre système permet de réduire significativement le temps d'attente passé dans une salle d'attente.

Ce mémoire est organisé comme suit. Le chapitre 1 présente le simulateur que nous avons développé et utilisé comme cadre de référence pour notre approche. Le chapitre 2 définit la problématique dans le cadre de l'assignation de clients à des cliniques médicales. Le chapitre 3 résume l'état de l'art des méthodes existantes

pouvant être appliquées à l'assignation de clients à des cliniques. Le chapitre 4 présente notre système proposé. Puis, le chapitre 5 présente une évaluation empirique de notre approche à l'aide de notre simulateur. Enfin, nous concluons ce mémoire et proposons quelques avenues possibles pour la suite du projet.

CHAPITRE I

SIMULATION D'UN RÉSEAU DE CLINIQUES

Pour concevoir, implémenter, tester, expérimenter et évaluer nos algorithmes d'assignation de clients, il est essentiel de disposer d'un simulateur de réseau de cliniques. Ce chapitre présente le simulateur que nous avons développé dans le cadre de nos travaux. Ce simulateur sera utilisé comme cadre de référence dans les chapitres suivants.

Le simulateur consiste à reproduire virtuellement le fonctionnement du modèle étudié. La simulation, une exécution à l'intérieur du simulateur, constitue l'une des approches importantes permettant d'exploiter celui-ci. Une simulation nous permet notamment de valider ou d'invalider des hypothèses, d'obtenir des informations quantitatives, de valider certaines approximations, d'évaluer la sensibilité d'un modèle à certaines hypothèses ou à certains paramètres ou, tout simplement, d'explorer le comportement du modèle lorsque celui-ci est mal connu ou mal compris (Dyke et MacCluer, 2014).

La simulation de modèles stochastiques a recours à des nombres générés aléatoirement et est connue sous le nom générique de méthode de Monte-Carlo (par référence aux jeux de hasard des casinos). De nombreux problèmes numériques à priori sans aucun rapport avec le hasard ou les phénomènes aléatoires (évaluation d'intégrales, résolution de systèmes linéaires ou d'équations aux dérivées

partielles) peuvent cependant, de manière plus ou moins artificielle, être traduits en termes de modèles stochastiques. La portée des méthodes de Monte-Carlo dépassent donc très largement le cadre de la modélisation de phénomènes aléatoires (Gould *et al.*, 1988; Juillard et Ocaktan, 2008).

Fondamentalement, notre simulateur simule deux types d'entités : des clients et des cliniques. Un client apparaît dans le simulateur lorsqu'un habitant se considère malade et disparaît lorsqu'il se considère comme traité. Les cliniques sont des ressources, chacune située géographiquement à un endroit précis pour accueillir des clients. Les cliniques disposent d'une certaine capacité de traitement proportionnelle au nombre de médecins y travaillant.

1.1 Territoire simulé : le Québec

Pour être le plus fidèle à la réalité que possible, un simulateur doit idéalement être alimenté par des données réelles. C'est ainsi que nous avons choisi de simuler un réseau de cliniques sur le territoire du Québec. Le choix du territoire québécois s'explique par l'accès relativement facile aux données requises.

Tel que présenté à la figure 1.1, le Québec est découpé en 17 régions administratives [1]. Les données qui alimentent le simulateur sont spécifiées en fonction de ces régions.

Notre simulateur ne simule pas tous les aspects du réseau de la santé. La simulation se limite aux cliniques médicales. Nous avons décidé de prendre des cliniques pour trois raisons : (1) il est facile de les situer géographiquement ; (2) les ressources qui traitent le font de cet endroit ; (3) c'est de cette façon et à cet endroit que la population peut recevoir des soins dans l'immédiat. Nous ne considérons pas

1. `http://www.axl.cefan.ulaval.ca/amnord/quebec-Regions_admin-carte.htm`

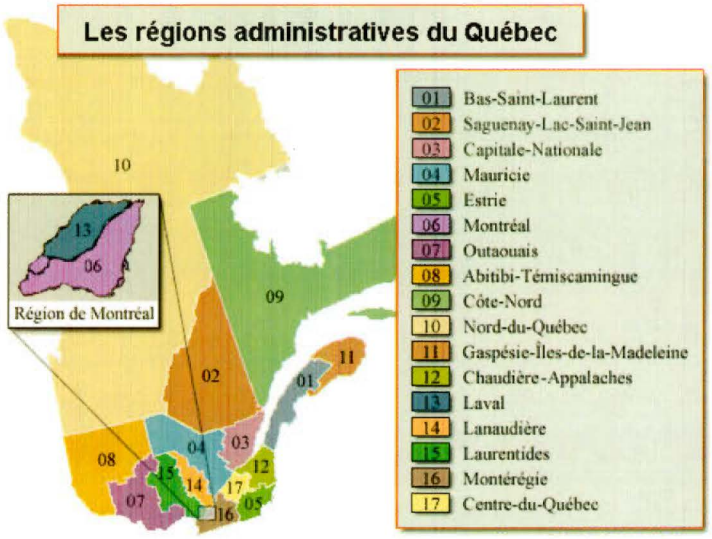


Figure 1.1: Carte des 17 régions administratives du Québec

les hôpitaux, car notre ambition est le traitement de patients ambulatoires. Dans ce mémoire, nous utilisons ambulatoire [2] comme un antonyme de stationnaire qui sera ajouté au glossaire. Stationnaire [3] implique la notion de séjour, y passer la nuit. De façon générale, il n'y a pas possibilité de séjour dans les cliniques du Québec. Elles sont donc des cliniques ambulatoires. Les cliniques privées et les cliniques de spécialités ne sont également pas simulées.

Nous avons extrait (entre 2014 et 2016) à partir du site Web du MSSS une liste qui contient 714 cliniques [4]. Pour chacune de ces cliniques, nous avons son nom et son emplacement géographique en coordonnées latitude et longitude. À noter que certaines cliniques (52) ont dû être éliminées de la simulation en raison de limites d'implémentation. Par exemple, l'implémentation d'arbre-kd que nous utilisons ne

2. définition au glossaire à la page 93

3. `https://blog.prestaflex-service.ch/1397/ambulatoire-et-stationnaire`

4. `http://www.sante.gouv.qc.ca/repertoire-ressources/`

permettait pas d'avoir deux objets (cliniques) à des coordonnées géographiques identiques, c'est-à-dire sur un même point ou à une même adresse. Ceci n'a pas d'impact significatif sur les résultats, car le nombre de ressources traitantes est distribué de façon uniforme dans les cliniques de la région. Nous gardons la couverture optimale de service.

La population du Québec est en sorte la matière première du corps médical. Il va de soi que bien comprendre et avoir toutes les informations la concernant était vital. Le tableau 1.1 de la page 13 présente le portrait de la population qui est présente dans l'étude et ce, pour chacune des régions administratives du Québec.

Au Québec, il y a un peu plus que 19000 médecins selon le site web du CMQ[5] et l'ICIS[6]. Le tableau 1.2 présente la situation au 31 décembre 2013. Ces données ont été utilisées pour nos simulations.

Pour simplifier et augmenter la fluidité de la partie visualisation graphique de la simulation, nous avons utilisé les codes RTA (région de tri d'acheminement)[7] de Postes Canada. Cela correspond aux trois premiers caractères des codes postaux. Ainsi, les patients ayant le même RTA apparaîtront sur le même point géographique. Au Québec, il existe 417 codes RTA. Cela simplifie l'implémentation visuelle (la carte interactive) figure 1.2 de la simulation, car nous n'affichons que le nombre de clients pour chacun de ces points. Lorsqu'un client apparaît dans la simulation, il est affecté à un RTA de façon aléatoire avec une loi uniforme.

---

5. http://www.cmq.org/page/fr/repartition-region.aspx

6. https://secure.cihi.ca/free_products/Rapport_Sommaire_2015_FR.pdf

7. https://www.canadapost.ca/assets/pdf/KB/nps/nps_lettermail_fsalist_jan2016.pdf

Tableau 1.1: Population du Québec par région par groupe d'âge 2014

No	Région	0 – 19	20 – 64	65+	Total
01	Bas-Saint-Laurent	37 506	118 546	44 114	200 166
02	Saguenay–Lac-Saint-Jean	54 469	168 756	54 416	277 641
03	Capitale-Nationale	135 511	459 010	138 256	732 777
04	Mauricie	47 764	159 655	59 164	266 583
05	Estrie	66 404	192 104	62 123	320 631
06	Montréal	393 651	1 283 153	308 652	1 985 456
07	Outaouais	85 178	243 287	54 871	383 336
08	Abitibi-Témiscamingue	33 188	90 258	24 466	147 912
09	Côte-Nord	21 020	58 634	15 501	95 155
10	Nord-du-Québec	15 475	25 635	3 204	44 314
11	Gaspésie Îles-de-la-Madeleine	15 947	54 873	21 516	92 336
12	Chaudière-Appalaches	89 430	251 509	78 804	419 743
13	Laval	96 064	256 822	68 847	421 733
14	Lanaudière	111 062	303 079	78 219	492 360
15	Laurentides	128 707	362 271	95 185	586 163
16	Montérégie	334 539	922 318	251 954	1 508 811
17	Centre-du-Québec	51 177	142 148	46 443	239 768
	Total	1 717 092	5 092 058	1 405 735	8 214 885

Source : ISQ [a]

---

a. `http://www.stat.gouv.qc.ca/statistiques/population-demographie/bilan2015.pdf`

Tableau 1.2: Médecins inscrits actifs par région administrative (2013)

		Nombre de médecins		
No	Région	Total	Généralistes [a]	Spécialistes
01	Bas-Saint-Laurent	515	274	241
02	Saguenay–Lac-Saint-Jean	593	335	258
03	Capitale-Nationale	2 3	934	1 239
04	Mauricie	957	527	430
05	Estrie	859	377	482
06	Montréal	6 306	2 144	4 162
07	Outaouais	639	370	269
08	Abitibi-Témiscamingue	364	207	157
09	Côte-Nord	245	167	78
10	Nord-du-Québec	144	130	14
11	Gaspésie Îles-de-la-Madeleine	311	196	115
12	Chaudière-Appalaches	759	432	327
13	Laval	634	345	289
14	Lanaudière	680	390	290
15	Laurentides	872	532	340
16	Montérégie	2 404	1 360	1 044
17	Centre-du-Québec	957	527	430
	Total	19 412	9 247	10 165

Source : CMQ [b]

---

a. médecins de famille

b. `http://www.cmq.org/page/fr/repartition-region.aspx`

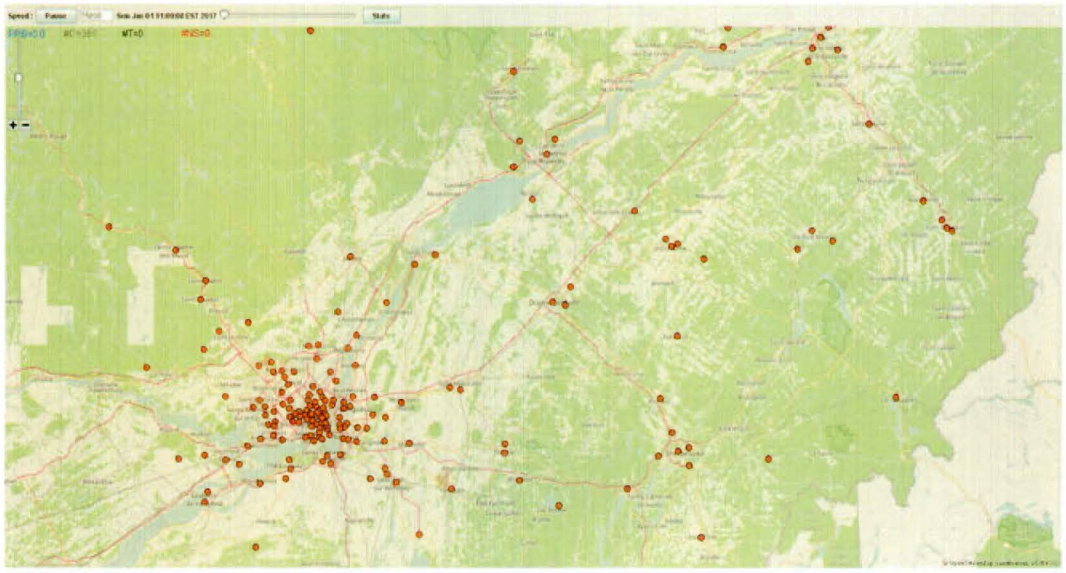


Figure 1.2: Représentation visuelle et géographique des patients

1.2 Temps et simulation à événements discrets

Les systèmes à événements discrets tels que les files d'attente, qui modélisent des phénomènes se déroulant en temps réel, mais dont les évolutions se produisent en des instants ponctuels, constituent une classe importante de modèles stochastiques, et interviennent dans de nombreux domaines d'application dont le nôtre.

Notre simulateur est de type à événements discrets (Brailsford *et al.*, 2014). Les événements sont placés dans une file prioritaire pour être traités en ordre chronologique. À chaque pas de la simulation, le temps est avancé au plus petit temps dans la file prioritaire. Le traitement d'un événement peut engendrer d'autres événements à des dates ultérieures.

Les types d'événements sont :

1. nouvelle requête d'un client qui se déclare malade ;

2. un client annule sa demande ou ne se présente pas, appelé désistement[8] ;
3. un client arrive à la salle d'attente d'une clinique ;
4. un médecin commence le traitement du patient ;
5. un client est traité.

Lorsque la simulation est visuellement présentée (affichage graphique), la progression du temps est limitée par un régulateur de temps, et ce, selon un ratio temps simulé sur temps réel. Ce ratio peut être ajusté via l'interface utilisateur.

1.3 Modèle d'un client

La figure 1.3 présente le diagramme d'états d'un client. Un client a un cycle de vie commençant par son apparition (état «malade») et se terminant à sa disparition (états «désisté» et «vu»). Une fois un client dans l'état «malade», il soumet une requête au système et bascule dans un état «attente d'une réponse». Le système calcule une assignation, c'est-à-dire la clinique où le client doit se présenter. Une première heure de rendez-vous est attribuée. L'heure de rendez-vous pourra être éventuellement révisée par le système. Ensuite, un client attend à la maison jusqu'à ce qu'il soit l'heure de quitter pour se rendre à la clinique. L'heure de départ est l'heure requise pour arriver x minutes avant l'heure de rendez-vous. Le nombre x est tiré aléatoirement par le simulateur 1.3.1. Une fois le client rendu dans la salle d'attente, il attend son rendez-vous. S'il arrive en retard, il perd sa place et disparaît du système. Lorsque son tour vient, le client obtient sa consultation médicale ayant une durée variable de y minutes tirée aléatoirement par le simulateur 1.4.1. Enfin, après la consultation, le client est marqué «vu» et disparaît de la simulation. À noter que le client peut se désister à plusieurs moments. Par exemple, il peut décider de ne pas aller à son rendez-vous ou même quitter la salle

8. terme officiel anglais *no-show* voir section 1.3.3

d'attente.

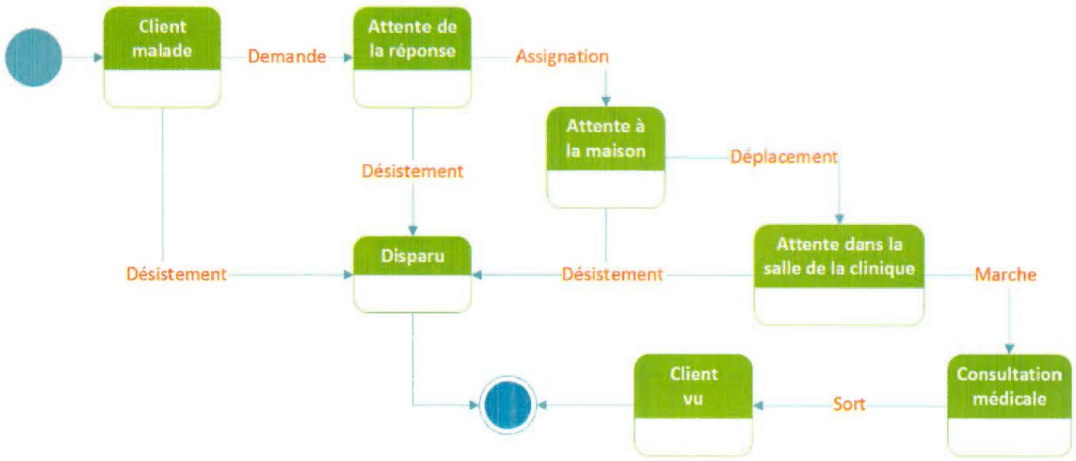


Figure 1.3: Diagramme d'états pour un client

1.3.1 Apparition des clients

Chaque jour, on estime qu'il y a environ $P_j = 75\,000$ clients[9] au Québec qui souhaitent obtenir une consultation médicale. L'apparition des clients n'est cependant pas uniforme dans une journée. Cela est considéré par le simulateur. Le débit d'apparition des clients suit la distribution présentée dans la figure 1.4. L'histogramme défini le taux de clients (f_h) qui demandent une consultation pour chaque heure h de la journée. Par exemple, l'intervalle 6 débute à 6 heure et se termine à 7 heure (exclusivement). Durant cette période, $f_6 = 4$, c'est-à-dire qu'il y apparaitra 4% des patients d'une journée.

Il est à noter que malgré plusieurs tentatives pour obtenir les données officielles de la part de ICIS, nous n'avons pas obtenu des données officielles pour construire cette distribution de la figure 1.4. La distribution représente donc une hypothèse intuitive de la réalité. Nous estimons que cela est raisonnable pour expérimenter

9. Une constante fournie par le MSSS, modifiable dans l'interface

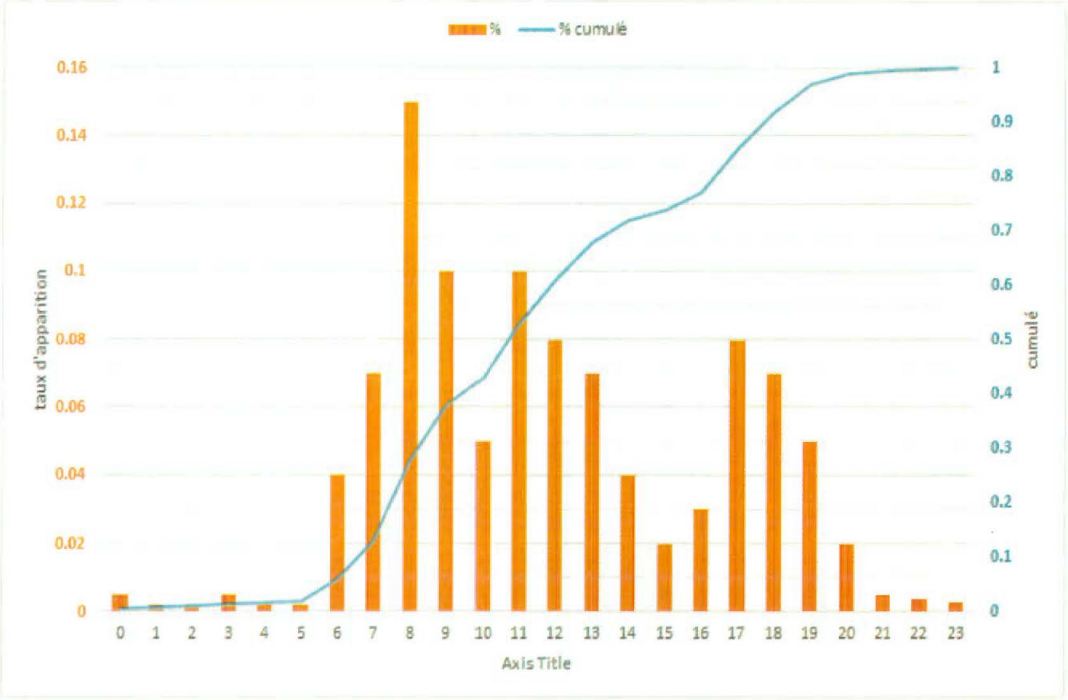


Figure 1.4: Distribution de l'apparition des clients selon heure de la journée

nos algorithmes. Avant d'être mis dans un environnement de production, il faudrait bien sûr tenter d'obtenir des données de meilleure qualité afin de faire des tests plus poussés.

Pour chaque heure simulée, les clients arrivent dans la simulation avec un intervalle de temps déterminé aléatoirement suivant une distribution exponentielle ayant un taux d'arrivée suivant $\lambda_h = f_h \cdot P_j$.

Par exemple, entre 6h et 7h, $\lambda_6 = 4\% \cdot 75\,000 = 3\,000$. Nous pouvons donc dire qu'il y aura en moyenne un client toutes les $\frac{3000}{3600sec \cdot 1000ms} = 833$ millisecondes.

1.3.2 Caractéristiques des clients

Les clients apparaissent dans une région tirée aléatoirement selon le poids démographique de chaque région (voir section 1.1). Les caractéristiques des clients (âge et sexe) sont aussi tirées aléatoirement selon la distribution démographique présentée à la section 1.1. Il y a donc plus de chance qu'un client (patient), généré par le simulateur, soit de la région de Montréal que de la Gaspésie.

1.3.3 Désistements (*no-show*)

Le désistement ou renonciation est un facteur des plus critiques pour ce type de problème. Il provoque un impact majeur s'il n'est pas correctement géré. Depuis des dizaines d'années, les chercheurs ont tenté de contrôler, prédire, expliquer et modéliser ce problème (Cayirli *et al.*, 2012; Chakraborty *et al.*, 2010; Daggy *et al.*, 2010; Goldman *et al.*, 1982; Johnson *et al.*, 2007; Lacy *et al.*, 2004). Le désistement, abandon ou renonciation en français, est toujours présent de nos jours. Il y a plusieurs raisons à son existence.

Une de celles-ci est la sur-réservation (*double-booking*). Elle est produite lorsqu'un client malade veut une place en clinique. Dans la vie réelle, celui-ci ira dans plusieurs cliniques pour ainsi tenter d'obtenir une place. Il réservera dans chacune d'entre elles pour obtenir et augmenter ses chances d'avoir une place à une heure intéressante. Il aura ensuite le choix d'aller où il veut, ce qui génèrera des abandons dans toutes les autres cliniques. Le patient génère ainsi des demandes virtuelles qui ne seront jamais utilisées.

Notre approche résout de façon implicite le désistement provoqué par la sur-réservation. Nous avons préconisé un système unique et central pour l'acceptation de patients. Celui-ci est clé afin d'éliminer les doublons avant de procéder avec

l'assignation. Il s'agit de la première étape d'une saine gestion du flux en entrée.

Maintenant que les patients ne peuvent qu'avoir une seule demande active, une portion importante du désistement a disparu. Mais il n'est pas possible de l'éliminer complètement. Donc, nous allons gérer la portion qui reste et qui peut survenir à tout moment. Voici quelques exemples de désistements qui peuvent survenir dans la vie réelle.

— le patient ne nécessite plus de soins ;
— le patient renonce à bout de patience ;
— le patient est en retard ;
— le patient a simplement oublié.

Dans les deux derniers cas, nous prenons pour acquis que le système est en mesure de prévenir les gens de leurs rendez-vous. Ceci atténue grandement les retards et les oublis.

1.4 Modèle d'une clinique

Tout d'abord, il est important de dire qu'une clinique en tant que telle ne traite pas de client, ce sont les médecins qui effectuent les consultations médicales. C'est aussi dans cette optique que le simulateur a été conçu. De plus, un médecin ne peut traiter que dans les heures d'ouverture de la clinique. La clinique n'est pas ouverte en continu, elle ferme comme dans la réalité. Chaque clinique a une file d'attente qui représente une salle d'attente. On procède aux consultations dans l'ordre de la file d'attente (le plan). Si un client n'est pas là, on parle ici d'un désistement, la clinique cumule alors du temps mort jusqu'au prochain rendez-vous. Lorsqu'un désistement est détecté, la clinique fait une demande de remplacement en temps réel pour trouver un remplacement au client délinquant. La figure 1.5 représente les états successifs dans une journée comme dans la réalité. Il faut aussi noter que

la clinique est en mode attente en tout temps. Ceci veut dire, que la file d'attente peut accueillir des patients en continu, même pendant la fermeture. Donc la figure 1.5 ne présentera pas un état en attente, puisque nous présentons seulement les états d'actions.

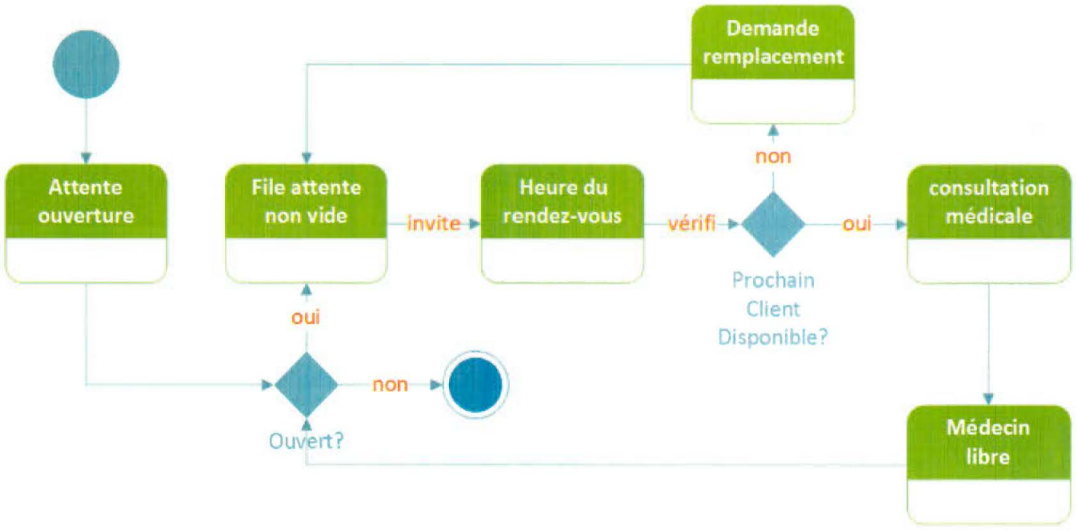


Figure 1.5: Diagramme d'états pour une clinique

Dans le simulateur il n'y aurait pas d'état fin. Ici l'état fin représente l'heure de fermeture. Dans la simulation c'est un instant très bref qui envoie la clinique à l'état du début en mode attente de l'ouverture.

Les cliniques ouvrent et ferment comme dans la réalité, soit de 09h00 à 22h00. Il est impossible d'assigner un rendez-vous dans la dernière heure précédant la fermeture. Le choix des heures d'ouverture est un choix d'implémentation. Les cliniques du Québec doivent garantir des périodes de fonctionnement selon les lois en vigueur, nous avons basé notre choix sur ce règlement.

Les médecins d'une région sont répartis de façon uniforme dans chacune des cliniques selon la formule (1.4).

Quelques variables sont requises par le simulateur et la simulation. Celles-ci sont

les bases pour que la simulation soit conforme et le plus proche de la réalité que possible.

- R_c : est le nombre de ressources pour la clinique c;
- R_t : ressources totales au Québec;
- C_r : nombre de cliniques dans une région r;
- R_{st} : ressources médecins spécialistes total;
- R_{ot} : ressources médecins omnipraticiens total;
- R^{sr} : ressources médecins spécialistes de la région;
- R^{or} : ressources médecins omnipraticiens de la région;
- k : une constante réelle, qui représente le taux de ressources actives.

$$R_{ot} = \sum_{i=1}^{17} R_i^{or} \tag{1.1}$$

$$R_{st} = \sum_{i=1}^{17} R_i^{sr} \tag{1.2}$$

$$R_t = R_{ot} + R_{st} \tag{1.3}$$

$$R_c = \left\lfloor \frac{R_{or} \times k}{C_r} \right\rfloor \tag{1.4}$$

1.4.1 Durée de consultation

La durée des consultations est simulée par un nombre aléatoire suivant une distribution de la loi normale (gaussienne). Dans notre cas, nous avons fixé le paramètre $\mu = 28$ et $\sigma = 5$ tel qu'illustré à la figure 1.6. Ceci permet d'obtenir des durées de consultations entre 14 et 44 minutes dans 99% des cas. Il nous semble raisonnable et sensiblement réaliste de débuter nos expérimentations avec ces paramètres. Ce

paramètre peut être changé durant nos expérimentations. Dans une future version du simulateur, les paramètres μ et σ pourront varier selon l'âge du client et seront fixés à partir de données d'historiques de consultations.

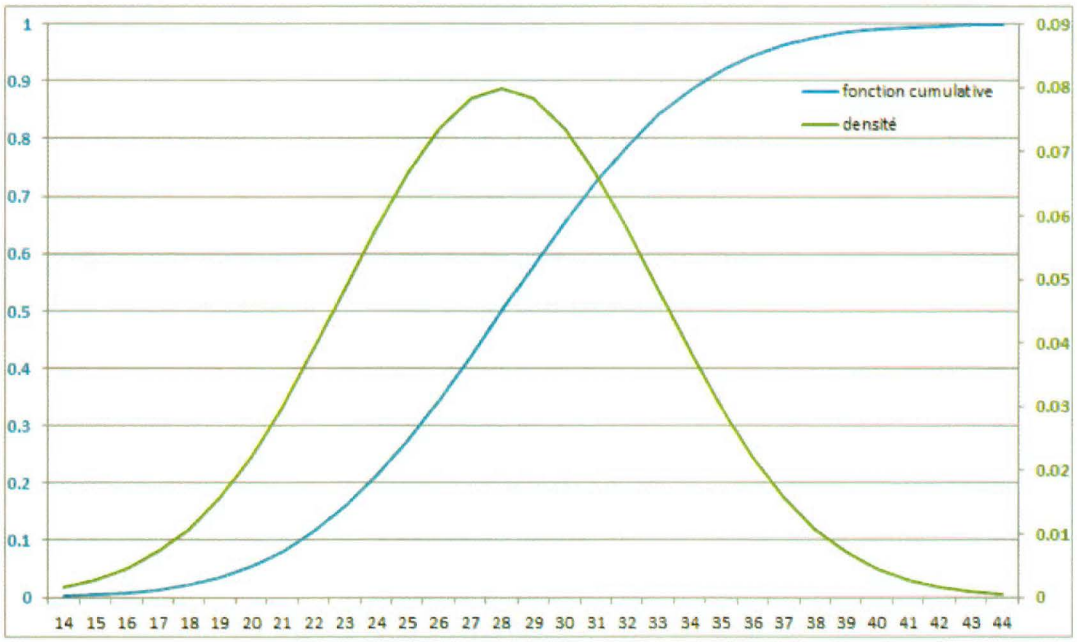


Figure 1.6: Distribution durée des consultations

1.5 Détails d'implémentation

De nombreux outils logiciels sont disponibles pour écrire des programmes de simulation. L'utilisation d'un langage compilé tel que C, C++ ou Java, garantit en général une exécution rapide, ce qui peut être crucial pour effectuer des simulations de notre envergure. Ils offrent aussi de contrôler le fonctionnement de la simulation dans un niveau de détails qui est appréciable. Dans le cas présent, nous avons choisi Java. Ce langage est mature et une quantité appréciable de librairies sont disponibles. Java, dans notre cas la version 8, est aussi choisie pour sa simplicité et la facilité d'y inclure des aspects visuels. Car, pour la suite, les résultats des simulations seront rendus à la fois sous une forme interactive afin de

constater comment le simulateur se comporte. Finalement, nous avons eu recours à des captures d'écran généré à des moments précis pour comparer les résultats.

Notre simulateur est une application Java avec plusieurs fils d'exécution (*multi-thread*). Le simulateur s'exécute en continu sans aucune limite de temps, de son lancement jusqu'à son arrêt. Nous considérons ce segment d'exécution comme une expérimentation. L'application est composée de trois fils d'exécution. Un premier fil d'exécution pour la gestion du temps, un deuxième pour l'exécution du modèle, un troisième pour la gestion d'événements temporels. La différence entre l'exécution du modèle et la gestion d'événements temporels est minime mais tout de même importante. L'exécution du modèle est considérée comme le planificateur. Il assigne et prend les décisions en plus de traiter les clients. La gestion des événements temporels est considérée comme un processus qui écoute afin de gérer et détecter les cas problèmes comme les désistements. Une fois un désistement détecté, un événement avec importance est créé pour qu'il soit pris par l'algorithme principal présenté à la section 4.2.3. Il faut voir ceci comme un fil d'exécution avec priorité haute pour les replacements et un fil exécution normal pour les clients qui viennent d'apparaitre.

1.6 Interface graphique

Plusieurs interfaces visuelles ont été incluses pour valider les résultats produits par le simulateur.

1.6.1 Carte

En premier lieu, il s'agit de la gestion visuelle sur une carte qui est rafraîchie en temps réel du début jusqu'à la fin de l'expérimentation. La figure 1.7 montre la composante visuelle qui dispose géographiquement les clients. Ils sont visibles sous

la forme de points rouges. Plusieurs clients peuvent coexister sur un même point (coordonnée [latitude, longitude]), le nombre est indiqué.

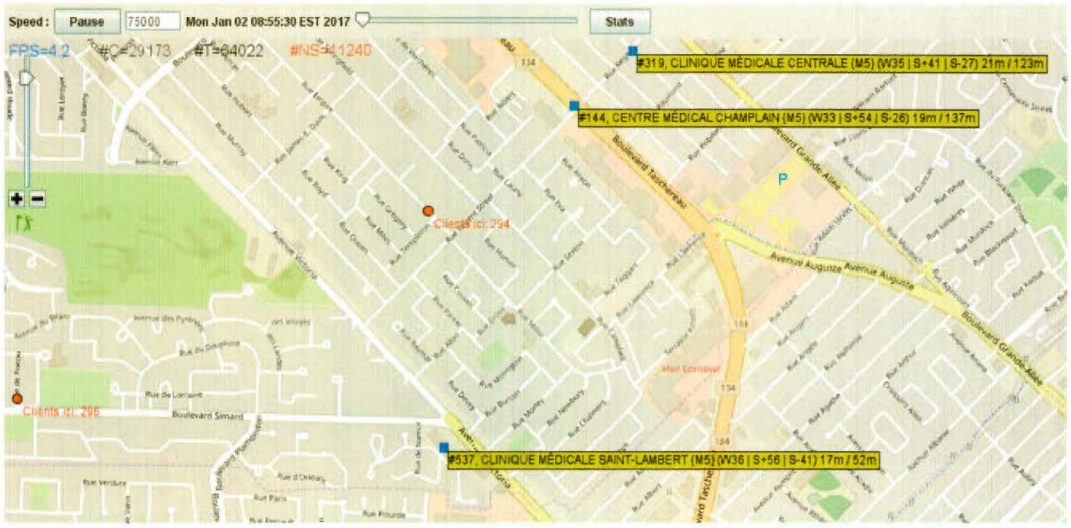


Figure 1.7: Capture d'écran qui présente les patients géographiquement

La figure 1.7 est un agrandissement de la forme originale qui ne se limite pas aux clients, car nous avons aussi les cliniques géolocalisées représentées par les points bleus. Pour chacune des cliniques, il est possible de consulter le nombre de clients dans la file d'attente, le nombre traité, ainsi que le délai de traitement moyen. Toujours dans la figure 1.7, les variables, pour la clinique #537 , clinique médicale Saint-Lambert, sont les suivantes :

— M5 : Nombre de médecins (5) ;

— W36 : nombre de patients en attentent (36) ;

— S+ : nombre de consultation débutées à l'heure prévue ;

— S- : nombre de consultation débutées en retard (41) (même de 61 secondes) ;

— 17m : durée des consultations en moyenne (17 min) ;

— 52m : durée de l'attente en moyenne (52 min).

1.6.2 Plan

Les figures 1.8 et 1.9 présentent le plan qui est généré par le planificateur. Un plan est en fait l'horaire qui contient une série de tâches qui devront être prises en charge ou exécutés. Dans notre cas les tâches sont exécutes dans le temps, c'est pourquoi nous pouvons le considérer comme un horaire. Ce qu'il faut comprendre dans la figure est l'allure globale. Il ne sert à rien de s'attarder à chacun des patients. D'un seul coup d'œil, il est possible de constater que le plan (l'horaire) est raisonnable. Si tous les rendez-vous étaient à la même heure il y aurait potentiellement un problème.

La figure 1.8 présente les patients qui ont été traités et à la figure 1.9, les clients qui le seront prochainement. Nous devons considérer les deux figures mentionnées précédemment comme un tout (une seule). Car il s'agit de la partie 1 de 2 et 2 de 2. Sur l'axe des ordonnées les numéros de patients, sur l'axe des abscisses le temps. La ligne rouge est la durée de l'attente et la ligne bleue est la durée estimée du rendez-vous. Cette figure représente les données pour une seule clinique de 08h00 à 13h00 le jour suivant. Lorsque le rouge disparaît, il s'agit de clients traités qui ne sont plus dans le statut attente, ils sont donc en traitement ou déjà traités. Le plan est la représentation visuelle de la file d'attente telle que présentée par le système en temps réel.

1.6.3 Statistiques

Nous présentons ici la gestion visuelle des données (figure 1.10) et des statistiques (figure 1.11) [10]. Ce sont deux interfaces visuelles qui ont permis de valider efficacement les résultats obtenus durant les expérimentations.

---

10. jour 2, 08h55.

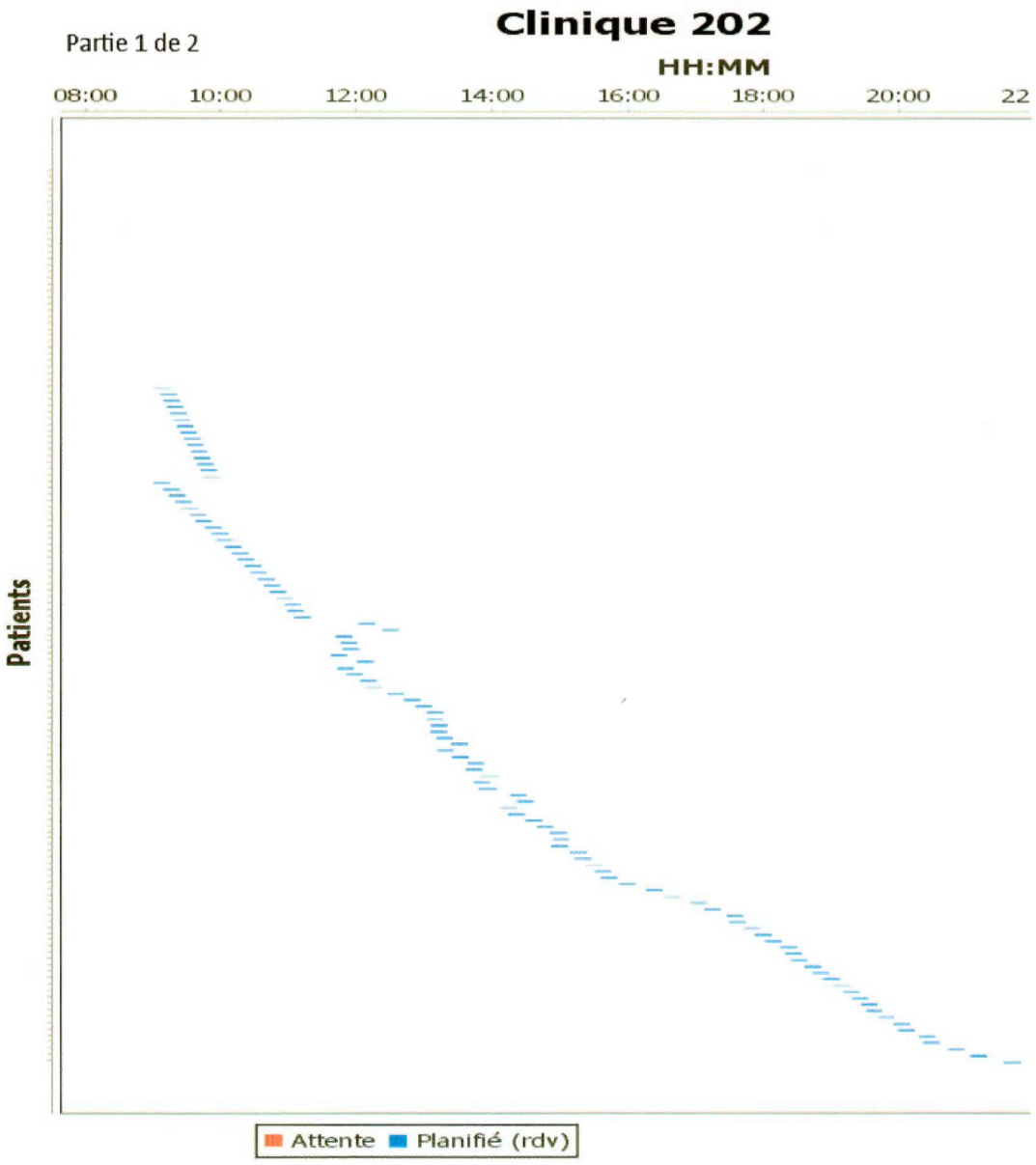


Figure 1.8: Le plan, Gantt, rendez-vous pour une clinique, partie 1

La figure 1.10 représente la situation de l'attente dans tout le réseau en temps réel. Il est possible d'obtenir les fluctuations de l'attente et le nombre clients en attente en fonction du temps. Ce graphique est mis à jour en temps réel. La figure est une capture prise cinq minutes avant l'ouverture des cliniques, soit le jour 2

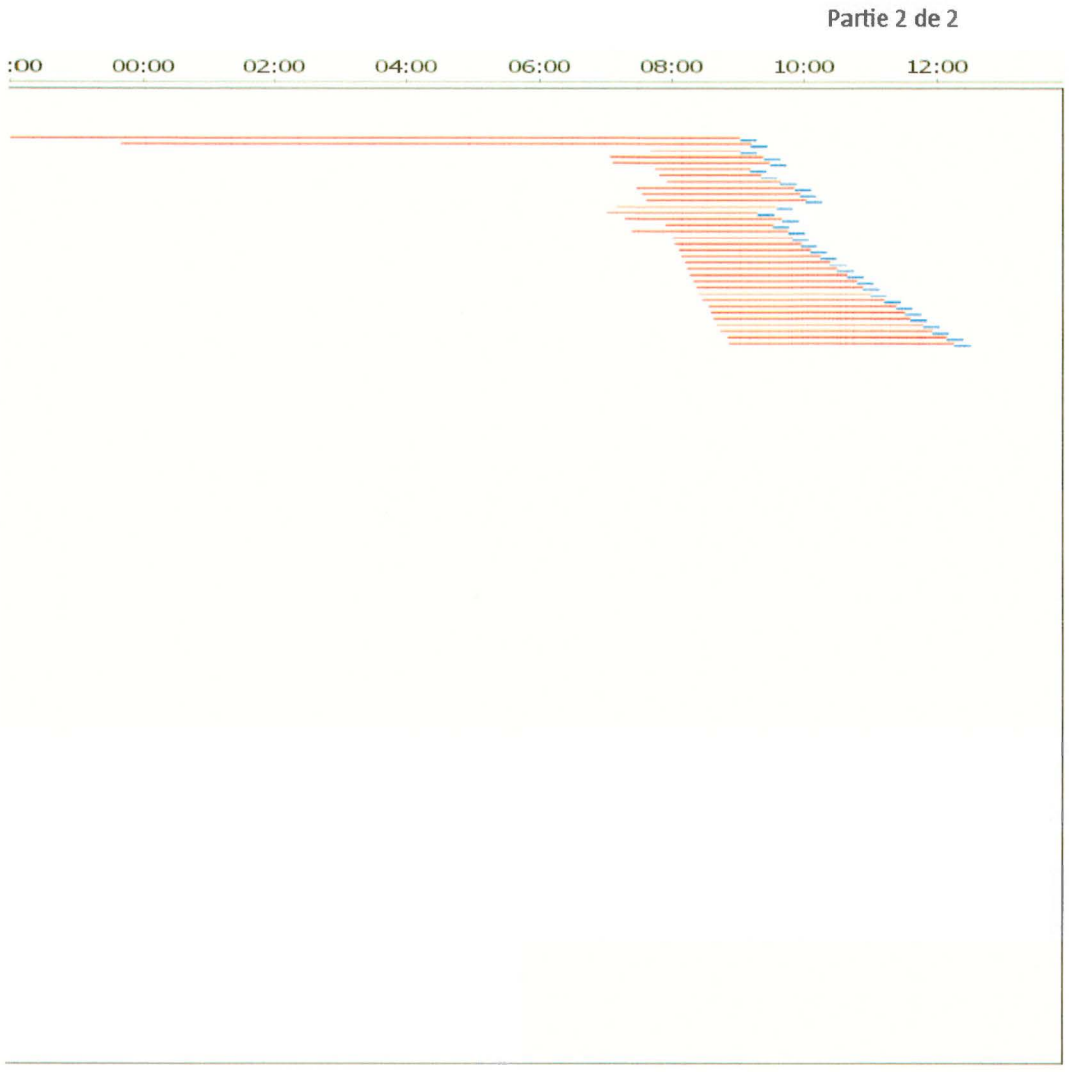


Figure 1.9: Le plan, Gantt, rendez-vous pour une clinique, partie 2

à 08h55. Il faut noter que le résiduel, tous les clients qui ne sont pas traités dans la journée, ceci avant que la clinique soit fermée, seront traités dans la journée suivante. Il est possible de voir que la ligne rouge de la figure 1.10 n'est pas à zéro, il s'agit de résiduel (*overflow*).

La figure 1.11 est purement statistique. Elle nous indique de quelle façon le système

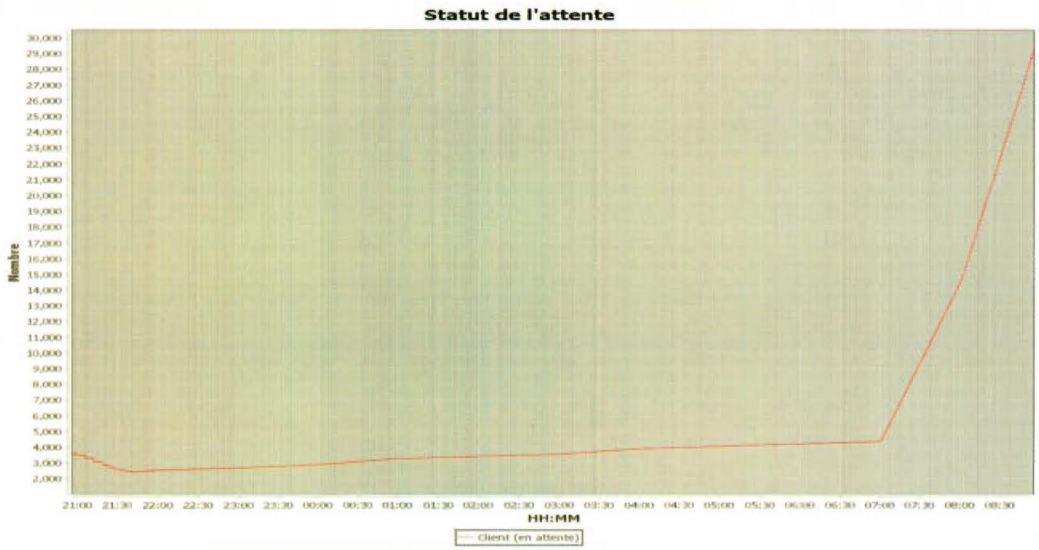


Figure 1.10: Le nombre total de patients dans tout le réseau selon l'heure

traite les clients. Ce que nous devons comprendre de celle-ci, c'est que s'il y a du vert le système est bogué. Il ne peut pas y avoir de l'attente si le rendez-vous est positif (+), c'est-à-dire que les clients sont vus avant ou à l'heure de rendez-vous programmé. Par opposition, le moins (-) indique l'inverse, soit qu'il comptabilise les clients qui ont été vus après l'heure de rendez-vous. Le bleu nous donne le nombre de clients dans la région qui sont vus en retard. Le jaune indique de combien de minutes en moyenne.

Dans la figure 1.11 toujours, nous avons approximativement 1000 clients en retard pour une moyenne inférieure à 4 minutes, sur un peu plus de 64 000 traitements au moment de la capture soit après 57 heures de temps de simulation. Capturé le jour 2 à 08h55 juste avant l'ouverture théorique des cliniques de 09h00.

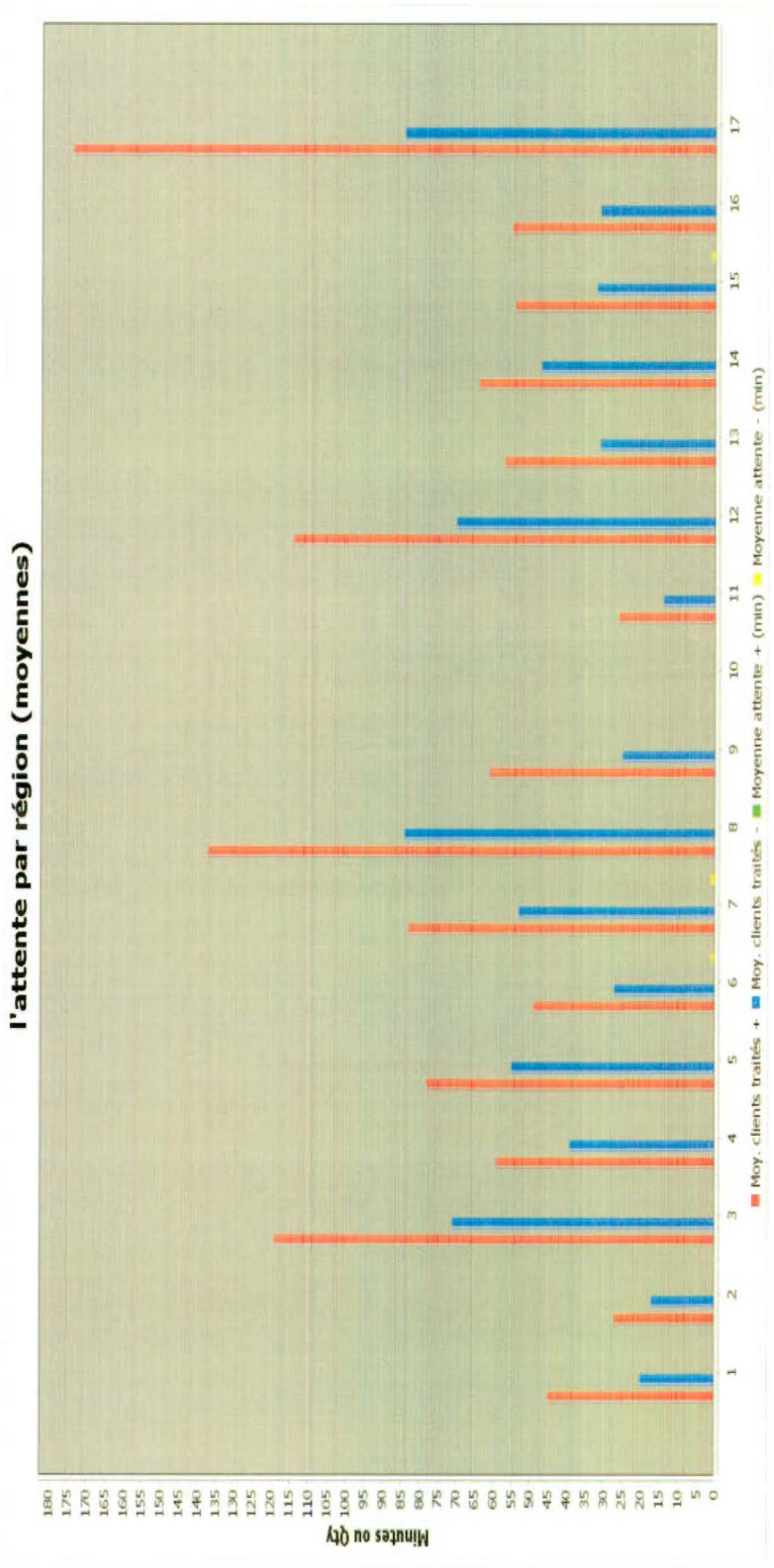


Figure 1.11: Statistiques sur le délai moyen et le nombre de clients moyen par région

CHAPITRE II

PROBLÉMATIQUE : ASSIGNATION EN TEMPS RÉEL DE CLIENTS À DES CLINIQUES

La problématique traitée dans ce mémoire est l'assignation en temps réel de clients à des cliniques. Concrètement, lorsqu'un client apparaît dans le système, on doit l'assigner à une clinique en lui disant à quelle heure il doit s'y présenter. **Fondamentalement, il s'agit d'un problème d'optimisation multicritères** (Roy, 2013). On désire répondre rapidement au client tout en minimisant son attente. Une difficulté particulière de la problématique est son caractère à temps réel. En effet, il s'agit d'un système dynamique, en continu.

Pour bien exprimer notre pensée, nous allons faire le rapprochement avec un système de gestion de colis qui, à un moment précis, est figé dans le temps pour permettre de traiter tous les nouveaux colis. Les assignations sont ainsi effectuées sur une base de lot (*batch*). Quel colis sera pris en charge par quel camion. Un plan en émergera. Il s'agit d'un trajet ou d'une route que le livreur prendra pour faire les livraisons.

Cette problématique n'est pas nouvelle. Un exemple est connu sous le nom du problème du voyageur de commerce *Traveling Saleman Problem* (TSP). Le problème du voyageur de commerce consiste à trouver le chemin minimal ou optimal qui permet au voyageur de visiter les n villes mises sur sa route. Un objectif est

initialement énoncé tel que nous voulons minimiser le temps total ou encore minimiser la distance totale du parcours. Lorsque nous voulons optimiser les deux, il s'agit d'une optimisation multi-objectifs.

2.1 Critères d'optimisation

Notre approche optimise un compromis, qui est une somme pondérée de divers paramètres dont :

- tt_x la durée d'attente totale d'un client x, soit le délai entre la date de prise de rendez-vous (date d'apparition) (da_x) et le moment où un client débute sa consultation (dc_x), $tt_x = dc_x - da_x$;
- td_x le temps déplacement requis pour un client x pour se rendre à la clinique ;
- tp_x la précision de l'heure prévue, soit la différence entre l'heure initialement (dri_x) prévue et l'heure effective du début de la consultation (dc_x) (2.1).

$$tp_x = si\ ((dri_x - dc_x)\ <\ 0)\ donc\ |dri_x - dc_x|\ sinon\ 0 \tag{2.1}$$

Pour compléter la définition de la formule multicritères, nous avons fixé les valeurs (poids) des critères de la façon suivante :

- $\alpha_t = 3$, poids pour le temps d'attente total ;
- $\alpha_d = 9$, poids pour le temps de déplacement ;
- $\alpha_p = 1$, poids pour le temps dans la salle d'attente (précision).

Nous aurions pu définir les poids de façon différente, mais les poids établis nous semblent conformes et réalistes au cas à l'étude. Le poids le plus grand est celui qui coûte le plus, donc le plus important à nos yeux. Le but est d'accorder un poids plus important à la distance (temps de déplacement) et ensuite à l'attente dans la

salle d'attente. Les autres critères sont utilisés afin de départager un résultat versus un autre. Il est important de noter que les cliniques ne peuvent pas accumuler des clients outre mesure, avec la grille poids établi. Nous préconisons une distribution équitable des clients dans le réseau de cliniques.

$$minimize \sum_{x \in X} (\alpha_t tt_x + \alpha_d td_x + \alpha_p tp_x) \tag{2.2}$$

Nous allons utiliser une approche statistique pour guider le planificateur sur la durée probable, et ceci pour chaque clinique.

2.2 Plan (horaire) en sortie

En sortie, le système doit produire des files d'attente. Visuellement, ceci peut être représenté par des graphiques de Gantt tel que la figure 2.1.

Chaque clinique aura son plan en mémoire. Le plan correspond à une file d'attente avec des informations utiles. La file d'attente est en fait une structure de données qui n'est pas persistée sur disque. Elle reste en mémoire pour la durée de l'expérimentation. Une file contient des identifiants clients à servir, chacun annoté de l'heure de passage planifiée, en plus de maintenir la date/heure à laquelle le patient est apparu.

2.3 Problématique double

Ce mémoire veut répondre à deux problématiques, la première qui n'est pas scientifique : sommes-nous en mesure de proposer une solution globale aux problèmes que vivent les patients ? Soit, obtenir un lieu et un médecin pour traiter les patients dans un délai raisonnable.

La deuxième problématique est que scientifiquement, nous sommes confrontés à un problème NP-complet qui vise à optimiser, on parle ici, de minimiser le coût pour un client ou maximiser l'usage d'une clinique. Essentiellement, nous voulons minimiser une fonction f en tenant compte de plusieurs critères et nous voulons effectuer ce calcul le plus rapidement possible.

2.4 Ce qui rend le problème complexe

D'abord, nous pouvons considérer la taille du problème en utilisant la matrice de complexité. Quoi, Où et Quand, sont les questions à se poser. Quelle ressource (où) devra exécuter quoi et quand ? Utiliser cette simple classification, sans tenir compte des contraintes diverses, nous permet d'obtenir une approximation de l'envergure du problème (Wall, 1996).

L'incertitude et la nature dynamique des problèmes réels (de la vie) nous forcent à trouver des solutions approximatives car les solutions dites optimales ne seraient pas en mesure d'être accomplies. Plusieurs raisons expliquent qu'une solution ne soit pas satisfaite. La première peut être le temps. L'autre est liée aux contraintes qui réduisent et limitent de façon significative l'espace de recherche et, par le fait même, la découverte d'une solution optimale (Wall, 1996).

Voici quelques problèmes connus en science présents dans ce travail :

- optimisation combinatoire clients, cliniques ; problème voyageur de commerce (TSP) (Charon *et al.*, 1996; Ulungu et Teghem, 1994) ;
- problème multicritères ou multi-objectifs : distance, attente, nombre ; problème du sac a dos (KP) (Ulungu et Teghem, 1994) ;
- représentation de l'espace de recherche multi-dimensionnel et problème de situation géographique (LP) : arbre-kd (Bentley, 1975; Ulungu et Teghem, 1994).

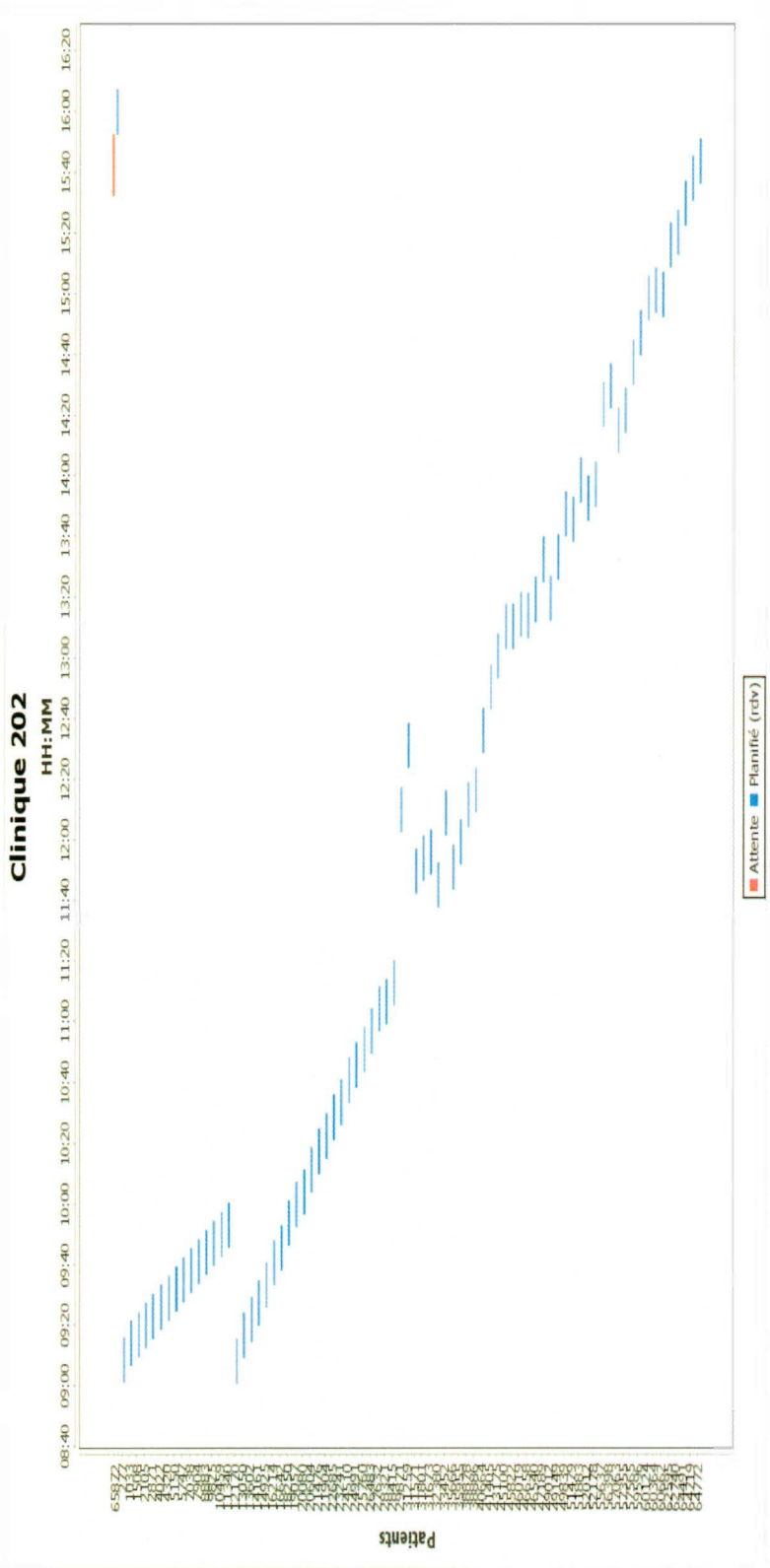


Figure 2.1: Le plan Gantt des rendez-vous pour une clinique

CHAPITRE III

ÉTAT DE L'ART

Ce chapitre fait un survol des approches existantes dans les domaines de la planification et de l'ordonnancement et de la théorie des files d'attentes, domaines centraux de ce travail. Nous présentons dans cette section des travaux qui d'abord se penchent sur les problématiques que vivent les points de services dans le domaine de la santé. Par la suite, des travaux qui traitent de méthodes (qui ne sont pas nécessairement informatiques) pour améliorer la situation de l'attente. Finalement, des méthodes issues de la science ou de l'informatique.

3.1 Identification des problèmes

Plusieurs travaux et recherches nous indiquent que de longues files d'attente, pour des cliniques médicales, causent un grand nombre de rendez-vous manqués. Cela a pour effet de réduire considérablement l'efficacité des opérations cliniques. Suite à ce constat, un nouveau concept de gestion des patients ambulatoires et de planification de rendez-vous est apparu dans les années 1990 intitulé, *advanced access scheduling*, *open access scheduling*, et *same-day scheduling*. Ces méthodes permettent de prendre en charge les patients sans tenir compte des raisons ou du type de mal, et ce, à l'intérieur de 12 à 72 heures (Herriott, 1999; Murray et Tantau, 2000). Le principe de cette méthode est qu'elle s'appuie sur le principe

just-in-time qui se retrouve et qui a fait ses preuves dans l'industrie manufacturière.

L'implémentation de *advanced access scheduling* montre une amélioration de la continuité des opérations cliniques, ainsi que d'avoir augmenté l'utilisation des ressources (cliniciens, médecins, etc.), en plus d'augmenter la satisfaction de clients, et ce en réduisant les coûts de soins de santé (Herriott, 1999; Murray et Tantau, 2000; Kennedy et Hsu, 2003; Kodjababian, 2003; Murray *et al.*, 2003; O'Hare et Corlett, 2004; Hassol *et al.*, 2004). Il faut par contre mentionner que l'implémentation de la planification avancée de patients ambulatoires requiert une transformation toute en profondeur de la part des fournisseurs de services. Les cliniques doivent aligner leurs capacités à la quantité de demandes patients (Kodjababian, 2003; Murray et Berwick, 2003). Il est aussi mentionné qu'une disponibilité inappropriée de capacité de traitement (places en clinique) qui n'est pas correctement calculée va résulter dans l'échec de la planification avancée.

3.2 Renonciation

Lorsqu'un patient attend dans une file d'attente, il peut décider de renoncer au service, car il ne souhaite plus attendre. Ce phénomène, appelé renonciation (*Reneging*), est une caractéristique importante de nombreux systèmes de santé. La probabilité que des patients renoncent augmente habituellement avec la longueur de la file d'attente ou/et par l'estimation approximative faite par le patient du temps qu'il estime devoir attendre pour être servi. Dans les systèmes où la demande dépasse la capacité de service, la renonciation est la seule façon qu'un système atteigne un état d'équilibre dysfonctionnel (Hall *et al.*, 2013).

3.3 Files prioritaires

Dans la plupart des établissements de soins de santé, à moins qu'un système de rendez-vous ne soit mis en place, la discipline de la file d'attente est soit de type *FIFO* ou de type triage. Ce qui correspond à un ensemble de classes de patients qui ont des priorités différentes (comme dans un service d'urgence, qui traite les patients présentant des blessures mortelles avant d'autres blessures plus légères).

McQuarrie (1983) montre qu'il est possible, lorsque l'utilisation est élevée, de minimiser les temps d'attente en donnant la priorité aux clients qui ont besoin de délais de service plus courts. Cette règle est une forme de la règle du temps de traitement la plus courte qui est connue pour minimiser les temps d'attente. Elle se retrouve rarement dans la pratique en raison de l'injustice perçue (à moins que cette classe de clients ne reçoive un prestataire de service[1] dédié, comme dans les systèmes de contrôle des supermarchés) et de la difficulté à estimer les temps de service avec précision.

Lorsque les patients arrivés sont placés dans des files d'attente différentes, dont chacune a une priorité de service différente, la discipline de la file d'attente peut être préemptive ou non préemptive. Dans ce dernier cas, les patients de faible priorité reçoivent du service tant que le traitement n'est pas complété, et ce, même si un patient hautement prioritaire arrive et que tous les serveurs[2] sont occupés. Dans la discipline de la file d'attente préemptive, cependant, le service à un patient de faible priorité est interrompu et sera repris ultérieurement. Green (2006) présente des modèles pour les deux disciplines de files d'attente.

---

1. aussi appelé serveur, pourrait être une unité de traitement comme un ordinateur ou un robot

2. terme informatique lié à la discipline, qui exécute le service

Siddharthan *et al.* (1996) analysent l'effet sur les temps d'attente des patients lorsque les patients de soins primaires utilisent le service d'urgence. Ils proposent une discipline prioritaire pour différentes catégories de patients et ensuite une discipline de premier rang pour chaque catégorie. Ils constatent que la discipline prioritaire réduit le temps d'attente moyen pour tous les patients; cependant, alors que le temps d'attente pour les patients prioritaires diminue, les patients moins prioritaires souffrent d'un temps d'attente moyen plus long.

Haussmann (1970) étudie la relation entre la composition des files d'attente prioritaires et le nombre d'infirmières répondant aux demandes d'hospitalisation. La recherche constate que l'augmentation légère du nombre de patients affectés à une infirmière et / ou un mélange de patients avec plus de demandes de priorité élevée entraîne des délais d'attente très importants pour les patients de faible priorité.

Worthington (1991) analyse le transfert de patients de médecins ambulatoires aux médecins hospitalisés. Le patient reçoit l'un des trois niveaux prioritaires. Sur la base du niveau de priorité, il existe une période de temps standard avant laquelle un patient référé devrait être programmé pour voir le médecin hospitalier. Le modèle suppose une capacité de patients suffisante pour traiter la catégorie de priorité la plus élevée dans son temps standard et propose de partager la capacité de service restante parmi les niveaux de priorité inférieurs de telle sorte qu'ils dépassent chacun leurs temps de cible standard par le même pourcentage.

3.4 Conception système

Parce que l'attente patient n'est pas souhaitable, limiter les temps d'attente est un objectif important lors de la conception d'un système de soins de santé. Cette section examine les travaux sur la détermination de la capacité du système en fonction des objectifs et des exigences du système souhaité. Les variables d'intérêt

sont généralement des effectifs, des niveaux, des lits ou d'autres ressources clés.

Bailey (1954) établit d'abord l'existence de cliniques ambulatoires et d'hospitalisation d'une capacité de seuil qui se situe au point où l'offre de services est égale à la demande. Lorsque le nombre de serveurs est inférieur à ce seuil, une clinique développe une file d'attente infinie. Légèrement au-dessus de ce seuil, le temps d'attente et la longueur de la queue sont faibles. Il soutient qu'il est donc suffisant de concevoir pour une capacité supérieure à la demande attendue (avec une erreur stochastique comptabilisée) d'une valeur de 1 ou 2. Les longues listes d'attente sont probablement le résultat d'un arriéré accumulé qui peut être épuisé par une augmentation temporaire de l'offre. Les variations saisonnières de l'offre entraîneraient également une forte augmentation de la longueur de la liste d'attente.

3.5 Minimiser le coût

Déterminer la capacité du service en minimisant les coûts dans un système de file d'attente pour délivrer des soins de santé est un cas particulier de la conception du système. La plupart des recherches attribuent des coûts au temps d'attente du patient et à chaque service[3]. Après avoir modélisé le système à l'aide de la théorie des files d'attente, la réduction des coûts se réduit à un exercice consistant à trouver l'allocation de ressources qui coûte le moins ou qui génère le plus de profit.

Keller et Laughhunn (1973) ont décidé de déterminer la capacité avec les coûts minimaux requis pour servir les patients au centre médical de l'Université Duke. Ils trouvent que la capacité actuelle est bonne, mais doit être redistribuée en temps réel pour tenir compte du modèle d'arrivée des patients.

---

3. il peut s'agir de médecins ou de départements qui délivrent des services

Plusieurs cliniques ou pratiques médicales permettent de réserver des rendez-vous des mois à l'avance. DeLaurentis *et al.* (2006) soulignent que l'abandon du patient sans annuler son rendez-vous pourrait entraîner un gaspillage de ressources. Ils proposent de mettre en place des systèmes de rendez-vous à court terme basés sur une analyse de réseau de file d'attente adaptée aux réalités d'une clinique ambulatoire particulière. Leur approche suppose la disponibilité d'un certain nombre de médecins (membres du personnel) qui peuvent être répartis entre les différentes cliniques du réseau selon plusieurs combinaisons. Une combinaison est choisie en fonction de l'utilisation et de la durée attendue du patient dans la clinique. Les mises en œuvre de ces idées n'ont pas amélioré le système de rendez-vous, un échec qu'ils attribuent à la clinique en utilisant de nombreux médecins et aux patients incapables d'organiser des visites auprès de leur médecin de première ligne avec un court préavis.

3.6 Système de rendez-vous

Par rapport aux systèmes sans rendez-vous, les systèmes avec rendez-vous réduisent la variabilité d'arrivée et les temps d'attente dans la clinique. Cependant, il est important de noter que les systèmes avec rendez-vous favorisent une attente à l'extérieur du lieu de rendez-vous puisque l'heure est connue à l'avance. Bien sûr, parce que ce n'est pas à l'établissement, cette attente peut être un temps productif et, par conséquent, coûte moins cher pour le patient. De plus, il n'occupe pas d'espace dans les salles d'attente de l'installation. Un problème majeur a été de réduire les temps d'attente des patients sans provoquer une augmentation significative des temps morts du médecin qui représentent un coût indirect et direct important.

Bailey (1952, 1954) propose (a) un intervalle de rendez-vous et (b) l'heure d'arrivée du consultant (médecin) en tant que deux variables qui déterminent l'efficacité d'un système de rendez-vous. Afin de trouver un équilibre entre le temps d'attente du patient et le temps d'inactivité du consultant, d'abord, déterminer les valeurs relatives du temps patient et du temps de consultation. Le rapport du temps total perdu par tous les patients au temps d'inactivité du médecin devrait être égal à la valeur du temps du médecin par rapport aux patients. Il choisit d'assigner des heures de rendez-vous individuelles à des intervalles égaux au temps moyen de traitement du patient et constate que le consultant devrait arriver en même temps que le deuxième patient.

Brahimi et Worthington (1991) élaborent un système de rendez-vous pour réduire le nombre de patients en file d'attente à tout moment et réduisent le temps d'attente du patient sans augmenter significativement le temps d'inactivité du médecin. Ils explorent l'effet des patients qui ne se présentent pas pour leur rendez-vous. La clinique commence avec un certain nombre de patients en attente et un nombre maximum de patients autorisés à tout moment.

3.7 La théorie des files d'attente

Les théories des files d'attente sont des outils extrêmement puissants et vastes permettant de prendre en compte et de modéliser les goulots d'étranglement dans les processus des entreprises soit au niveau de la logistique, des centrales téléphoniques, de l'exécution de requêtes SQL sur les serveurs, des caisses de grands magasins ou encore de l'usage des salles de toilettes des grands stades sportifs en fonction des hypothèses et contraintes de départ.

Dans un système de file d'attente, nous devons identifier les probabilités des propriétés suivantes : requêtes en entrée, le temps de service et la discipline de service.

Le processus d'arrivée peut être caractérisé par une distribution des temps inter-arrivées Ti de client par $A(t)$ tel que :

$$A(t) = P(Ti < t) \tag{3.1}$$

Le temps inter-arrivée est indépendant et aléatoirement distribué. Les autres variables aléatoires sont le temps de service Ts, quelque fois appelé la tâche, dont la fonction de distribution est identifiée par $B(x)$ exprimé ainsi :

$$B(x) = P(Ts < x) \tag{3.2}$$

$A(t)$ et $B(x)$ sont présumés des variables aléatoires indépendantes.

La structure de services nous indique le nombre de serveurs et la capacité de ceux-ci, c'est-à-dire le volume de clients en traitement et le volume de clients en attente. La discipline de service détermine la règle de sélection du prochain client. Les lois les plus souvent rencontrées sont :

— Premier arrivé premier servi (*FIFO : First In First Out*) ;
— Dernier arrivé premier servi (*LIFO : Last In First Out*) ;
— Service Aléatoire (*RS : Random service*) ;
— Par Priorité (*Priority*).

La motivation derrière la théorie des files d'attente est de trouver et connaître les mesures de performance du système, premièrement les propriétés (fonction de distribution, fonction de densité, moyenne, écart) en fonction des variables aléatoires : nombre de clients dans le système et en attente, utilisation, temps réponse, attente d'un client, temps mort système et temps d'occupation système. Bien sûr, les réponses ont un lien de dépendance fort avec les hypothèses qui concernent la distribution du temps inter-arrivée, le temps de service, le nombre de serveurs, la capacité et la discipline de service. Pour résoudre des problèmes

pratiques, la première étape est de bien identifier le cas et le type de file d'attente et de calculer les mesures de performance propres à celui choisi. Il est entendu que le niveau de modélisation est lié aux hypothèses. Logiquement, il est recommandé de débuter avec un système simple et si les résultats ne satisfont pas le problème, dès lors il est possible de continuer avec un système plus complexe. Et par conséquent, plus couteux et plus difficile à interpréter.

3.7.1 Notation de Kendall

On peut aussi utiliser un modèle simplifié pour lequel les mesures s'expriment par des équations analytiques. Le modèle de base en files d'attente se nomme *M/M/1* et se généralise en notation de Kendall $A/B/m/K/n/D$:

- A : processus d'arrivée ;
- B : processus de service ;
- m : nombre de serveurs ;
- K : capacité du système ;
- n : taille de la population des clients ;
- D : discipline de service.

3.7.2 Modèle *M/M/1*

Le modèle de base en files d'attente se nomme *M/M/1*. M est pour Markov et est souvent une distribution exponentielle. Aussi, une file *M/M/1* est servie par un seul serveur. Les clients se présentent au système aléatoirement selon une loi exponentielle de taux λ. Le temps de service suit une loi exponentielle de taux λ et μ pour moyenne $\frac{1}{\lambda}$, indépendamment d'un client à l'autre.

L'arrivée et le départ sont tous deux considérés comme des critères importants

dans un système de file d'attente. Il est aussi naturel de dire qu'il existe une relation d'intimité entre l'arrivée et le départ. Nous pouvons la formaliser comme ceci :

Temps pour qu'une nouvelle arrivée se produise :

$$A \sim Exp(\lambda) \tag{3.3}$$

Temps pour qu'un nouveau départ se produise :

$$D \sim Exp(\mu) \tag{3.4}$$

Probabilité qu'une arrivée se produise avant un départ :

$$P(A < D) = \frac{\lambda}{\lambda + \mu} \tag{3.5}$$

$$P(D < A) = \frac{\mu}{\lambda + \mu} \tag{3.6}$$

Il est difficile d'étudier la variable aléatoire $N(t)$ représentant le nombre de clients au temps t dans le système. On s'intéresse plutôt à $N = \lim_{t \to \infty} N(t)$. On parle alors d'analyse en régime stationnaire (ou analyse à l'équilibre). Pour qu'une file *M/M/1* puisse atteindre l'équilibre, il faut que $\lambda < \mu$ sinon la taille de la file augmentera à l'infini. À l'équilibre, on peut montrer que :

$$P(N = n) = \lambda \frac{\lambda}{\lambda + \mu} P(N = n - 1) + \frac{\mu}{\lambda + \mu} P(N = n + 1) \tag{3.7}$$

Il s'agit de la règle des probabilités totales. Le terme $\frac{\lambda}{\lambda+\mu}$ représente la probabilité qu'un nouveau client arrive avant que le client en service quitte le système, et $\frac{\mu}{\lambda+\mu}$ est la probabilité que le client en service quitte avant qu'un nouveau client n'arrive.

3.7.3 Loi de Little

D'abord les Notions :

- $\overline{N}_Q$: nombre moyen de clients faisant la queue ;
- $\overline{N}_S$: nombre moyen de clients en train d'être servis ;
- $\overline{N} = E(N) = \overline{N}_Q + \overline{N}_S$ nombre total moyen de clients dans le système en équilibre ;
- $P(N = k) = \pi_k$;
- N_Q, N_S, N sont les variables aléatoires correspondantes ;
- $\overline{T}_Q$: temps moyen d'attente ;
- $\overline{T}_S$: temps moyen de service ;
- $\overline{T} = \overline{T}_Q + \overline{T}_S$: temps moyen qu'un client passe dans le système ;
- T_Q, T_S, T sont les variables aléatoires correspondantes.

La loi s'énonce ainsi :

$$\overline{N} = \lambda_e \overline{T} \overline{N}_Q = \lambda_e \overline{T}_Q \tag{3.8}$$

$$T = T + \frac{1}{\mu} \tag{3.9}$$

ou λ_e est le taux d'entrée dans le système ($\lambda_e = \lambda$ pour une file $M/M/1$). Puisque $\overline{N} = \overline{N}_Q + \overline{N}_S$ *et* $\overline{T} = \overline{T}_Q + \overline{T}_S$, on trouve également que $\overline{N}_Q = \lambda_e \overline{T}_Q$ *et* $\overline{N}_S = \lambda_e \overline{T}_S$. De plus, la loi de Little s'applique à tous les modèles de file d'attente rencontrés en pratique, pas seulement à la file M/M/1.

Introduisons la notion de facteur d'utilisation : $p = \frac{\lambda}{\mu}$. p représente en quelque sorte la proportion du temps que le serveur est occupé. il s'ensuit que $P_n = (1 - p)p^n$ et que $n = 1, 2, 3, \ldots$ P_n est probabilité qu'il y ait n client dans le système.

Calculons maintenant les caractéristiques de la file d'attente *M/M/1* :

A) le nombre moyen de clients dans le système :

$$\begin{aligned} N &= \sum_{n=0}^{\infty} nP_n \\ &= \sum_{n=0}^{\infty} n(1-p)p^n \\ &= (1-p)p\sum_{n=0}^{\infty} np^{n-1} = (1-p)p\sum_{n=0}^{\infty}\frac{d}{dp}(p)^n \\ &= (1-p)p\frac{d}{dp}\left(\sum_{n=0}^{\infty} p^n\right) \\ &= (1-p)p\frac{d}{dp}\left(\frac{1}{1-p}\right) = (1-p)p\frac{1}{(1-p)^2} \\ &= \frac{p}{1-p} = \frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}} = \frac{\lambda}{\mu-\lambda} \end{aligned} \tag{3.10}$$

B) Nombre moyen de clients dans la file d'attente

$$\begin{aligned} N_Q &= \sum_{n=1}^{\infty}(1-n)P_n \\ &= \sum_{n=0}^{\infty} nP^n - \sum_{n=1}^{\infty} P^n \\ &= N - (1-P_0) \\ &= \frac{\lambda}{\mu-\lambda} - \frac{\lambda}{\mu} \\ &= \frac{\lambda^2}{\mu(\mu-\lambda)} \end{aligned} \tag{3.11}$$

C) Temps moyen pour un client dans le système

$$\begin{aligned} T = \frac{N}{\overline{\lambda}} = \frac{N}{\lambda} = \frac{\lambda}{\mu-\lambda}\frac{1}{\lambda} = \frac{1}{\mu-\lambda} \\ \left(puisque\ \overline{\lambda} = \sum_{n=0}^{\infty}\lambda_n P_n = \lambda\sum_{n=0}^{\infty} P_n = \lambda\right) \end{aligned} \tag{3.12}$$

D) Temps pour un client dans la file d'attente

$$T_Q = \frac{N_Q}{\overline{\lambda}} = \frac{N_Q}{\lambda} = \frac{\lambda^2}{\mu(\mu-\lambda)}\frac{1}{\lambda} = \frac{\lambda}{\mu(\mu-\lambda)} \tag{3.13}$$

3.8 Préambule à la planification et l'ordonnancement

La planification et l'ordonnancement sont des domaines importants de l'intelligence artificielle (IA) et de la recheche opérationnelle (RO). Beaucoup de problèmes réels sont connus sous le nom de problèmes de planification et de l'ordonnancement de l'IA et RO, où les ressources doivent être allouées afin d'optimiser les objectifs à atteindre. Par conséquent, la résolution de ces problèmes nécessite un mélange adéquat de planification, d'ordonnancement et d'allocation de ressources aux activités. Activités qui souvent ont des buts concurrents au fil du temps et qui présentent des contraintes complexes dépendantes de l'état. La satisfaction de la contrainte joue également un rôle important pour résoudre les problèmes de la vie réelle, de sorte que les techniques intégrées qui gèrent la planification et l'ordonnancement avec la satisfaction des contraintes restent nécessaires (Baki, 2006).

Ce type de projet se déroule en quatre étapes :

1. La planification : qui vise à déterminer les différentes opérations à réaliser et les moyens matériels et humains à y affecter.
2. L'ordonnancement : qui vise à déterminer les différentes dates correspondant aux activités.
3. L'exécution : qui consiste à la mise en œuvre des différentes opérations définies dans la phase d'ordonnancement.
4. Le contrôle : qui consiste à superviser l'exécution et voir si celle-ci respecte les prévisions.

La planification, un élément indispensable dans l'élaboration de systèmes intelligents, se définit en termes d'un domaine de planification et de problèmes à résoudre. Un domaine de planification est décrit par un ensemble d'actions qui vont

permettre des transitions entre les états (Baki, 2006).

L'ordonnancement consiste à organiser dans le temps un ensemble d'activités de façon à satisfaire un ensemble de contraintes et optimiser le résultat. Les techniques d'ordonnancement ont pour objectif de répondre aux besoins exprimés par un client, au meilleur coût et dans les meilleurs délais, ceci en tenant compte des différentes contraintes (Baki, 2006).

3.9 Planification temporelle

Jusqu'à maintenant, la plupart des recherches dans ce domaine se sont concentrées sur les difficultés techniques à résoudre lorsqu'on veut introduire des actions duratives dans les problèmes de planification classiques. Parmi ces difficultés, il y a la résolution de problèmes temporellement expressifs c'est-à-dire de problèmes qui ne peuvent être résolus qu'en utilisant des actions concurrentes (Cushing *et al.*, 2007). Un exemple typique de ce type de problèmes est celui qui se pose dans le domaine de la préparation culinaire où plusieurs ingrédients doivent être cuisinés simultanément pour être ensuite être utilisés au même moment. À une plus grande échelle, des problèmes temporellement expressifs sont ceux posés par la gestion d'aéroports ou de gares ferroviaires. Seuls certains planificateurs temporels sont capables de traiter ce type de problèmes (Cooper *et al.*, 2010).

3.10 L'ordonnancement

Historiquement, les problèmes d'ordonnancement ont été initialement abordés dans le domaine de la recherche opérationnelle (Esquirol et Lopez, 1999; Lopez et Roubellat, 2001) tel que : la théorie des graphes, programmation dynamique, programmation linéaire, méthodes d'optimisation combinatoire. Mais, ceux-ci ont vite montré leurs limites en terme d'expressivité. L'intelligence artificielle s'est

alors penchée sur le problème, renouvelant les techniques employées grâce à une représentation plus riche des connaissances du domaine telles que : problèmes de satisfaction de contraintes, algorithmes de propagation de contraintes, langages de programmation par contraintes. Un problème d'ordonnancement se pose lorsqu'il s'agit d'organiser dans le temps l'exécution d'un ensemble de tâches. De tels problèmes se rencontrent dans différents contextes tels que la gestion de grands projets, la conduite d'ateliers de fabrication, l'organisation d'activités de service.

La résolution d'un problème d'ordonnancement consiste à placer dans le temps des activités ou tâches, en tenant compte de contraintes temporelles telles que : délais, contraintes de préséance et contraintes portant sur l'utilisation et la disponibilité des ressources nécessaires par les tâches. Un ordonnancement décrit l'exécution des tâches en précisant : dates de début, dates de fin et l'allocation des ressources au cours du temps, et vise à satisfaire un ou plusieurs objectifs.

3.11 Ordonnancement complexe en temps réel

Par ordonnancement en temps réel, nous voulons dire que le plan est toujours un reflet de l'environnement qui change. Prenons deux exemples : (1) Nous avons une nouvelle tâche qui est plus importante, urgente que des tâches déjà présentes. Nous devrons donc refaire le plan pour que celle-ci soit prise en charge immédiatement. (2) Nous avons un problème avec l'exécution de la tâche, elle n'est pas disponible ce qui cause un délai. En d'autres mots, un temps mort pour la ressource qui devait l'exécuter. Nous devons replanifier pour combler le trou laissé par la tâche disparue [4].

Il y a deux types de contraintes qui sont présentes dans les problèmes d'ordon-

4. http://www.groupes.polymtl.ca/inf2610/documentation/notes/

nancement. (a) Les contraintes dures, celles qui doivent être satisfaites pour que le plan soit considéré légal ; (b) Les molles, elles sont essentiellement des préférences. Idéalement il serait souhaitable que celles-ci soient satisfaites. Un exemple de contrainte dure pourrait être que la tâche soit exécutée uniquement par une ressource en particulier. Un exemple de contrainte molle serait que la tâche pourrait être prise en compte dans une région, mais pas limitée à celle-ci. Les contraintes dures sont essentiellement celles qui vont limiter l'espace de recherche. Tandis que la contrainte molle aide pour la définition et l'évaluation de la fonction de coût.

3.12 Algorithme génétique

Nous aurions pu aborder le problème en utilisant des algorithmes génétiques multi-objectifs, ce qui confirme une fois de plus que cette classe de problème est scientifiquement bien fournie et qu'un intérêt soutenu y est dédié. Il va de soi que les algorithmes génétiques sont fort intéressants, mais ils posent problème lorsque nous sommes dans un environnement dynamique et temps réel (Kachitvichyanukul, 2012).

Ce type d'algorithme se prête pour générer, muter une population en lot. Donc il y a un début et une fin. Dans le détail, l'algorithme veut générer des plans. En d'autres mots, la solution est un ensemble de plans. La fabrication de la solution est produite à partir d'une accumulation de données qui seront combinées pour produire, durant le temps alloué, une bonne solution en tenant compte d'un ou plusieurs critères (Nunes *et al.*, 2011). Souvent, la solution est aussi bonne que le temps que nous offrons à l'algorithme pour son exécution.

Nous sommes intéressés par le temps réel et la construction de plans dynamique. Les algorithmes génétiques (*genetic algorithm*) GA sont mieux adaptés pour une accumulation préalable des données. Par la suite, l'algorithme génétique génère

une population pour ainsi obtenir une solution adéquate souvent non soupçonnée. Cette solution requiert une exécution sur une longue période, une durée de temps qui n'est pas courte (Hornby *et al.*, 2006).

3.13 Apprentissage Machine : Machine Learning

Une autre approche très tendance présentement est l'apprentissage machine supervisé (ML) (*Machine Learning*). Cette technique pourrait être très intéressante éventuellement pour valider ou obtenir un deuxième avis (solution machine). Si nous avions des données, elles devraient être en nombre suffisant et les assignations passées devraient s'y trouver. Avec des données valides nous pourrions par la suite créer un modèle qui pourrait répondre aux demandes des patients. Il serait même possible de croire que le modèle pourrait répondre en temps réel aux demandes. Le défi serait évidemment de maintenir un modèle à jour. Assez actuel pour qu'il soit performant.

Pour être en mesure d'utiliser ce type de technique, nous devons au préalable avoir des données. Ce que nous n'avons pas. Nous devons donc ignorer cette méthode.

CHAPITRE IV

SYSTÈME PROPOSÉ

Tout d'abord, il est important de mentionner que l'approche est assez ambitieuse, car elle doit respecter le fait que nous fonctionnons avec des humains et non pas avec des objets (boîtes). Nous considérons que planifier et ne pas replanifier outre mesure (changer les horaires) est un élément qui a influencé notre implantation et ce mémoire. Il va de soi que replanifier peut permettre de trouver une solution encore meilleure.

Une approche informatique implique que nous ayons recours à des technologies et des outils, mais que l'usage de ceux-ci ne doit pas se faire au détriment des humains qui restent un élément important dans ce mémoire. Offrir une solution qui repousse le problème à plus tard n'est pas une solution. Il n'est pas question d'offrir uniquement de la technologie. Il est vital que cette technologie soit au service des humains. Il s'agit de réaliser une solution qui répond à la fois aux clients malades et aux médecins qui traitent. Ceci en respectant les deux pôles, qui ont parfois des objectifs conflictuels et opposés.

Dans cette optique, nous sommes en mesure de présenter un système et ses composantes comme un outil muni de techniques et de concepts innovants qui vont au-delà de la simple implémentation de techniques naïves à actions restreintes.

4.1 Hypothèses

Notre système est centralisé. Notre planificateur analyse, choisit et ordonne un ensemble de clients reliés entre eux par des contraintes de préséance ainsi que des contraintes telles que l'heure, l'âge, le sexe et l'emplacement géographique. Le processus de planification est élaboré en temps réel. Nous nous sommes intéressés à la planification et son exécution dans un environnement incertain et sous contraintes temporelles et de coût. Nous faisons aussi l'hypothèse que les ressources demandées pour le traitement des clients sont en nombre limité et rendues disponibles seulement une fois que l'exécution de la tâche précédente est terminée.

4.1.1 Médecins

Pour simplifier le problème, nous considérons que les médecins sont en nombre fini et constant pour chacune des cliniques. Ce nombre, un pourcentage, tient compte du nombre de médecins, œuvrant dans chacune des régions du Québec (figure 1.2) . Un facteur du total est utilisé lors de nos expérimentations, car toutes les ressources (les médecins) ne travaillent pas toutes en même temps. Les cliniques sont ouvertes selon des données recueillies auprès du Ministère de la Santé du Québec. Les médecins travaillent et sont disponibles de l'ouverture à la fermeture de la clinique. Nous avons pris cette hypothèse au lieu de faire des changements de médecins ou des quarts de travail. Avoir un médecin pendant 10 heures est la même chose (équivalent) que 2 médecins pendant 5 heures. Un nombre égal de ressources est attribué à chacune des cliniques d'une même région selon la formule (1.4).

4.1.2 Traitements

Dans cette version du travail, les tâches (consultations/traitements d'un client) sont atomiques, non préemptives et elles ont toutes le même niveau de priorité. En d'autres mots, les clients sont tous égaux. Nous assumons que les ressources ne peuvent pas disparaître. Les patients ne sont jamais traités plus de 60 secondes en avance sur le temps du rendez-vous établi. Par cette mesure nous simulons que les patients arrivent sur le site au moment de leur rendez-vous, pas avant. Ceci même si la clinique est disponible et qu'un médecin est disponible. Ce qui veut dire que la clinique accumule du temps mort dans l'attente du prochain patient. Les exemples de temps morts (histogramme jaune) peuvent être consultés sur la figure (1.11).

Quelques autres hypothèses de travail :

- aucune plage de temps n'est préalablement établie dans les cliniques ;
- aucun horaire n'est établi pour les médecins ;
- aucune limite de patient pour chacune des cliniques.

Nous avons pris ces hypothèses de travail pour explorer un axe opposé aux recherches (Qu *et al.*, 2007; Wang, 1997) qui pré-établissent (fixent) les plages, les horaires et/ou le nombre de patients à voir dans une journée. Nous croyons que pour conserver un système dynamique nous devons être capables de rester flexibles et ouverts aux changements et ainsi gérer efficacement les éléments tels que les annulations, les désistements [1] ou les délais variables de traitement.

Le système se veut une initiative différente si on le compare à d'autres systèmes dans le domaine de la santé. Les systèmes de gestion de la clientèle malade ou logiciel de gestion du dossier client *Electronic Medical Record* (EMR), qui inclus

1. les effets des désistements *(no-shows)* (Moore *et al.*, 2001)

parfois une gestion d'horaire locale, sont souvent rigides et laissent peu de place à l'innovation ou la collaboration inter-cliniques.

4.1.3 Centralisé

Donc, un client ne peut pas demander ou faire de multiples réservations *client doublebooking, overbooking* et il n'y aura pas de périodes laissées ouvertes *open slots* dans notre simulateur. Ces mesures avaient été introduites dans les recherches de (Rising *et al.*, 1973; Klassen et Rohleder, 2004; White et Pike, 1964) pour tenter de contrer les problèmes liés à l'absentéisme. Par cette mesure (zéro sur-réservation), nous pourrons obtenir de meilleurs résultats afin de contrer l'absentéisme qui, selon nous et les recherches de (Cayirli et Veral, 2003; Hassin et Mendel, 2008; Chakraborty *et al.*, 2010), est causé en partie par la sur-réservation ce qui prolonge les heures d'ouverture. Les recherches de (LaGanga et Lawrence, 2007b,a) utilisent la surallocation outre mesure. Dans ce cas-ci la surallocation est causée par la clinique pour tenter d'augmenter la productivité des cliniques. Est-il possible que si nous affectons le même client à deux ressources, que l'une ou l'autre des ressources ne verra certainement pas le client ? Ce qui créera de toute évidence une absence pour l'autre.

Un avantage découlant du système centralisé est que chaque patient ne pourrait pas être dans plusieurs files d'attente. Pour les raisons décrites précédemment, nous avons choisi de construire un système central qui atténue les dommages liés aux désistements. À notre connaissance, il n'existe pas de système qui fait ce genre de choix. Les demandes des clients seront donc traitées par un système central et assignées à une et une seule file d'attente clinique à un moment donné.

4.1.4 Désistements

Certaines recherches (Lacy *et al.*, 2004; Lee *et al.*, 2005; Cayirli *et al.*, 2006; Gupta et Denton, 2008) confirment des taux d'absentéisme allant jusqu'à 40% malgré toutes les tentatives pour éviter celui-ci. Dans les recherches de (Johnson *et al.*, 2007), on fait appel à plusieurs mesures telles que : rappels, notification, éducation des nouveaux patients, cadeaux et pénalités ou encore offrir des transports pour réduire les absences. Dans les recherches de (Montecinos *et al.*, 2015), nous tentons de modéliser l'absentéisme et le désistement. Le désistement reste un facteur important (Hassin et Mendel, 2008), difficile à prédire, il faut donc penser autrement.

4.2 Approches

Avant même de présenter notre approche, il est important pour nous de dire que, malgré des années d'étude et plusieurs recherches dans le domaine des files d'attente en santé, très peu de solutions et de recherches sont basées sur l'emploi de la technologie et de techniques informatiques. La plupart des études visent à contourner le problème ou elles visent l'étude d'un angle unique. Par exemple, (Zeng *et al.*, 2010; Daggy *et al.*, 2010; Goldman *et al.*, 1982; Samorani et LaGanga, 2015) se concentrent sur la modélisation et la prédiction des désistements. Nous sommes d'avis que simplifier le problème ou changer l'angle d'attaque est souvent une bonne façon d'aborder ce type de problème, mais nous sommes aussi persuadés que l'introduction de technologies, nouvelles ou anciennes, peuvent être bénéfiques afin de résoudre le problème globalement.

Ce mémoire n'est pas seulement la mise en place de techniques informatiques, il est une réflexion et une prise de conscience, une analyse des recherches et études du passé pour mettre de l'avant non seulement une nouvelle solution, mais une

solution qui tienne compte de ce qui fonctionne, ne fonctionne pas et de ce qui est réaliste et envisageable pour la société d'aujourd'hui.

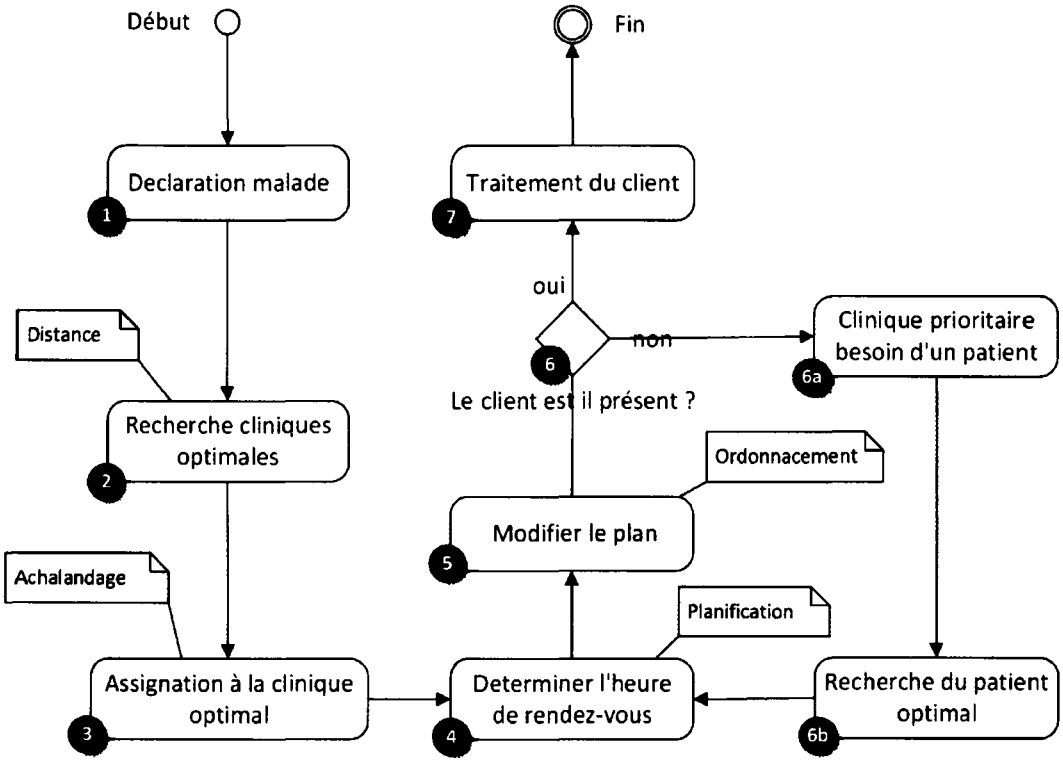


Figure 4.1: Diagramme du système : Étape de prise en charge

La figure 4.1 illustre le système sous la forme d'un diagramme d'activité les étapes de prise en charge d'un client qui serait malade. Pour les étapes 2 et 3, nous avons attaché à l'activité le critère qui influence cette activité soit la distance et achalandage respectivement.

4.2.1 Approche naïve

Nous présentons trois versions afin de résoudre le problème. Notre version initiale, dite naïve à la figure 4.2, ne sera pas comparée avec les deux autres dans les évaluations. Cette version naïve est fonctionnelle, mais d'entrée de jeu nous la

considérons limitée. Afin de bien illustrer le cas, $P1$ arrive avant $P2$ ainsi de suite jusqu'à $P4$. Nous pouvons remarquer que les patients ($P1, P2, P3, P4$) sont assignés à la clinique $C1$ car elle est située plus proche. $P5$ ira dans $C4$ encore une fois située à proximité. Un débalancement des files d'attente, identifié par Q, est créé, ce qui n'est pas souhaitable pour les cliniques.

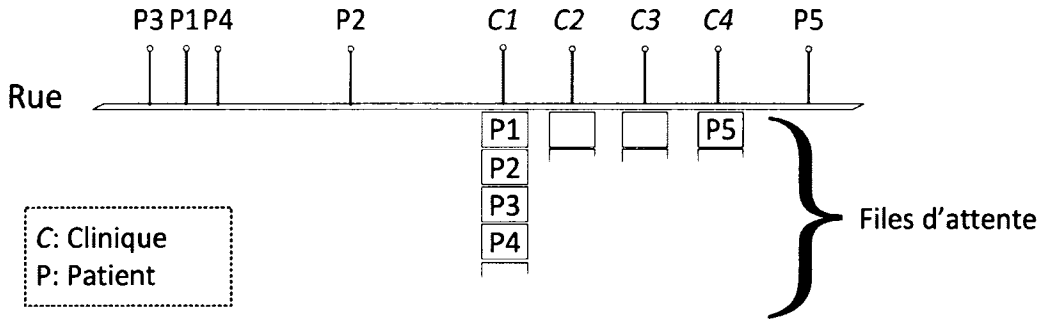


Figure 4.2: Ordonnancement de la version naïve

Dans le détail la version naïve fonctionne comme suit. Lorsque nous trouvons une clinique susceptible de traiter un client, disons que nous allons prendre la plus proche, aussitôt, nous l'assignons à celle-ci. Nous constatons un problème avec cette démarche. Certaines cliniques sont beaucoup plus sollicitées. Leur situation géographique est favorable. Donc trop de clients pour certaines et les autres en ont moins ou pas. Il devient impossible de traiter tout le monde et beaucoup de retard est ainsi créé. Aucune amélioration dans l'immédiat, mais une bonne piste pour la prochaine version (simple) qui va d'abord équilibrer les demandes. Donc assigner des patients à une clinique proche n'est pas souhaitable, car elle ne résout pas le problème, au contraire. Ceci confirme aussi qu'il ne s'agit pas d'un problème simple, exemple trouver la clinique la plus proche en utilisant un arbre-KD. L'arbre-KD est détaillé à la section 4.5. Il faut maintenant considérer quelques critères, car l'utilisation d'un seul critère comme la distance n'est pas suffisant.

4.2.2 Approche simple

Dans la version simple nous allons ajouter un critère. À chaque fois qu'un patient apparaît, nous allons vérifier quelles cliniques peuvent servir notre client afin d'équilibrer l'achalandage. Nous allons donc considérer d'autres critères afin d'améliorer la qualité de l'assignation. Ce critère est la longueur de la file d'attente. Dans la figure 4.3, les patients sont équilibrés dans chacune des files d'attente. Cette version est bonne si toutes les cliniques sont dans un rayon proche. Ceci serait applicable pour une grande ville dense. Il n'est pas réaliste d'assigner bêtement un patient à une clinique, même si la file d'attente est courte, car celle-ci pourrait être trop éloignée. Pour le territoire du Québec, nous devons l'améliorer car cette version est insuffisante. Sachant cela, est-ce réaliste de vérifier toutes les longueurs des files d'attente ceci pour chaque patient ? On parle ici de recherche optimal. Un coût asymptotique $O(p \times c)$ est alors nécessaire. Tout vérifier pourrait être long et onéreux en temps processeur.

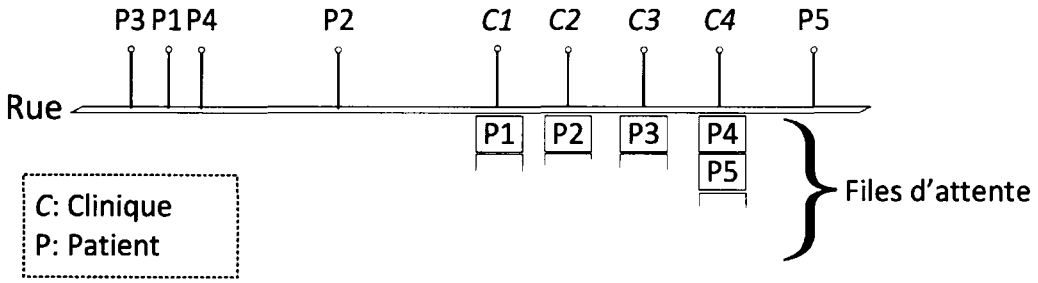


Figure 4.3: Ordonnancement de la version simple

4.2.3 Approche avancée

Pour la suite des choses, supposons que p est un patient et c est une clinique. Trouver quelle est la clinique la plus proche et qui est libre, à un coût équivalent

à $O(n)$. Il est cher payé d'agir ainsi seulement pour trouver la prochaine clinique à remplir.

Optimisation 1 : limiter l'espace de recherche

Pour ce faire nous devons réduire l'espace de recherche. À l'aide d'un arbre-KD, détaillé à la section 4.5, nous sommes en mesure de trouver de façon efficace les k cliniques à proximité du patient. Notre coût de recherche est maintenant de $O(log\ n + k)$ pour la recherche des k plus proches cliniques qui sont susceptibles d'être prises par le client. Ce coût est plutôt encourageant.

Il y a tout de même un léger problème avec cette technique lorsque nous avons plusieurs demandes de patients qui proviennent du même endroit (ou géographiquement proche). Limiter l'espace de recherche peut éliminer des solutions qui sont potentiellement bonnes. Pour un quartier avec une population dense, réduire l'espace de recherche concentrera les clients vers un groupe restreint de cliniques, ce qui augmentera la longueur des files d'attente d'un petit nombre de cliniques. Si tous les patients viennent du même point géographique, il est donc important d'élargir un peu, non pas en distance géographique (le diamètre de la zone), mais augmenter le nombre de cliniques proches.

Optimisation 2 : élagage de cliniques

C'est une fois ce sous-ensemble calculé que nous recherchons la clinique qui a le coût le plus bas. Toutes les cliniques du sous-ensemble avec $Q.size \leq \overline{Q.size}$ seront conservées. $\overline{Q.size}$ est la moyenne de toutes les files d'attente dans le sous-ensemble. Sur ce nouveau sous-ensemble, nous appliquons deux contraintes additionnelles, soit le temps de déplacement et le temps approximatif du début du traitement. Nous ajoutons ensuite une fonction pour déterminer celle qui est à

même de mieux servir. La clinique qui a le plus haut taux de succès. En d'autres mots, celle qui respecte ses engagements (stable). Nous sommes ainsi capables de traiter toutes les demandes en temps réel.

Lors de nos tests nous avons remarqué que certaines cliniques traitent les clients plus lentement. Nous avons initié une réflexion qui voulait prévenir l'accumulation de patient et ce avant même le calcul de la fonction à minimiser. Nous réduisons ainsi de façon importante le nombre de calcul et augmentons aussi le rayon d'action de l'arbre-KD. Cette technique permet un traitement équitable des demandes tant pour le client que pour les cliniques qui ont un service optimal.

Optimisation 3 : traitement par lot

Prenons par exemple le cas de la figure 4.3, il est montré que, lorsque nous planifions en temps réel et en fonction de l'ordre dans lequel se présentent les clients, il est possible de mal assigner les clients aux cliniques. Il est préférable d'ajouter les clients à une file temporaire, un accumulateur de tâches à distribuer. Ceci aide grandement comparé au placement en temps réel unitaire. La création du lot à planifier est de taille variable et est construit pendant une durée de temps fixe, exemple 60 secondes. La figure 4.4 nous illustre le résultat de la planification lorsque nous avons le mode accumulateur actif.

Nous avons raffiné le traitement par lot pour le rendre plus flexible. Cette modification concerne un paramètre additionnel fourni à l'algorithme (voir Algorithme 1, p.65). Il s'agit d'une borne supérieure qui limite, à la (ligne 7), le nombre de clients que LP peut contenir. L'accumulation s'arrête au premier critère satisfait soit : x secondes ou y clients dans l'accumulateur.

La version finale de notre travail est en temps réel avec accumulation dans un lot avant d'initier la planification. Dans l'algorithme 1, nous avons trois appels

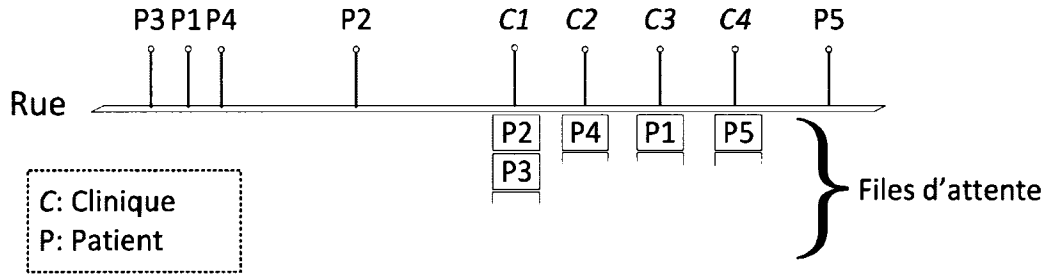


Figure 4.4: Ordonnancement de la version avancée

de fonctions, dites principales. Celles-ci sont : la gestion des abandons (ligne 10) ; la planification qui inclus l'assignation (ligne 11) ; et finalement, l'exécution du service (les traitements patients) (ligne 12). La variable FIN est liée au bouton fermer dans l'application.

Algorithme 1 : Algorithme principal

```
1  Function main()
2      LC ← chargement de la liste de cliniques;
3      D ← durée de l'accumulation;
4      n ← nombre de patients ou +∞;
5      KDCliniques ← LC;
6      while (not FIN) do
7          while (tsim ≤ tsim + D) et (LP.size() ≤ n) do
8              ajouteClient(LP) ;          // LP est la liste de patients
9          KDClients ← construireKD(LP);
10         gestionAbandons(KDClients, LC);
11         planification(LP, LC, KDCliniques);
12         executionTraitement(LC);
```

4.3 Remplacement des désistements

Puisque nous sommes maintenant en mesure de traiter toutes les demandes en temps réel, nous allons présenter notre approche pour le remplacement des désistements en mode séquentiel. Le principe est similaire, mais innovant dans le sens que nous créons un arbre-KD à partir des clients accumulés dans l'intervalle de création de lot. À chaque fois qu'une clinique détecte un abandon, elle devient prioritaire et une propriété remplacement est mise à VRAI. Au prochain cycle de replacement des désistements, l'algorithme 2 sera lancé. L'algorithme 2 aurait aussi pu être dans un fils d'exécution séparé (*thread*). Ceci aurait permis un remplacement encore plus rapide, moins de temps mort, mais aurait aussi complexifié le processus et nécessité le développement d'une structure de données capable de gérer la concurrence efficacement. Lors du cycle de remplacement des abandons, l'algorithme 3 est lancé afin de trouver le meilleur client, celui qui a le coût le plus faible.

Prenons les variables suivantes :

- $tsim$: temps de la simulation (moment présent) ;
- trv_c : temps du rendez-vous disponible ;
- td_x : temps de déplacement ;
- $Cout_x$: Coût de la présence, si le client sera présent avant le rendez-vous disponible sinon $+\infty$;
- K : une constante de temps à ajouter (exemples 5 minutes).

Algorithme 2 : Remplacement des désistements

```
1 Function gestionAbandons(KDClient, Cliniques)
2     KD ← KDclient;
3     C ← Cliniques;
4     n ← 20 ;              // patients (limiter l'espace de recherche)
5     while c ∈ C do
6         if c.remplacement then
7             g ← C.Coordonnées;
8             p ← trouverMeilleurClient(KD, g, n);
9             C[c.position du replacement] ← p;
```

Algorithme 3 : Trouver le meilleur client au coût minimal

```
1 Function trouverMeilleurClient(KD, g, n)
      Output : p
2     X ← KD.nearestneighbor(n, g);
3     Cout_x ← +∞;
4     p ← NULL;
5     while x ∈ X do
6         if (tsim + td_x + K ≤ trv_c) then
7             Cout_x = tsim + td_x + K;
8         p ← minimize(Cout_p, Cout_x, x);
9     return p
```

4.4 Formalisation de l'approche

Nous allons formaliser notre approche. Prenons, $p(c, Q, G, H) : c \rightarrow \{q, g, h\}$, est une fonction d'association (*mapping*), qui a un client c l'assigne une file q appar-

tenant à l'espace de toutes les files Q (cliniques), à un emplacement géographique g appartenant à l'espace des coordonnées géographiques (latitude, longitude) et à une plage horaire h issue de l'espace de toutes les plages horaires H. De plus, il existe une correspondance entre Q et G via l'opérateur dual $'$. C'est-à-dire que $\forall q \in Q, \exists q' \in G$ et inversement que $\forall g \in G, \exists g' \in Q$. Toutefois, il faut prendre en compte que q' est toujours un singleton alors que g' ne l'est pas toujours. En effet, plusieurs cliniques peuvent coexister sur un même point. Exemple : sur différents étages d'un même immeuble.

Ainsi, $\Pi(C, Q, G, H) : \{c_1, \ldots, c_{|C|}\} \rightarrow \{\{q_1, q_1', h_1\}, \ldots, \{q_{|C|}, g_{|C|}, h_{|C|}\}\}$, pourrait représenter la fonction qui assigne tous les clients à toutes les cliniques et les plages horaires.

À partir de là, la solution de référence (KD) peut être définie comme suit :

$$\hat{p}(c, Q, G, H) = argmin_{q \in KD.closest(c,n)}(\phi(c, q)) \tag{4.1}$$

ϕ est une fonction qui, à chaque clinique, donne un score de pertinence (temps de déplacement + délai minimal d'attente dans clinique en secondes) et n le nombre de cliniques les plus proches retournées par l'arbre-KD. La clinique qui minimise $\phi(q|c)$ sera sélectionnée pour c.

$$\Pi(C, Q, G, H) = argmin_{pl \in \Omega_\Pi}(\sum_{c \in pl.C} p(c, Q, G, H)) \tag{4.2}$$

L'équation (4.2) serait la combinaison linéaire à optimiser. Autrement dit, la planification globale optimale serait l'ensemble des planifications optimales (en termes de temps d'attente et de distance) pour chacun des clients. Nous représentons l'espace de toutes les planifications possibles par Ω_Π.

La seconde heuristique pourrait être formulée à partir de (4.2) mais avec la

contrainte supplémentaire suivante :

$$\forall pl \in \Omega_{\Pi}, \sum_{g \in G} |min(\hat{Q}) - avg(\hat{Q})| = 0 \tag{4.3}$$

En d'autres termes, pour chaque emplacement géographique parmi ceux choisis, la planification globale doit répartir au maximum la charge de clients sur $\hat{Q}$, les cliniques aux alentours. Les cliniques aux alentours étant renseignées par l'arbre-KD. Dans la pratique, satisfaire une telle contrainte s'avère particulièrement difficile en tout temps. Elle ne sera donc que partiellement satisfaite. Ici, son rôle principal est de forcer la distribution de la charge entre les cliniques. Elle nous permet d'éliminer les solutions qui ne distribuent que peu ou pas du tout la charge.

La troisième heuristique :

$$\Pi(C, Q, G, H) = argmin_{pl \in \Omega_{\Pi}}(\sum_{w_i \in pl.W} \left[\sum_{c \in w_i} p(c, Q, G, H) \right]) \tag{4.4}$$

Toujours avec la contrainte (4.3). W représente l'accumulateur. Un client c n'appartient qu'à un seul accumulateur.

4.5 Arbre-KD

> L'arbre-KD, *k-dimensional tree*, est un cas particulier des *Binary Space Partitionning (BSP) trees*, qui est un partitionnement spatial de l'espace à k-dimensions permettant de structurer les données en fonction de leur position dans l'espace (Fleury, 2008).

«Le rôle de l'arbre-KD est double : il permet, d'une part, d'avoir une subdivision spatiale optimisée de l'espace permettant d'accélérer le traitement des données et,

d'autre part, de stocker les données sous la forme d'un arbre binaire» (Fleury, 2008).

Nous pouvons voir, dans la portion supérieur de la figure 4.5, que nous avons la représentation géographique des points, cliniques et patients. Nous avons repris la figure 4.4 pour illustrer comment l'arbre-KD fonctionne avec un exemple de ce mémoire.

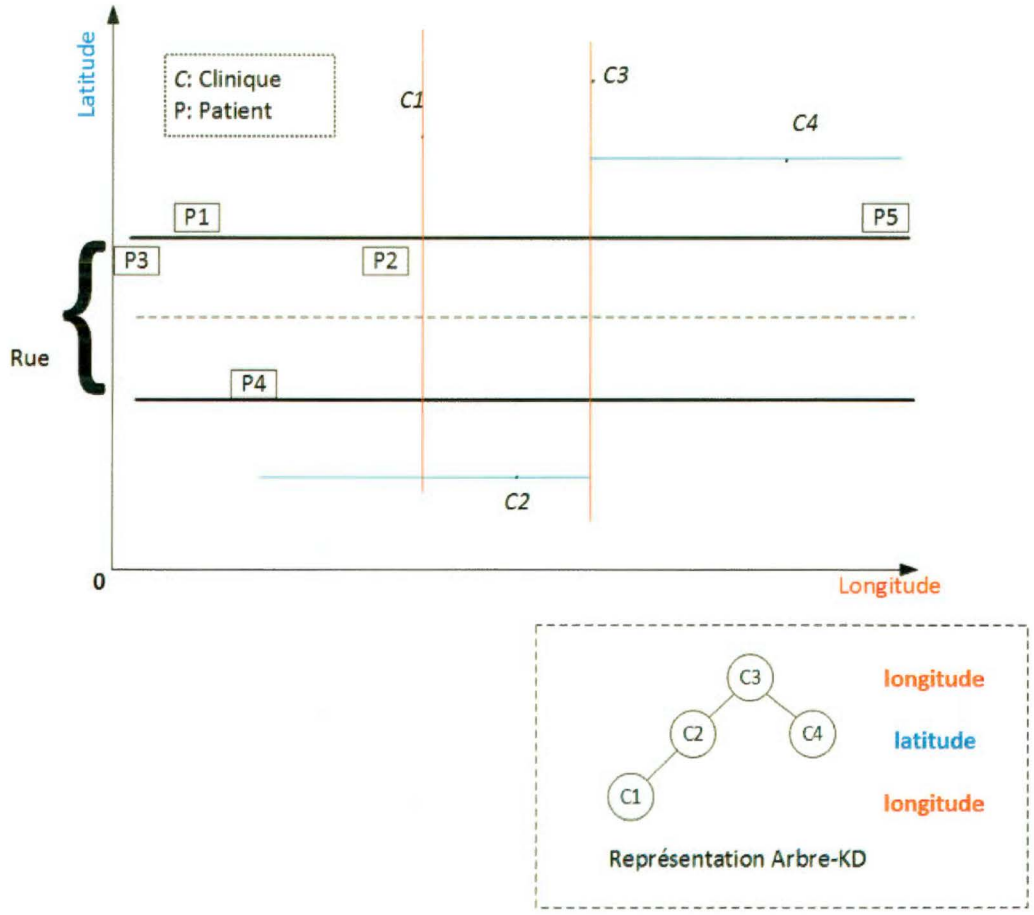


Figure 4.5: Représentation géographique et l'arbre-kd

Pour les besoins de nos travaux, nous avons utilisé un arbre-KD sur deux dimensions (latitude, longitude). Nous pouvons à l'aide de cette structure de données créer, en mémoire, un arbre-KD qui contient toutes les cliniques. Il est

ainsi possible par la suite d'interroger la structure avec une fonction telle que $LC \leftarrow nearestneighbor(n)$. La fonction nous retournera les n plus proches cliniques à proximité du client demandeur. Naïvement, il aurait fallu maintenir dans une liste chainée la liste des cliniques et itérer sur la liste de clinique LC. Pour chacune nous devons calculer la distance et vérifier si elle est dans les n plus proches. Nous avons un coût moyen de $O(n)$ pour la version naïve comparativement à $O(log\ n)$ pour l'arbre-KD.

Simplement, l'arbre-KD nous permet d'indexer puis de récupérer efficacement un ensemble (restreint) des cliniques les plus proches (spatialement) pour un client donné.

CHAPITRE V

ÉVALUATION EMPIRIQUE

Dans ce chapitre, nous allons présenter et discuter de nos résultats. Cette partie est structurée comme suit. Tout d'abord, nous présentons un résumé des hypothèses ainsi que les paramètres utilisés. Par la suite, les différentes simulations sont présentées. Ensuite, nous proposons une synthèse des évolutions de notre système sous la forme d'un ensemble de tables de résultats. En terminant, une discussion comparative des résultats.

Pour la suite des choses, nos expérimentations ont été exécutées sur une machine Surface Pro 3, Windows 8.1, Intel(R) Core(TM) i7-4650U CPU @ 1.70Ghz, RAM 8GB, SSD mSata 256MB.

5.1 Préambule aux résultats

Un rappel sur les variables et les paramètres utilisés lors de nos évaluations :

- Nombre de médecins : 50% des omnipraticiens, aucun spécialiste ;
- Heures de traitement : 9h00 @ 22h00 ;
- Heures d'assignation possible : 9h00 @ 21h00 ;
- Acceptation des demandes d'assignation : en continu, toujours ;
- Intervention humaine : aucune ;

— Limite de traitement par jour : aucune ;
— Nombre de demandes par jour : $75\,000 \pm 15\%$;
— Action prise pour les clients non traités dans la journée : aucune. Ils seront traités le lendemain avec un retard (la nuit) ;
— Nombre de désistements : 15% des demandes.

5.2 Paramètres pour évaluation empirique

Afin de bien évaluer le système et trouver les paramètres qui sont adéquats, nous avons procédé par évaluation empirique. Cette technique vise à exécuter plusieurs fois le simulateur avec différentes valeurs afin d'en arriver à un compromis raisonnable et conclure. Nos principaux paramètres d'exécutions sont : le temps de l'accumulation $(1, 30, 60)$ secondes et le nombre de patients $(1, 100, +\infty)$.

Certaines évaluations ne seront pas discutées, car certaines donnent de très mauvais résultats. Un exemple de ceci : assigner un patient unique dès son apparition n'est pas souhaitable (version naïve). Une seconde (1s) vient à dire sans accumulation et sans l'accumulation nous ne pouvons pas faire de remplacement des désistements.

5.3 Les simulations

Les tableaux qui suivent représentent plusieurs expérimentations. Le simulateur débute toujours à 00h00, il n'y a aucun client préchargé au temps 00h00 sur la carte, et toutes les files d'attente cliniques sont vides. La simulation se termine lorsque le système (simulation) est manuellement terminé. Les actions de la simulation sont décrites par l'algorithme 1 à la page 65.

Voici le sommaire de nos évaluations chronologiquement exécutées et listées :

— Le tableau 5.1 présente la version avancée **sans** remplacement des désistements avec accumulateur limité à 60 secondes et $+\infty$ clients ;
— Le tableau 5.2 présente la version avancée **avec** remplacement des désistements avec accumulateur limité à 60 secondes et $+\infty$ clients ;
— Le tableau 5.3 présente un sommaire du délai d'attente moyen ainsi que le temps mort **sans** versus **avec** le remplacement des désistements avec accumulateur limité à 60 secondes et $+\infty$ clients ;
— Le tableau 5.4 présente la version avancée **avec** remplacement des désistements avec accumulateur limité à 60 sec ou 100 patients (première occurrence) ;
— Le tableau 5.5 présente la version avancée **avec** remplacement des désistements avec accumulateur limité à 30 sec ou 100 clients (première occurrence) ;
— Le tableau 5.6 présente 3 versions avancées côte à côte, avec des paramètres d'exécution différents.

Nos tableaux présentent les variables suivantes :

— id : identifiant de ligne ;
— tsim : temps dans le simulateur lors de la capture ;
— #Client : nombre de clients en attente dans le réseau ;
— #T : nombre de traitements dans le réseau ;
— #D : nombre de désistements total dans le réseau ;
— #R : nombre de remplacements total dans le réseau.

5.4 Résultats

Dans un premier temps, nous avons simulé la réalité assez fidèlement ce qui permet d'avoir des expérimentations plus que satisfaisantes. Le tableau 5.1, colonne #T, ligne d, présente le nombre total de clients malades quotidiennement. Soit un total de 84 841 demandes en 24 heures. Si le simulateur était utilisé durant une année seulement les jours ouvrables, nous pourrions traiter un peu plus de 22 millions de demandes. En 2007, le nombre de services ambulatoires dispensés, au cabinet, à l'externe et à l'hôpital, étaient un peu plus de 28 millions [1] sur un peu plus de 80 millions de services médicaux dispensés à l'échelle du Québec en une année [2].

5.4.1 Évaluation avec ou sans remplacement des désistements

Tableau 5.1: Évaluation S (avancée) : **sans** remplacement des désistements

id		tsim	#Clients	#T	#D	#R
a	Jour 1 @ 08h55	08h55	26447	0	0	0
b	Jour 1 @ 10h00	10h00	19909	12760	2215	0
c	Jour 1 @ 16h00	16h00	9200	47076	8264	0
d	Jour 2 @ 00h01	24h01	4273	68608	11960	0
e	Jour 2 @ 12h05	36h05	27682	88881	15551	0
f	Jour 2 @ 22h00	46h00	8992	136373	24026	0
g	Jour 3 @ 22h00	70h00	15802	202936	35800	0
h	Jour 4 @ 22h00	94h00	22511	269546	47253	0

Clarification sur les tableaux, il n'est pas possible de faire une simple addition de

1. http ://publications.msss.gouv.qc.ca/msss/fichiers/2009/09-731-01F.pdf

2. https ://g74web.pub.msss.rtss.qc.ca/statistiques.asp

colonnes dans le cas du tableau 5.2 avec remplacement pour obtenir le nombre de demandes, car certaines demandes sont des désistements remplacés. Ils sont inclus dans la colonne #T. De plus, il est possible 15 fois sur 100 qu'un désistement remplacé soit lui-même un désistement. Pour cette raison, nous n'avons pas de colonne grand total dans nos tableaux. Nos résultats montrent qu'il est possible de planifier et assigner toutes les demandes en temps réel. Nos résultats démontrent aussi que nous sommes en mesure de remplacer les désistements de façon efficace. Nous obtenons en moyenne 43% de remplacement de tous les désistements (Tableau 5.2, colonne #R, ligne c).

Tableau 5.2: Évaluation R (avancée) : **avec** remplacement des désistements

id		tsim	#Clients	#T	#D	#R
a	Jour 1 @ 08h55	08h55	26524	0	0	0
b	Jour 1 @ 10h00	10h00	16343	15738	2833	1395
c	Jour 1 @ 16h00	16h00	6959	49049	8671	4035
d	Jour 2 @ 00h01	24h01	3032	69742	12211	5495
e	Jour 2 @ 12h05	36h05	24335	91924	16147	7446
f	Jour 2 @ 22h00	46h00	6556	138854	24335	10916
g	Jour 3 @ 22h00	70h00	11725	206764	36473	16347
h	Jour 4 @ 22h00	94h00	15280	275386	48651	21717

Il est montré, en regardant l'allure générale du diagramme de Gantt figures A.3a et A.4a, qu'il y a très peu de plages laissées vides. On peut voir cela par le nombre de rendez-vous, en bleu, mis côte à côte. On constate aussi qu'il y a quelques superpositions de rendez-vous. Plusieurs facteurs peuvent causer cela :

— les remplacements des désistements ;

— le débit de traitements ;

— le traitement concurrent de patients.

Dans les mêmes figures, nous sommes donc en mesure de constater que le plan est construit en fonction du débit de traitement. L'escalier superposé survient lorsque la file d'attente se vide plus rapidement que prévu. Nous pouvons voir cela par les deux étages bleus similaires à un escalier dans la figure A.2a. La longueur du trait bleu est la durée du traitement soit 15 minutes (fixe) pour des raisons de lisibilité. En début de journée, tous les médecins sont disponibles et un traitement simultané est possible. Le système est donc en mesure d'offrir plus de clients. Par contre s'il y a un trou, figure A.3a entre 11h20 et 11h40, ceci est dû à :

— du retard (délai), le système a décidé de ralentir ;

— un nivellement équitable des demandes dans le réseau ;

— moins de demandes que la capacité de traitement.

Nous pouvons constater que nous avons un résiduel jour après jour dans le tableau 5.1 colonne #Clients aux lignes d et f. Le résiduel est moindre dans la version avec remplacements tableau 5.2 colonne #Clients aux lignes d et f.

5.4.2 Évaluation : Comparaison avec ou sans remplacement

Les résultats montrent que nous sommes en mesure d'assigner des clients à des files d'attente et de remplacer les désistements, ceci très rapidement, mais que devient l'attente ? Les délais d'attente ainsi que les gains, entre la version avancée sans et avec remplacement, sont présentés dans le tableau 5.3.

Après un certain temps d'exécution, la moyenne d'attente $\overline{da}$ dans le réseau[5] est supérieur dans la version sans remplacement par rapport à la version avec

---

3. délai attente moyen

4. temps mort moyen

5. pour l'ensemble de toutes les cliniques

Tableau 5.3: Résultats sommaire : (délai moyen) **sans vs avec** remplacement des désistements

id	tsim	sans		avec		**gain**	
		$\overline{da}$ [3]	$\overline{tm}$ [4]	$\overline{da}$	$\overline{tm}$	$\overline{da}$	$\overline{tm}$
b	Jour 1 10h00	117	0	85	0	1.37	
c	Jour 1 16h00	156	0	136	0	1.14	
d	Jour 2 00h01	84	46	77	47	1.09	0.98
e	Jour 2 12h00	206	46	167	47	1.23	0.98
f	Jour 2 22h00	159	38	91	32	1.74	1.19
g	Jour 3 22h00	263	56	212	69	1.24	0.81
h	Jour 4 22h00	310	126	220	60	1.40	**2.10**

remplacement. Nous pouvons aussi dire que nous avons un très bon gain de temps mort moyen $\overline{tm}$ du côté des cliniques. Soit par un facteur de 2.1 fois : 60 au lieu de 126 minutes.

La version avec remplacement nous présente de façon générale une réduction l'attente moyenne des clients par un taux 24 pour cent.

5.4.3 Évaluation avec remplacement : 3 versions

Les tableaux 5.4 et 5.5 sont deux expérimentations qui ont suivi celles du tableau 5.2. Puisque la version **avec** remplacement des désistements était plus performante, nous nous sommes concentrés sur celle-ci.

Le tableau 5.4 fait la planification au premier événement parmi les suivants : soit une accumulation pendant 60 secondes ou 100 clients dans l'accumulateur.

Dans le cas du tableau 5.5 l'évaluation a été conduite avec une accumulation

Tableau 5.4: Évaluation Empirique (60s ou 100 clients) : **avec** remplacement des désistements

id	tsim	#Clients	#T	#D	#R	$\overline{da}$	$\overline{tm}$
a	Jour 1 08h55	26683	0	0	0	0	0
b	Jour 1 10h00	24196	9121	1536	453	128	0
c	Jour 1 16h00	6799	49630	8751	3403	168	1
d	Jour 2 00h01	3175	69760	12191	4821	65	41
e	Jour 2 12h00	24753	91900	16129	6796	184	41
f	Jour 2 22h00	6290	139282	24545	10265	98	26
g	Jour 3 22h00	11502	207127	36573	15461	206	16
h	Jour 4 22h00	15332	275878	48581	20557	221	44

Tableau 5.5: Évaluation Empirique (30s ou 100 clients) : **avec** remplacement des désistements

id	tsim	#Clients	#T	#D	#R	$\overline{da}$	$\overline{tm}$
a	Jour 1 08h55	26729	0	0	0	0	0
b	Jour 1 10h00	17040	15260	2651	1122	96	0
c	Jour 1 16h00	7172	48865	8448	3246	146	0
d	Jour 2 00h01	3017	69723	12046	4456	86	35
e	Jour 2 12h00	24023	98168	17008	6599	234	35
f	Jour 2 22h00	6449	138549	24262	9038	114	44
g	Jour 3 22h00	11953	206009	36254	13489	257	48
h	Jour 4 22h00	16304	274581	48242	17558	230	48

Tableau 5.6: Résultats finaux version avancée avec remplacements : Comparaison en fonction des paramètres d'exécution

id	tsim	60s et c[6]=∞		60s ou c=100		30s ou c=100	
		$\overline{da}$	$\overline{tm}$	$\overline{da}$	$\overline{tm}$	$\overline{da}$	$\overline{tm}$
b	J1 10h00	**85**	0	128	0	96	0
c	J1 16h00	**136**	0	168	1	146	0
d	J2 00h01	77	47	**65**	41	86	**35**
e	J2 12h00	167	47	184	41	234	**35**
f	J2 22h00	91	32	98	$\mathbf{26}^{f2}$	114^{f3}	44
g	J3 22h00	212	69	206	$\mathbf{16}^{g2}$	257^{g3}	48
h	J4 22h00	220	60	221	$\mathbf{44}^{h2}$	230^{h3}	48

pendant 30 secondes ou 100 clients dans la file globale d'attente (l'accumulateur). Le premier événement déclenche la planification.

5.4.4 Évaluation : Comparaison des résultats

Dans la figure 5.6 nous avons identifié en gras les données les plus importantes. Il faut aussi prendre en compte que planifier plus souvent implique un remplacement des désistements plus fréquent.

Pour ce qui est du temps d'attente moyen $\overline{da}$ vécu par le client, deux des trois versions avec remplacement donnent des résultats similaires après 4 jours de simulation. Toutefois, la version 30sec ou 100 clients est moins performante. Ceci s'explique par un nombre trop bas de clients dans la file d'attente globale et qui ne permet pas de planifier de façon efficace. Les mauvais résultats après chaque

6. nombre de clients accumulés durant la période d'accumulation

journée sont identifiés par : ^{f3} ,^{g3} et ^{h3} dans le tableau 5.6.

Nous avons constaté que si nous diminuons le temps entre les planifications, que nous obtenons une baisse du temps mort moyen $\overline{tm}$ clinique de façon intéressante. Dès que nous le diminuons trop, nous avons l'effet inverse. Dans le tableau 5.6, nous avons respectivement à la ligne h : 60, 44 et 48 minutes. Intéressant, le temps mort passe de 60 à 44 minutes. Un gain fort appréciable de 25%. Ceci, car le nombre de clients=100 est atteint avant 60 secondes. Donc une légère augmentation du nombre de planifications conduit à une diminution du temps mort moyen. Par contre planifier plus rapidement et plus souvent ne permet aucune amélioration du temps mort.

Grâce aux résultats obtenus de façon empirique, nous pouvons conclure, tant pour le temps mort moyen que pour le délai d'attente moyen, que la version avec remplacement des désistements avec accumulateur 60 sec ou 100 patients est plus performante que ces deux concurrents.

Petit rappel sur la section 4.1, notre simulateur (la simulation) agit en fonction d'hypothèses telles que :

1. le nombre de cliniques ;
2. le nombre médecins.

Avec nos résultats, nous constatons :

1. que la demande est supérieure à la capacité de traitement ;
2. qu'une portion du désistement est actuellement remplacée ;
3. que le délai d'attente diminue grâce aux remplacements des désistements ;
4. que le temps mort est minime ;

Dans nos expérimentations et dans la figure A.2b en comparaison avec la figure A.5b, il faut noter que le temps moyen de retard s'améliore. Il devient en sorte

négligeable.

Après 30 jours de simulation de la version simple, nous sommes en mesure de dire que le système a traité plus de 1.86M de patients en plus de subir 328 258 désistements. À ces nombres, nous avons 45 967 clients qui restent à traiter dans les différentes files d'attente. Il s'agit d'une moyenne de 1 532 clients non vus par jour.

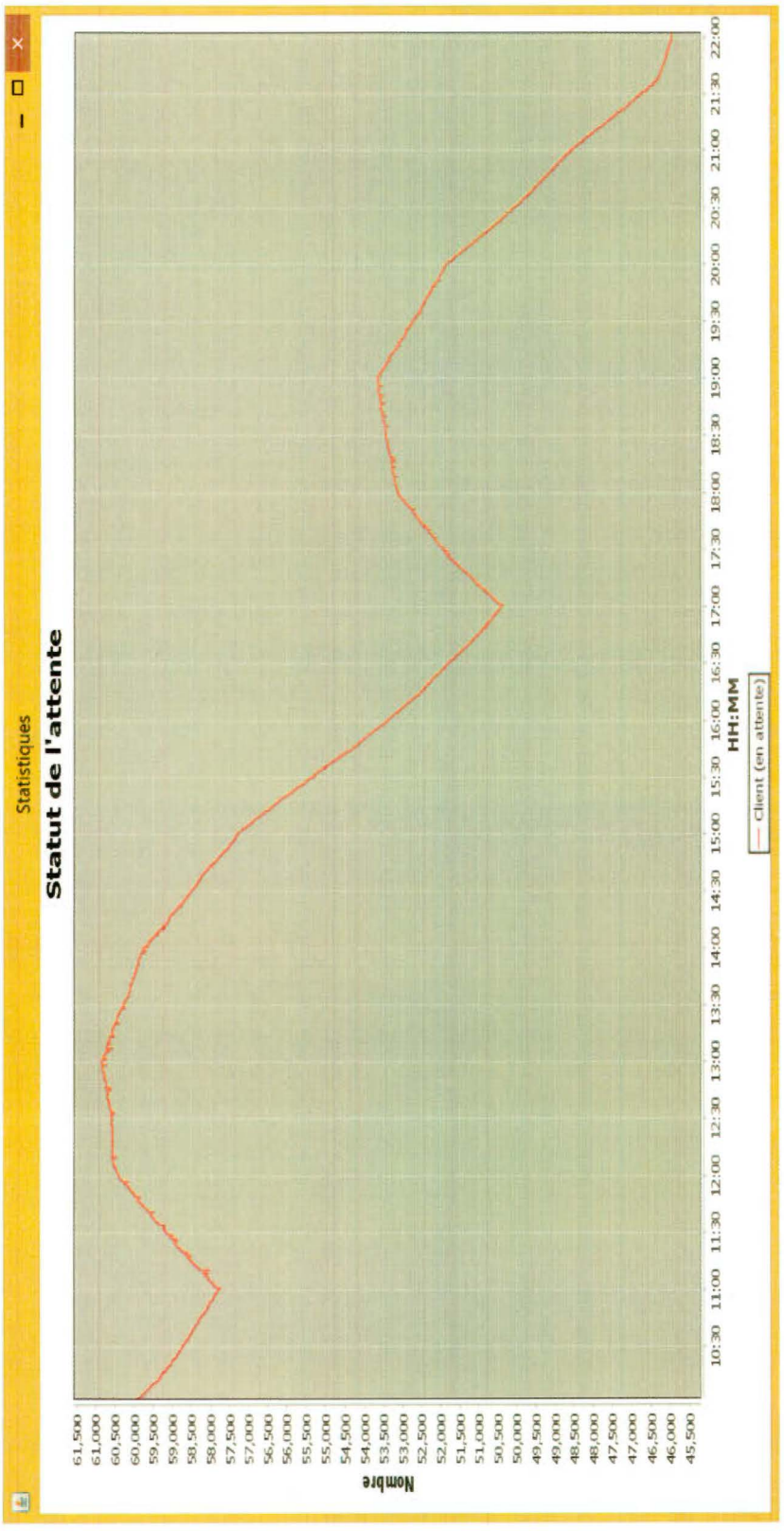


Figure 5.1: Le nombre total de patients dans tout le réseau (après 30 jours)

CONCLUSION

Dans ce mémoire, nous avons présenté un système pouvant améliorer la gestion des files d'attente de cliniques médicales en utilisant l'assignation en temps réel. Ceci en tenant compte de l'incertitude qui régit le monde réel. L'objectif principal est de construire des ordonnancements qui s'adaptent bien à l'environnement réel.

Nous avons vu, dans l'état de l'art, qu'aucune recherche n'a résolu le problème dans son ensemble. Certaines ont empiré la situation tandis que d'autres ont amélioré un aspect unique. Bien que de nombreux chercheurs se soient penchés sur le problème des files d'attente en santé, l'attente est toujours très présente et vivante en 2018. Quelques solutions partielles existent, mais rien qui puisse être considéré comme une solution qui puisse réduire l'attente en santé.

Les applications existantes, dites commerciales, ont très peu d'éléments de la science inclus dans leurs solutions. Ce qui est certainement une des raisons qui expliquent que nous ayons encore une situation déplorable dans le réseau de santé encore aujourd'hui.

Les recherches ont confirmé que le problème est réel et difficile. Surtout si nous tentons une résolution optimale. Nous avons grâce à elles pu comprendre et mieux orienter nos recherches. Nos résultats sont prometteurs.

Nous avons proposé un simulateur pour modéliser le problème. Notre simulateur simule de façon efficace la démographie du Québec. Nous avons développé un algorithme qui est en mesure de traiter tous les clients simulés. Cet algorithme est très efficace pour trouver, assigner et construire les plans (horaires) pour un

vaste réseau de cliniques médicales. Nous sommes en mesure de traiter toutes les demandes dans un temps court. Un cycle de traitement prend moins de 200 millisecondes pour un lot de clients accumulés pendant 60 secondes. Le nombre de clients dans un cycle de 60 secondes peut atteindre environ 190. De plus, nous sommes en mesure de répondre à toutes les demandes de désistements formulées par les cliniques médicales. Il n'est cependant pas toujours possible de trouver un remplaçant. Ceci est dû aux contraintes. Peut-être devrions-nous apprendre de ces cas difficiles ?

Puisqu'une assignation optimale des rendez-vous est hors de portée, car trop complexe en temps de calcul, nous avons donc opté pour une méthode basée sur des heuristiques qui nous donne des solutions approximatives satisfaisantes avec des temps de calcul courts. Nous pouvons, avec notre système et l'algorithme d'assignation à base de critères, affirmer que nous sommes en mesure de réduire de façon importante les délais liés : à la découverte, au transport (temps de déplacement et distance) ainsi que le temps global d'attente. Nous pouvons aussi réduire le temps lié à la non-productivité de façon significative.

Nous sommes capables de traiter les clients, des humains, comme des humains. En effet, les clients ne sont pas des objets sans conscience. Ceci était une préoccupation et un défi tout au long de nos recherches. C'est pourquoi nous n'avons pas replanifié des clients déjà assignés.

De plus, le nombre de nouvelles demandes semblait une hypothèse intéressante à évaluer. Nous avons alors pu observer cette variable grâce au taux de remplacement. Nos expérimentations démontrent bien la tendance qu'emploie cette variable. Le taux de remplacement des désistements est impressionnant. Il est de 43%. Plus de deux patients sur cinq, qui ne se présentent pas, sont remplacés en temps réel.

Nous avons, dans ce travail, construit un simulateur qui :

1. simule la population malade du Québec ;
2. simule les points de traitement et les ressources traitantes ;
3. simule l'exécution du traitement ;

À la lumière de nos résultats, nous pouvons affirmer que nous avons réussi à concevoir un système qui est à la fois très performant dans ces algorithmes, mais aussi pleinement capable de représenter les réalités humaines.

Nos deux algorithmes ont pu démontrer une différence dans les résultats. Il aurait été superflu d'ajouter de la complexité ou d'introduire des techniques qui n'avaient pas les prérequis pour répondre à nos hypothèses initiales.

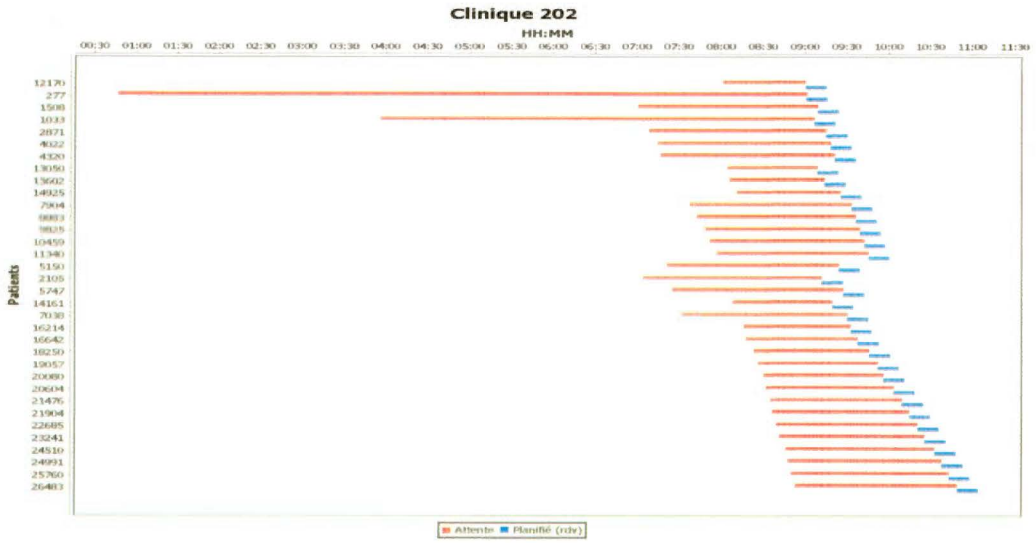
Par conséquent, nous sommes en mesure de répondre à une multitude de questions laissées ouvertes par d'autres études. L'une de ces questions étant : est-ce que la technologie et l'innovation peuvent aider ? Il va de soi que la réponse est oui, mais comment ? Nous avons répondu à cette question. Nous avons traité la question et le problème de façon globale, comme un tout. Il ne s'agit pas uniquement d'assignation ou de planification, mais bien de planification et d'assignation pour un réseau de cliniques indépendantes, qui, dans la réalité, sont aussi des concurrentes avec des buts parfois différents. De plus, nous sommes en mesure de conclure que nous avons innové favorablement à l'un des plus grands obstacles que les cliniques ambulatoires vivent quand on parle de planification optimale, de la gestion et le remplacement des abandons (Cayirli et Veral, 2003; Gupta et Denton, 2008). Nous pouvons affirmer que remplacer, avec de l'assignation de dernière minute, les désistements en temps réel est un grand pas dans la bonne direction.

Avec plus de temps et de moyens, voici quelques améliorations possibles du système et une piste pour des recherches futures :

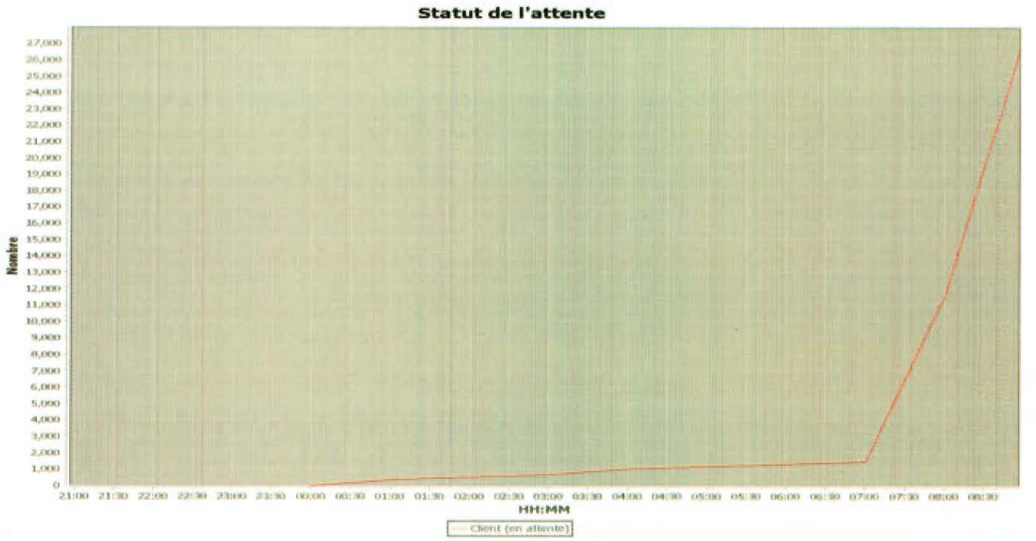
— limiter le nombre de rendez-vous, afin d'ajuster la demande à la capacité ;
— améliorer le dynamisme (temps réel) de certain visuel tel que statut de l'attente qui est générée au moment de la demande ;
— améliorer l'implémentation de l'arbre-kd lié aux cliniques situées sur la même coordonnée géographique ;
— simuler le réalisme et la fluctuation des ressources traitantes, c'est-à-dire le nombre de médecins par cliniques et par heure ;
— réutiliser la fonction (2.2) afin de replanifier des ressources déjà planifiées ;
— ajouter un module de prédiction afin d'augmenter le nombre de remplacements des désistements ou même la pertinence de ceux-ci, car le gain lié aux remplacements des désistements est prometteur.

ANNEXE A

EXPÉRIMENTATIONS VISUELLES

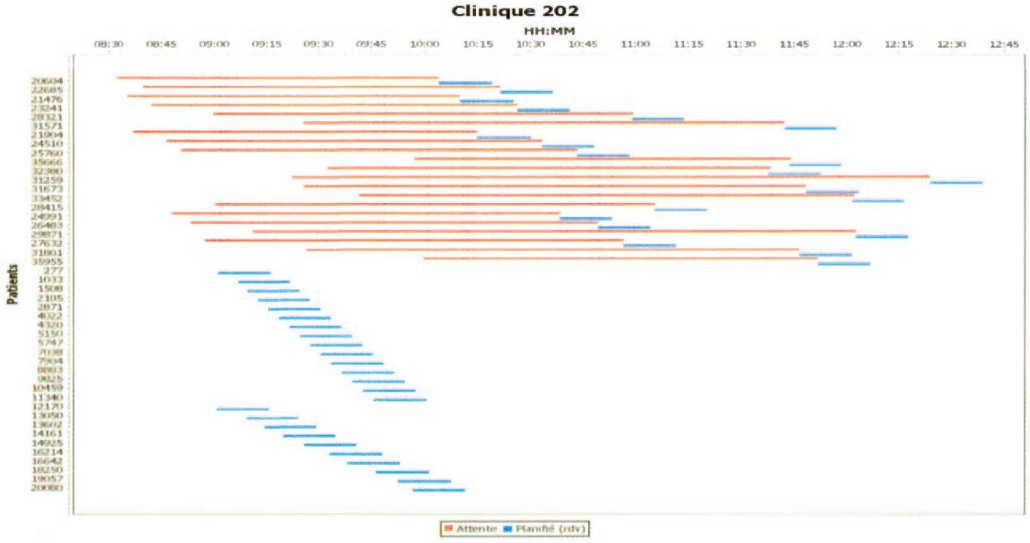


(a) Le plan Gantt des rendez-vous pour une clinique

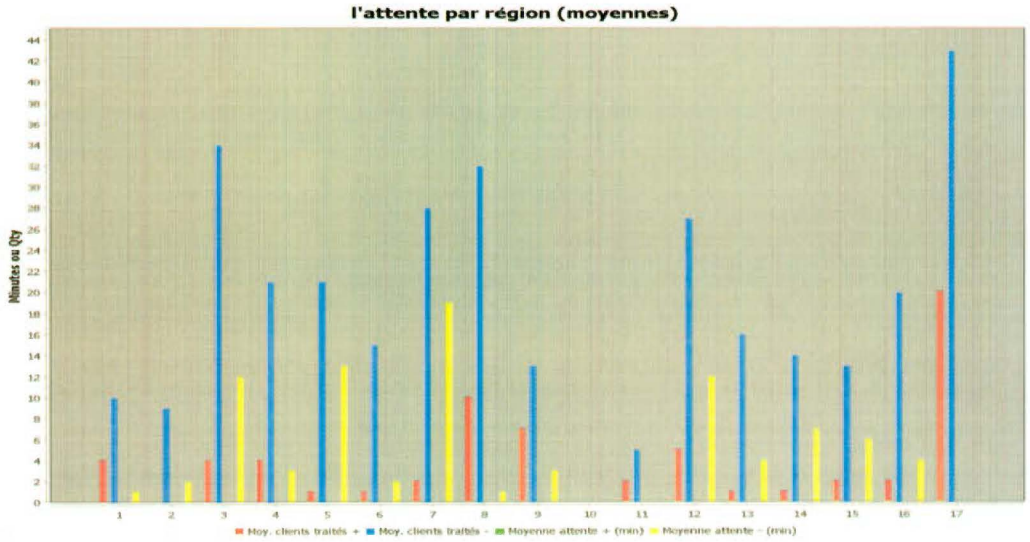


(b) Le nombre patients total dans tout le réseau

Figure A.1: Expérimentation, capture a : Jour 1 à 8h55

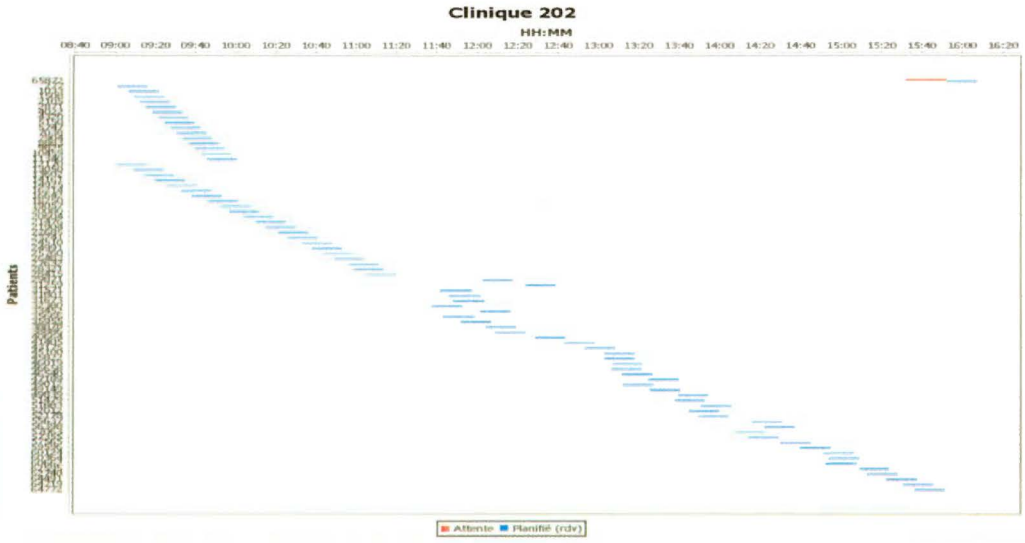


(a) Le plan Gantt des rendez-vous pour une clinique

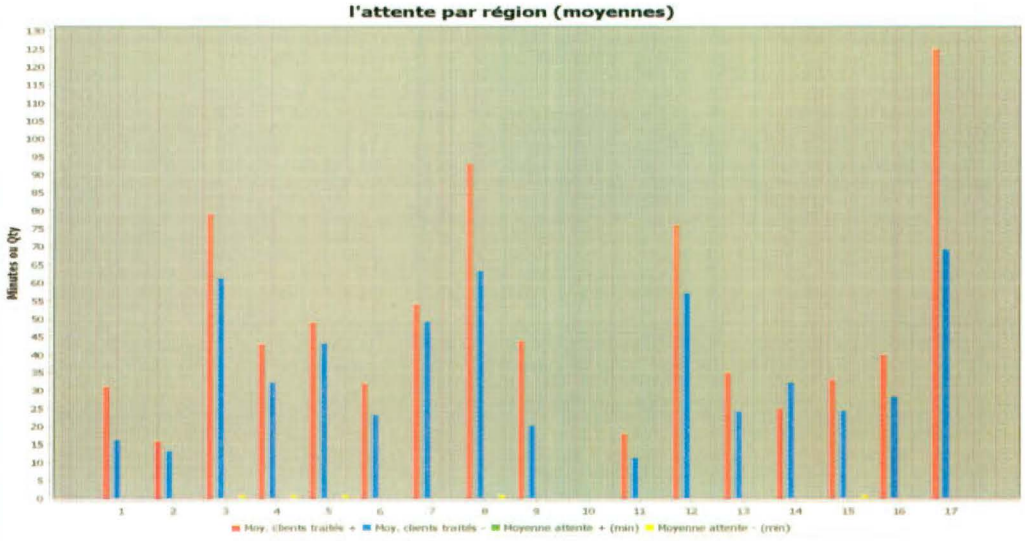


(b) Délai et le nombre de clients moyen pour l'ensemble de la région

Figure A.2: Expérimentation, capture b : Jour 1 à 10h00



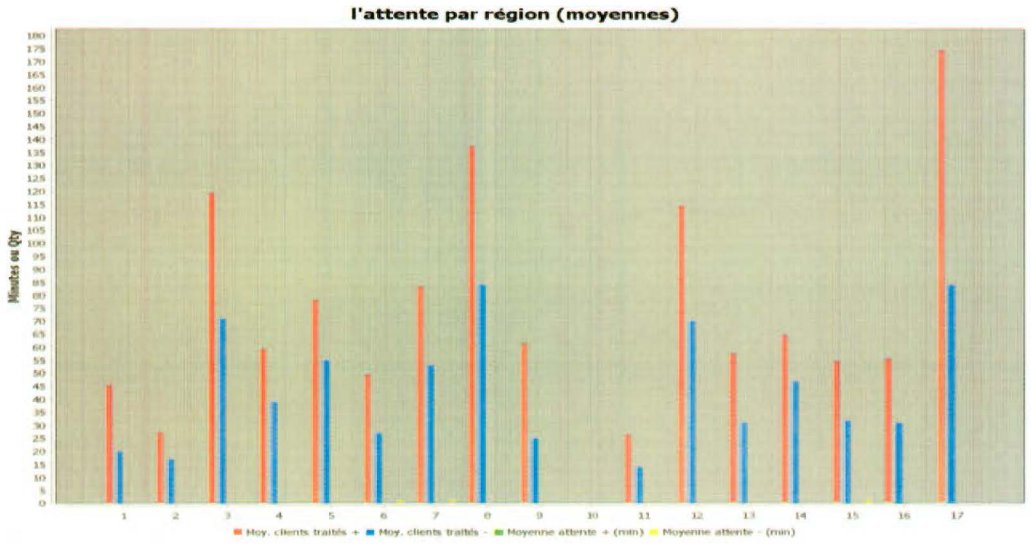
(a) Le plan Gantt des rendez-vous pour une clinique



(b) Délai et le nombre de clients moyen pour l'ensemble de la région

Figure A.3: Expérimentation, capture c : Jour 1 à 16h00

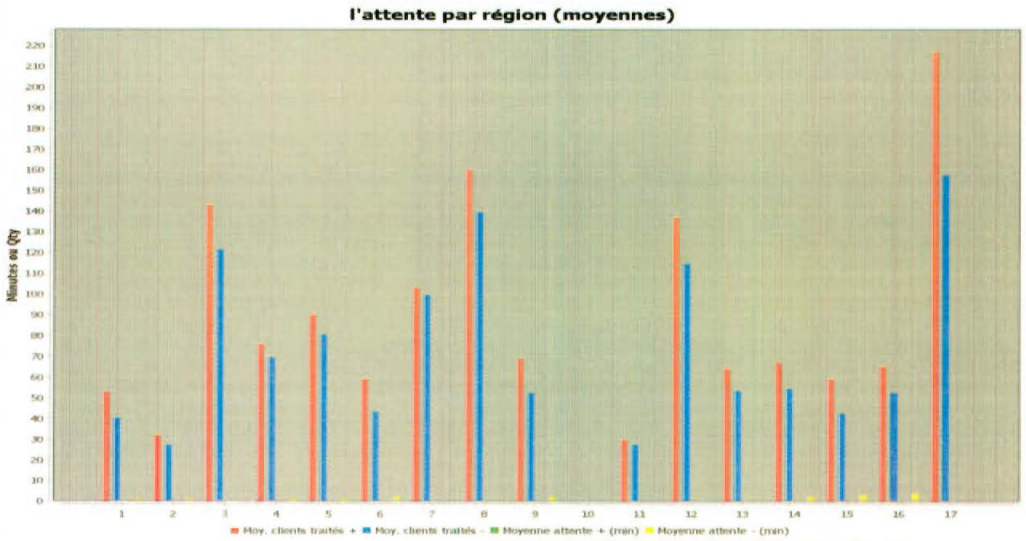
(a) Le plan Gantt des rendez-vous pour une clinique



(b) Délai et le nombre de clients moyen pour l'ensemble de la région

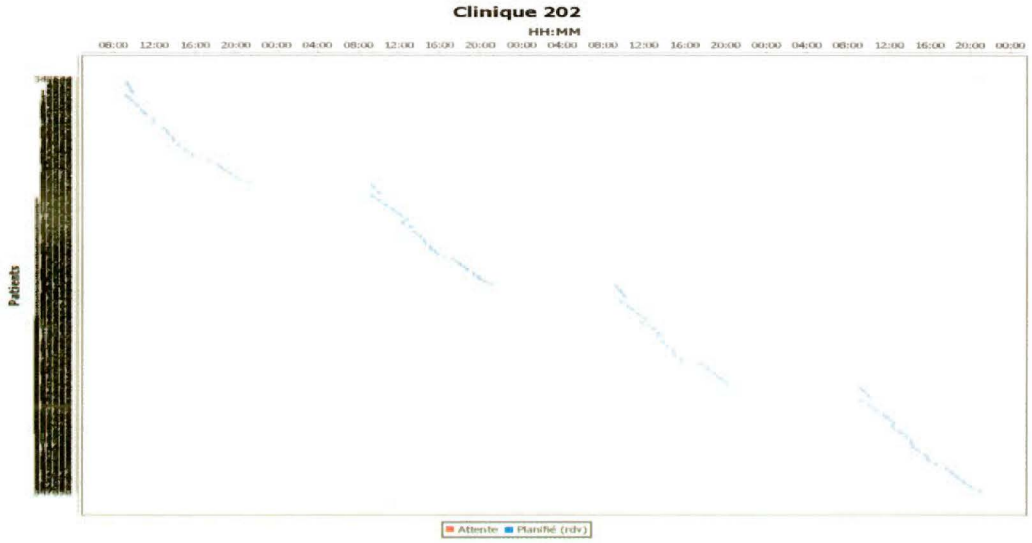
Figure A.4: Expérimentation, capture d : Jour 2 à 00h00

(a) Le plan Gantt des rendez-vous pour une clinique

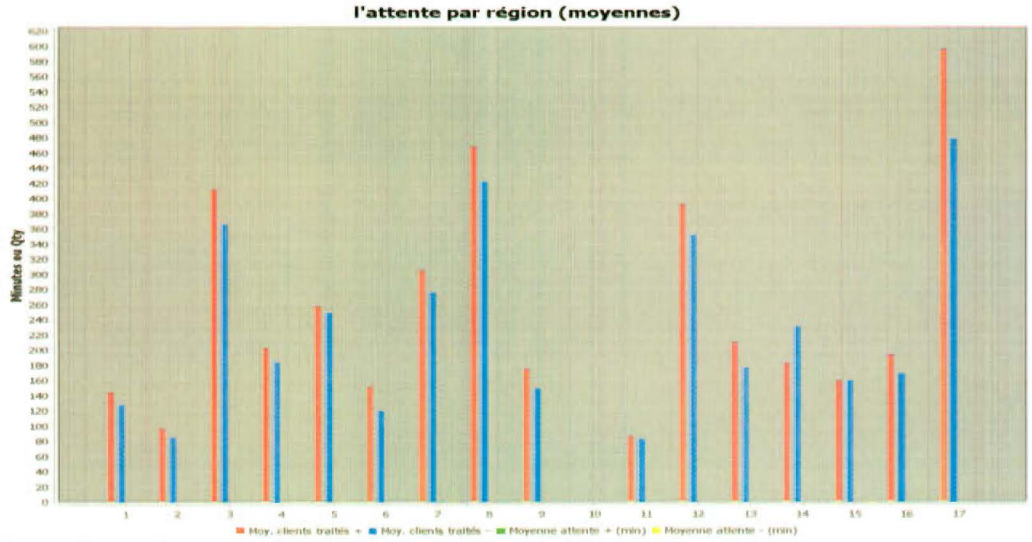


(b) Délai et le nombre de clients moyen pour l'ensemble de la région

Figure A.5: Expérimentation, capture e : Jour 2 à 12h00



(a) Le plan Gantt des rendez-vous pour une clinique



(b) Délai et le nombre de clients moyen pour l'ensemble de la région

Figure A.6: Expérimentation, capture h : Jour 4 à 22h00

GLOSSAIRE

ambulatoire	Est dit capable de se déplacer sur ses pieds, de se rendre et repartir par lui même ; implique aussi que le patient ne restera pas pour un séjour.
millisecondes	1 seconde est 1000 millisecondes (ms).
rendez-vous	Rencontre fixée en lieu et en temps entre individus.
réseau de santé	Ensemble de lieux où l'on peut servir et traiter des clients malades.
réseau de cliniques	Une portion du réseau de santé, on parle de cliniques médicales au sens général ; aussi appelé réseau.
solution	Il s'agit d'une offre, une option de résultats à considérer pour prendre une décision. Une solution est un choix dans l'ensemble des solutions.
stationnaire	Patient stationnaire, qui restera dans l'unité de soins pour la nuit, ou plus de 24 heures ; similaire à l'hospitalisation.

RÉFÉRENCES

Bailey, N. T. (1952). A study of queues and appointment systems in hospital out-patient departments, with special reference to waiting-times. *Journal of the Royal Statistical Society. Series B (Methodological)*, 185–199.

Bailey, N. T. (1954). Queueing for medical care. *Applied Statistics*, 137–145.

Baki, B. (2006). *Planification et ordonnancement probabilistes sous contraintes temporelles.* (Thèse de doctorat). Université de Caen.

Bates-Eamer, N. et Ronson, J. (2009). *Perceived Shortage, Relative Surplus : The Paradox of Quebec's Family Physician Workforce-with an intra and inter provincial comparison.* Rapport technique.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM, 18*(9), 509–517.

Boulenger, S., Castonguay, J., Dostie, B. et Vaillancourt, F. (2012). Des employés en santé, des employés productifs. *Le Québec économique 2011 : Un bilan de santé du Québec*, 125–150.

Brahimi, M. et Worthington, D. (1991). Queueing models for out-patient appointment systems-a case study. *Journal of the Operational Research Society, 42*(9), 733–746.

Brailsford, S., Churilov, L. et Dangerfield, B. (2014). *Discrete-Event Simulation and System Dynamics for Management Decision Making.* Willey.

Cayirli, T. et Veral, E. (2003). Outpatient scheduling in health care : a review of literature. *Production and Operations Management, 12*(4), 519–549.

Cayirli, T., Veral, E. et Rosen, H. (2006). Designing appointment scheduling systems for ambulatory care services. *Health care management science, 9*(1), 47–58.

Cayirli, T., Yang, K. K. et Quek, S. A. (2012). A universal appointment rule in the presence of no-shows and walk-ins. *Production and Operations Management, 21*(4), 682–697.

Chakraborty, S., Muthuraman, K. et Lawley, M. (2010). Sequential clinical scheduling with patient no-shows and general service time distributions. *IIE Transactions*, *42*(5), 354–366.

Charon, I., Germa, A. et Hudry, O. (1996). *Méthodes d'optimisation combinatoire.* Masson Paris.

Cooper, M. C., Maris, F. et Regnier, P. (2010). Solving temporally-cyclic planning problems. Dans *2010 17th International Symposium on Temporal Representation and Reasoning*, 113–120. IEEE.

Cushing, W., Kambhampati, S., Weld, D. S. *et al.* (2007). When is temporal planning really temporal? Dans *Proceedings of the 20th international joint conference on Artifical intelligence*, 1852–1859. Morgan Kaufmann Publishers Inc.

Daggy, J., Lawley, M., Willis, D., Thayer, D., Suelzer, C., DeLaurentis, P.-C., Turkcan, A., Chakraborty, S. et Sands, L. (2010). Using no-show modeling to improve clinic performance. *Health Informatics Journal*, *16*(4), 246–259.

DeLaurentis, P.-C., Kopach, R., Rardin, R., Lawley, M., Muthuraman, K., Wan, H., Ozsen, L. et Intrevado, P. (2006). Open access appointment scheduling-an experience at a community clinic. Dans *IIE Annual Conference. Proceedings*, p. 1. Institute of Industrial Engineers-Publisher.

Dyke, B. et MacCluer, J. W. (2014). *Computer simulation in human population studies.* Academic Press.

Esquirol, P. et Lopez, P. (1999). *L'ordonnancement.* Economica.

Fleury, C. (2008). Le kd-tree : une methode de subdivision spatiale. *Universite de Rennes*, *1*.

Goldman, L., Freidin, R., Cook, E. F., Eigner, J. et Grich, P. (1982). A multivariate approach to the prediction of no-show behavior in a primary care center. *Archives of Internal Medicine*, *142*(3), 563–567.

Gould, H., Tobochnik, J. et Christian, W. (1988). *An introduction to computer simulation methods*, volume 1. Addison-Wesley New York.

Green, L. (2006). Queueing analysis in healthcare. In *Patient flow : reducing delay in healthcare delivery* 281–307. Springer.

Gupta, D. et Denton, B. (2008). Appointment scheduling in health care : Challenges and opportunities. *IIE transactions*, *40*(9), 800–819.

Hall, R., Belson, D., Murali, P. et Dessouky, M. (2013). Modeling patient flows through the health care system. In *Patient Flow* 3–42. Springer.

Hassin, R. et Mendel, S. (2008). Scheduling arrivals to queues : A single-server model with no-shows. *Management Science*, *54*(3), 565–572.

Hassol, A., Walker, J. M., Kidder, D., Rokita, K., Young, D., Pierdon, S., Deitz, D., Kuck, S. et Ortiz, E. (2004). Patient experiences and attitudes about access to a patient electronic health care record and linked web messaging. *Journal of the American Medical Informatics Association*, *11*(6), 505–513.

Haussmann, R. D. (1970). Waiting time as an index of quality of nursing care. *Health services research*, *5*(2), 92.

Herriott, S. (1999). Reducing delays and waiting times with open-office scheduling. *Family practice management*, *6*, 38–43.

Hornby, G., Globus, A., Linden, D. et Lohn, J. (2006). Automated antenna design with evolutionary algorithms. In *Space 2006* p. 7242.

Johnson, B. J., Mold, J. W. et Pontious, J. M. (2007). Reduction and management of no-shows by family medicine residency practice exemplars. *The Annals of Family Medicine*, *5*(6), 534–539.

Juillard, M. et Ocaktan, T. (2008). Méthodes de simulation des modèles stochastiques d'équilibre général. *Economie & prévision*, (2), 115–126.

Kachitvichyanukul, V. (2012). Comparison of three evolutionary algorithms : Ga, pso, and de. *Industrial Engineering and Management Systems*, *11*(3), 215–223.

Keller, T. et Laughhunn, D. (1973). An application of queuing theory to a congestion problem in an outpatient clinic. *Decision Sciences*, *4*(3), 379–394.

Kennedy, J. G. et Hsu, J. T. (2003). Implementation of an open access scheduling system in a residency training program. *Family medecine Kansas-City*, *35*(9), 666–670.

Klassen, K. J. et Rohleder, T. R. (2004). Outpatient appointment scheduling with urgent clients in a dynamic, multi-period environment. *International Journal of Service Industry Management*, *15*(2), 167–186.

Kodjababian, J. G. (2003). Improving patient access and continuity of care : A successful implementation of open access scheduling. *Egg Management Consultants, Inc.*

Lacy, N. L., Paulman, A., Reuter, M. D. et Lovejoy, B. (2004). Why we don't come : patient perceptions on no-shows. *The Annals of Family Medicine*, *2*(6), 541–545.

LaGanga, L. R. et Lawrence, S. R. (2007a). Appointment scheduling with overbooking to mitigate productivity loss from no-shows. Dans *Proceedings of Decision Sciences Institute Annual Conference, Phoenix, Arizona.*

LaGanga, L. R. et Lawrence, S. R. (2007b). Clinic overbooking to improve patient access and increase provider productivity*. *Decision Sciences*, *38*(2), 251–276.

Lee, V. J., Earnest, A., Chen, M. I. et Krishnan, B. (2005). Predictors of failed attendances in a multi-specialty outpatient centre using electronic databases. *BMC health services research*, *5*(1), 1.

Lopez, P. et Roubellat, F. (2001). *Ordonnancement de la production.* Hermès science publications.

McQuarrie, D. (1983). Hospitalization utilization levels. the application of queuing. theory to a controversial medical economic problem. *Minnesota Medicine*, *66*(11), 679–686.

Montecinos, J., Ouhimmou, M. et Chauhan, S. (2015). Waiting-time estimation in walk-in clinics. *International Transactions in Operational Research*, *25*(1), 51–74.

Moore, C. G., Wilson-Witherspoon, P. et Probst, J. C. (2001). Time and money : effects of no-shows at a family practice residency clinic. *Family Medicine-Kansas City*, *33*(7), 522–527.

Murray, M. et Berwick, D. M. (2003). Advanced access : reducing waiting and delays in primary care. *Jama*, *289*(8), 1035–1040.

Murray, M., Bodenheimer, T., Rittenhouse, D. et Grumbach, K. (2003). Improving timely access to primary care : case studies of the advanced access model. *Jama*, *289*(8), 1042–1046.

Murray, M. et Tantau, C. (2000). Same-day appointments : exploding the access paradigm. *Family practice management*, *7*(8), 45–45.

Nunes, J., Matos, L. et Trigo, A. (2011). Taxi pick-ups route optimization using genetic algorithms. Dans *International Conference on Adaptive and Natural Computing Algorithms*, 410–419. Springer.

O'Hare, D. et Corlett, J. (2004). The outcomes of open-access scheduling. *Family practice management*, *11*(2), 35.

Qu, X., Rardin, R. L., Williams, J. A. S. et Willis, D. R. (2007). Matching daily healthcare provider capacity to demand in advanced access scheduling systems. *European Journal of Operational Research*, *183*(2), 812–826.

Rising, E. J., Baron, R. et Averill, B. (1973). A systems analysis of a university-health-service outpatient clinic. *Operations Research*, *21*(5), 1030–1047.

Roy, B. (2013). *Multicriteria methodology for decision aiding*, volume 12. Springer Science & Business Media.

Samorani, M. et LaGanga, L. R. (2015). Outpatient appointment scheduling given individual day-dependent no-show predictions. *European Journal of Operational Research*, *240*(1), 245–257.

Siddharthan, K., Jones, W. J. et Johnson, J. A. (1996). A priority queuing model to reduce waiting times in emergency care. *International Journal of Health Care Quality Assurance*, *9*(5), 10–16.

Ulungu, E. L. et Teghem, J. (1994). Multi-objective combinatorial optimization problems : A survey. *Journal of Multi-Criteria Decision Analysis*, *3*(2), 83–104.

Wall, M. B. (1996). *A genetic algorithm for resource-constrained scheduling.* (Thèse de doctorat). Massachusetts Institute of Technology.

Wang, P. P. (1997). Optimally scheduling n customer arrival times for a single-server system. *Computers & Operations Research*, *24*(8), 703–716.

White, M. B. et Pike, M. (1964). Appointment systems in out-patients' clinics and the effect of patients' unpunctuality. *Medical Care*, 133–145.

Worthington, D. (1991). Hospital waiting list management models. *Journal of the Operational Research Society*, *42*(10), 833–843.

Zeng, B., Turkcan, A., Lin, J. et Lawley, M. (2010). Clinic scheduling models with overbooking for patients with heterogeneous no-show probabilities. *Annals of Operations Research*, *178*(1), 121–144.